



TUM School of Engineering and Design

# Deep Learning based Dense Matching Optimization in Remote Sensing

Yuanxin Xia, M.Sc.

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung des akademischen Grades eines

**Doktors der Ingenieurwissenschaften**

genehmigten Dissertation

Vorsitzender: Prof. Dr.phil.nat. Urs Hugentobler

Prüfer der Dissertation:

1. Prof. Dr.-Ing.habil. Richard H. G. Bamler
2. Assoc. Prof. Dr.techn. Friedrich Fraundorfer,  
Technische Universität Graz, Österreich
3. Prof. Dr.-Ing. Peter Reinartz,  
Universität Osnabrück

Die Dissertation wurde am 02.12.2021 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 13.04.2022 angenommen.



## Abstract

Inspired by human stereo vision, dense matching technique is used to reconstruct depth information from overlapping images. Dense matching locates corresponding pixels in both stereo images. The image coordinate difference, termed as disparity in computer vision, can then be calculated and transformed to the depth value via known sensor orientation parameters, which is further simplified as the column difference for rectified images. The technique is widely applied in the field of remote sensing to create the digital surface models, reconstruct the 3D geometry of the landscapes, detect and monitor the surface deformation, etc., considering the sufficient data supply from advanced airborne and satellite platforms and the increasing demands in earth observation.

The central step of the stereo method is the correspondence determination. Through the last decades, the matching technique was developed from simply comparing single pixels or small image patches to additionally introducing a smoothness term for a globally consistent depth distribution, solving ambiguous correspondence assignments in difficult situations, for example, textureless, occluded or noisy image areas, but leading to higher computational cost. A major step forward was the introduction of Semi-Global Matching in 2005, which takes into account pixel similarity and spatial smoothness together at linear runtime. Due to its efficiency and robustness, it is widely used in many close range and remote sensing applications.

Deep learning brings a new concept for vision tasks, by learning from annotated samples in a supervised manner to acquire experience before tested on relevant data. Thus, it leads to new possibilities for dense matching, via deeply interpreting the stereo pair and extracting high-level features for pixel comparison and adaptive disparity determination. On the other hand, Semi-Global Matching consists of a sequence of fixed but replaceable modules, hence, the method leaves interfaces to be combined with a learning scheme for higher performance in dense matching. The thesis extends Semi-Global Matching by using recently machine learning techniques at different stages of the matching algorithm. With a main focus on remote sensing problems, three strategies have been designed.

As mentioned above, a learning based pixel comparison for matching cost calculation is the most intuitive attempt to enhance Semi-Global Matching. In remote sensing scenarios, however, the reference data collection is difficult to train a neural network, especially in challenging area with dynamic scene composition. To handle this problem, a self-supervised strategy is proposed to train a state-of-the-art matching cost network.

Semi-Global Matching simplifies the procedure to pursue two dimensional smoothness, via multiple one dimensional scanline optimization which could be finished within reasonable runtime consumption. The final fusion of the scanlines is, nevertheless, an ad-hoc step without a theoretical background. Therefore, we leverage a random forest to select only the most relevant scanlines, leading to improved disparity estimation.

State-of-the-art learning based stereo algorithms acquire high accuracy for disparity prediction. However, the efficiency normally cannot support the stereo processing of remote sensing data, considering the large disparity ranges and image dimension of typical remote sensing images. Regarding this dilemma, we design a pyramid architecture to approximate Semi-Global Matching within an end-to-end neural network. The disparity is estimated from coarse to fine, which highly enhances the efficiency and allows processing of big images with large disparity ranges.

# Zusammenfassung

In Anlehnung an das menschliche binokulare Sehen werden Methoden des maschinellen Stereosehens verwendet um Tiefeninformation aus überlappenden Bildern zu berechnen. Beim Dense Matching werden die einander entsprechenden Pixel in beiden Stereobildern lokalisiert. Die Bildkoordinatendifferenz, die in der Bildverarbeitung als Disparität bezeichnet wird, kann dann berechnet und mittels bekannten Kameraparametern in den Tiefenwert umgewandelt werden.

Diese Technik wird in der Fernerkundung häufig eingesetzt, um digitale Oberflächenmodelle zu erstellen, die 3D-Geometrie von Landschaften zu rekonstruieren, Oberflächenverformungen zu erkennen und zu überwachen usw., da die Daten von modernen luftgestützten und satellitengestützten Sensoren in ausreichender Menge zur Verfügung stehen und die Bilddaten in immer höherer Qualität aufgenommen werden.

Der zentrale Schritt der Stereomethode ist die Lösung des Korrespondenzproblems. Im Laufe der letzten Jahrzehnte wurde die Matching-Technik vom einfachen Vergleich einzelner Pixel oder kleiner Bildausschnitte durch Einführung eines räumlichen Regularisierungsschrittes einer global konsistenten Tiefenschätzung weiterentwickelt, wodurch mehrdeutige Korrespondenzzuweisungen in schwierigen Situationen, z. B. bei texturlosen, verdeckten oder verrauschten Bildbereichen, gelöst werden konnten, was jedoch zu höheren Rechenzeiten führte. Ein großer Schritt nach vorn war die Einführung des Semi-Global Matching (SGM) im Jahr 2005, das die Pixelähnlichkeit und die räumliche Regularisierung bei linearer Laufzeit gemeinsam berücksichtigt. Aufgrund seiner Effizienz und Robustheit wird es in vielen Nahbereichs- und Fernerkundungsanwendungen eingesetzt.

Auf Deep Learning basierte Algorithmen lernen anhand von Trainingsdaten die Lösung von Bildverarbeitungsaufgaben. Damit können auf Deep Learning basierte Verfahren zur dichten Stereokorrespondenzschätzung auch übergeordnete Information und Strukturen in den Daten erkennen und für die Disparitätsbestimmung verwenden. Semi-Global Matching besteht aus mehreren Einzelschritten, welche unabhängig voneinander implementiert werden können. Ein oder mehrere Schritte im SGM Algorithmus können nun durch Deep oder Maschine Learning basierte Verfahren ersetzt werden. Diese Dissertation untersucht verschiedene Möglichkeiten zur Verbesserung des SGM durch Deep Learning basierte Module mit einem Hauptaugenmerk auf die Anwendung für Fernerkundungsprobleme.

Ein erster, erfolgreicher Schritt war die Berechnung der Matching-Kosten für durch Neuronale Netze. In der Fernerkundung ist jedoch die Sammlung für eine erfolgreiches Training nötigen, hochgenauen Trainingsdaten insbesondere für dynamischen Szenen schwierig. Um dieses Problem zu lösen, wird ein selbstüberwachte Trainingstrategie entwickelt, mit der ein modernes Netzwerk zur Matchingkostenberechnung auf selbständig auf neue Datensätze adaptiert werden kann.

Semi-Global Matching approximiert eine 2-dimensionale Regularisierung durch mehrfache 1D Scanline-Aggregation aus unterschiedlichen Richtungen und erreicht damit eine lineare Laufzeit bezogen auf das rekonstruierte Volumen. Die Kombination der verschiedenen Aggregationsrichtungen wird durch eine einfache Addition erreicht, wodurch auch Scanlinien mit fehlerhaften Schätzungen mit einbezogen werden. Daher wird in dieser Arbeit ein Multi-Class Random-Forest Klassifikator vorgeschlagen, der die korrekten Aggregationsrichtungen bestimmt, um fehlerhafte Richtungen auszuschliessen und somit zu einer verbesserten Disparitätschätzung führt.

Moderne lernbasierte Stereo-Algorithmen erreichen eine hohe Genauigkeit bei der Disparitätsvorhersage, benötigen aber eine viel GPU-Speicher für die Verarbeitung relativ kleiner Bilder. Angesichts der großen Disparitätsbereiche und Bilddimensionen typischer

Fernerkundungsbilder sind sie deshalb normalerweise nicht für die Stereoverarbeitung von Fernerkundungsdaten geeignet. In Anbetracht dieses Dilemmas entwerfen wir eine Pyramidenarchitektur zur Annäherung an das Semi-Global Matching innerhalb eines Ende-zu-Ende neuronalen Netzes. Die Disparität wird von grob bis fein geschätzt, was die Effizienz erheblich steigert und die Verarbeitung großer Bilder mit großen Disparitätsbereichen ermöglicht.

---

# List of Abbreviations

---

Abbreviation	Description
AI	Artificial Intelligence
BP	Belief Propagation
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
CNN	Convolutional Neural Networks
CBCA	Cross based Cost Aggregation
CRF	Conditional Random Field
CSPN	Convolutional Spatial Propagation Network
DSI	Disparity Space Image
DSM	Digital Surface Model
EPE	End Point Error
FC	Fully-Connected
FOV	Field of View
FPGA	Field-Programmable Gate Array
GCP	Ground Control Point
GSD	Ground Sampling Distance
KNN	K Nearest Neighbor
LiDAR	Light Detection and Ranging
LSTM	Long Short-Term Memory
MC-CNN	Matching Cost based on Convolutional Neural Networks
MRF	Markov Random Field
NCC	Normalized Cross Correlation
PDF	Probability Density Function
ReLU	Rectified Linear Unit
RF	Random Forest
RNN	Recurrent Neural Networks
SAD	Sum of Absolute Differences
SDC	Stacked Dilated Convolutions

---

---

Abbreviation	Description
SGM	Semi-Global Matching
SGM-Forest	Semi-Global Matching based on Random Forest
SLAM	Simultaneous Localization and Mapping
SO	Scanline Optimization
SOTA	State-of-the-Art
SPN	Spatial Propagation Network
SPP	Spatial Pyramid Pooling
SVM	Support Vector Machine
SSVM	Structured Support Vector Machine
VNIR	Visible and Near Infrared
WTA	Winner-Take-All
ZSAD	Zero-Mean Sum of Absolute Differences

---

# Contents

<b>Abstract</b>	i
<b>Zusammenfassung</b>	ii
<b>List of Abbreviations</b>	iv
<b>1 Introduction</b>	1
1.1 Motivations and Research Challenges	1
1.2 Objectives	3
1.3 Dissertation Organization	4
<b>2 Basics</b>	6
2.1 Binocular Stereo	6
2.1.1 <i>Matching Cost Computation</i>	7
2.1.2 <i>Cost (Support) Aggregation and Disparity Computation</i>	8
2.1.3 <i>Post-Processing</i>	12
2.2 Machine/Deep Learning Assisted Stereo	13
2.3 Monocular and Multi-View Stereo	15
<b>3 State-of-the-Art in Learning based Dense Matching</b>	17
3.1 Enhancement of existing algorithms through Machine/Deep Learning	17
3.1.1 <i>Learning based Matching Cost</i>	17
3.1.2 <i>Learning based Cost Aggregation and Regularization</i>	20
3.1.3 <i>Learning based Post-Processing</i>	23
3.2 End-to-End Neural Networks	24
3.2.1 <i>3D Cost Volume (2D Convolution based Networks)</i>	25
3.2.2 <i>4D Cost Volume (3D Convolution based Networks)</i>	27
3.2.3 <i>Multi-Task Learning</i>	31
3.3 Confidence Measurement	33
3.4 Cross Domain Estimation	35
3.5 Datasets	37
3.5.1 <i>Real Stereo Data</i>	38
3.5.2 <i>Synthetic Stereo Data</i>	40
<b>4 Summary of the Contributions</b>	42
4.1 Self-Supervised Matching Cost Network with Case Study: Plant Reconstruction (ForDroughtDet)	42
4.1.1 <i>Background</i>	42
4.1.2 <i>Dense Matching based on MC-CNN and SGM</i>	43
4.1.3 <i>Training Strategy with Ground Truth</i>	45
4.1.4 <i>Training Strategy without Ground Truth</i>	45
4.1.5 <i>Case Study: Plant Reconstruction and Drought Detection</i>	47
4.1.6 <i>Discussion and Outlook</i>	57
4.2 Adaptive Scanlines Selection in Semi-Global Matching	58
4.2.1 <i>Background</i>	59
4.2.2 <i>Limitations of SGM</i>	59
4.2.3 <i>Limitations of SGM-Forest</i>	60
4.2.4 <i>Multi-Label Classification based Scanlines Determination</i>	62
4.2.5 <i>Performance Evaluation on Multiple Data Sources</i>	65
4.2.6 <i>Conclusion and Outlook</i>	75



---

4.3	End-to-End Hierarchical Disparity Estimation and Refinement	75
4.3.1	<i>Background</i>	76
4.3.2	<i>Differentiable Approximation of SGM</i>	76
4.3.3	<i>Efficiency Enhancement via Coarse-to-Fine Strategy</i>	78
4.3.4	<i>Performance Evaluation on Multiple Data Sources</i>	80
4.3.5	<i>Conclusion and Outlook</i>	89
<b>5</b>	<b>Conclusion and Outlook</b>	93
5.1	Conclusion	93
5.2	Outlook	95
	<b>References</b>	97
	<b>Acknowledgement</b>	108
	<b>Appendices</b>	114
<b>A</b>	<b>Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2019. Self-supervised convolutional neural networks for plant reconstruction using stereo imagery. <i>Photogrammetric Engineering &amp; Remote Sensing</i>, 85(5), pp.389-399.</b>	115
<b>B</b>	<b>Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2020. Multi-label learning based semi-global matching forest. <i>Remote Sensing</i>, 12(7), p.1069.</b>	155
<b>C</b>	<b>Xia, Y., d'Angelo, P., Fraundorfer, F., Tian, J., Fuentes Reyes, M. and Reinartz, P., 2022. GA-Net-Pyramid: an efficient end-to-end network for dense matching. <i>Remote Sensing</i>, 14(8), p.1942.</b>	179



# 1 Introduction

Dense stereo matching aims at estimating the depth of every single pixel of the images for 3D reconstruction. The technique is broadly applied in the field of computer vision and remote sensing, since it complements the information gap from 2D imagery with complete 3D knowledge of the world, providing additional clues to assist vision tasks (d'Angelo and Reinartz, 2012; Yang et al., 2018; Song et al., 2020b). On the other hand, the depth information could be acquired simply via images rather than the active sensing sensors, e.g. Light Detection and Ranging (LiDAR), structured light projection, etc., which are more expensive and demanding for data collection especially when high point density is required. The depth of the object can be naturally perceived by humans according to the visual difference perceived between two eyes, thus 3D stereo information is acquired for better understanding the scene rather than utilizing the pure 2D information. Inspired by human visual perception, dense matching exploits two cameras (or a single camera capturing the scene at different positions) to obtain image sequences with overlap in between, from which the displacement between the corresponding points is computed to recover the object depth. Accordingly, the central step is locating the dense correspondences between stereo images, which remains to be an open problem through decades due to the practical difficulties such as lack of distinguishable texture, occlusion induced by different view points, radiometric noises, etc. (Bleyer and Breiteneder, 2013).

## 1.1 Motivations and Research Challenges

With known camera intrinsic and extrinsic parameters, the stereo image pair can be rectified such that the corresponding points lie on the same row (epipolar line) of the left and right images. Thus, locating the correspondence is simplified as a 1D problem, leading to a stereo result containing the horizontal displacement of each corresponding pixel pair, namely disparity. In the field of computer vision, the term disparity is mostly used which is inversely proportional to the distance of the corresponding object point from the camera (Bleyer and Breiteneder, 2013). Hence in this thesis, we also mainly utilize disparity instead of depth since the former is more straightforward from the matching results, simply the coordinate difference between corresponding points. The transformation in between is simple if the camera focal length and the stereo baseline are available. Also, we regard the input stereo pair as rectified to exclude extra complexity and focus on the matching strategies. Stereo matching has been applied in various fields to compute the parameters of interest, e.g. the land height via subtracting the object depth from the flight height for Digital Surface Model (DSM) generation in geodesy, the 3D coordinates of the scene points to create delicate object models in virtual reality, the pose estimation for navigation and obstacle evasion in robotics and self-driving, etc. A disparity map of an indoor tree is shown in Figure 1.1, for which the color represents the disparity value of each point. The disparity map makes it intuitive to perceive the distance of each leaf to the viewing camera, from long range to close position with the color from blue to yellow.

Stereo matching is broadly applied in practice, due to its simple device setup requiring inexpensive hardware (Murray and Little, 2000; Hirschmüller, 2011). Off-the-shelf cameras or even smart phones can already supply the stereo input for 3D sensing. On the other hand, the passive sensing principle makes this technique independent from other sensors without interference. Therefore, the integration of multiple information sources is possible for data fusion based estimation (Schmid et al., 2013). Among the decades, stereo matching was firstly applied as a stereo plotter (Kelly et al., 1977), which naturally exploited its 3D attributes to better understand the landscapes in cartography. As time goes by, the tech-

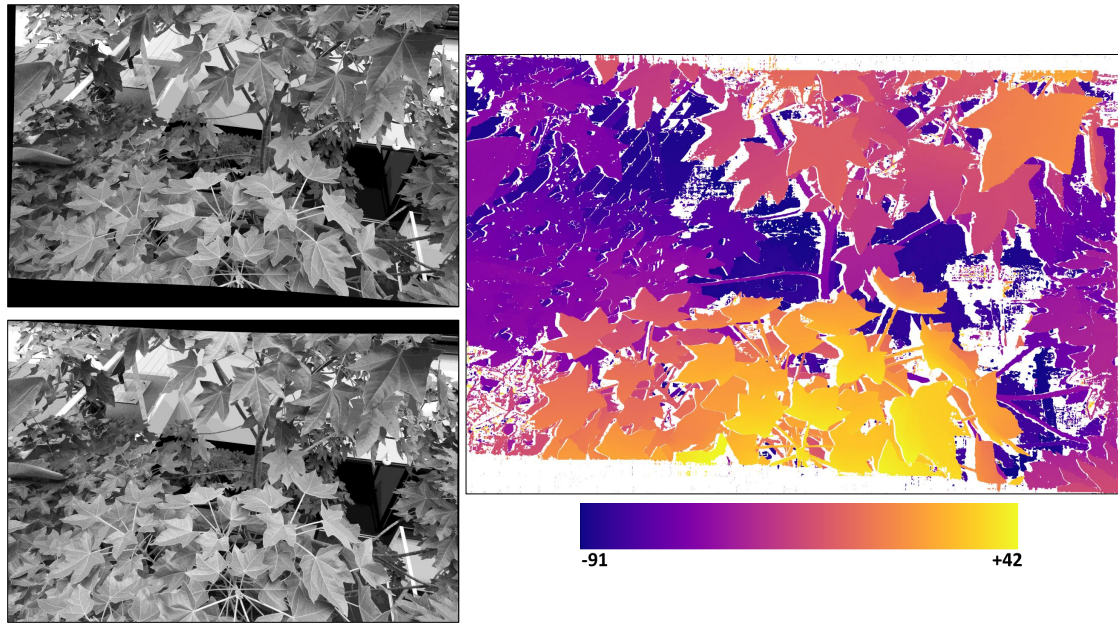


Fig. 1.1. Stereo results of an indoor tree. The two images on the left constitute a rectified stereo pair, in which the corresponding points lie on the same row. The stereo disparity map is displayed on the right, from which the depth of the scene can be perceived. The color bar at the bottom represents the disparity range.

nique was also used to support large scale object detection for smart monitoring (Hengstler et al., 2007) or digital surface/terrain model generation in topography mapping until recent years (Gehrke et al., 2010; Shean et al., 2016). Along with the development of related research areas including navigation, SLAM, etc., stereo matching was also utilized for artificial intelligence. German Aerospace Center (DLR) proposed the DLR crawler, which employed a synchronized stereo sensor to guide the robots in extreme terrains for search and rescue (Görner et al., 2010). Schmid et al. (2013) combined the stereo estimation from a Field-Programmable Gate Array (FPGA) based matching algorithm and an Inertial Measurement Unit (IMU), to provide the environmental knowledge for flying robots, so that autonomous navigation was realized for path planning and obstacles evasion. The stereo correspondences could be further extended to the complete scene flow information for self-driving (Menze and Geiger, 2015). In addition to the explicit depth products, stereo matching can also be combined with other advanced vision tasks for more robust prediction, e.g. semantic segmentation (Yang et al., 2018), edge detection (Song et al., 2020b), etc.

Although the dense matching technique has obtained a broad and promising practical significance through numerous fields, it should be admitted that the disparity estimation can still be erroneous even when assisted by powerful learning strategies (Laga et al., 2020; Poggi et al., 2020). Locating the correspondences is natural for human vision, however, challenging for computers considering different imaging issues especially when the stereo acquisition is sub-optimal. As one of the most heavily studied topics in computer vision (Scharstein and Szeliski, 2002), dense matching draws research attention to disclose the factors restricting the algorithms' performance. Starting from the object space, photometric or geometric matching ambiguities may happen depending on the object materials. Firstly, the target region could be textureless, e.g. a white wall, or reflective/specular surface, e.g. windows or mirrors, from which there is no distinctive clues to compare and locate the corresponding pixels of the same object. Besides, the texture can be uniform, repetitive or periodic. Thus the neighboring pixels may have similar appearance to confuse the correspondence determination. It can be imagined that even humans feel difficult to judge the scene's depth in above situations. In addition, the object could be visible only in one image of the stereo pair due to different viewing angles, which leads to occlusion and further increases the matching difficulties. The visualization of the mentioned challenges is displayed in Fig-



Fig. 1.2. Challenges in stereo matching, including repetitive texture as the grids on the flag, part of which is occluded by the tree crown (occlusion also happens among leaves), reflective surface on the window, and textureless region on the wall.

Moreover, problems emerge during the data collection, such as the sensor noise which contaminates the image intensity. It is also possible that the pixels from a certain target have varied values among multi-view images, if the illumination or exposure conditions cannot be rigidly controlled between the stereo acquisition, or simply caused by different camera gains and biases or reflectance properties. Last but not the least, it could be dilemmatic to decide the stereo strategies. Traditional matching algorithms (Hirschmüller, 2008; Rothermel, 2017) have steady and robust performance across diverse data types and practical scenarios, however, suffer from challenging regions, e.g. shadows, occlusions, depth discontinuities, etc. Better stereo results are obtained by learning based methods, including pure data driven models such as (Dosovitskiy et al., 2015; Mayer et al., 2016) or hybrid of learning and engineered algorithms (Kendall et al., 2017; Seki and Pollefeys, 2017; Chang and Chen, 2018). Nevertheless, a large amount of well-annotated training data is needed to reach the network's best performance, which is cumbersome and time-consuming. It should be mentioned that the top-ranking methods on Middlebury benchmark (Scharstein et al., 2014) are still conventional ones, when limited training data are available.

## 1.2 Objectives

In order to promote the development of dense stereo matching, the thesis stays with the machine/deep learning (Mitchell, 1997; Goodfellow et al., 2016) side to utilize its powerful data representation ability for better feature extraction, cost aggregation, etc, to search the dense correspondences between the stereo pair. Following the pipeline of a general stereo algorithm (see Chapter 2), the thesis aims at improving the state-of-the-art methods regarding each stereo unit or an overall optimization, with a focus on remote sensing applications. The objectives of the thesis are summarized as follows:

### ◇ Learning based matching cost optimization

Matching cost calculation is the first step in stereo matching, which measures the photometric consistency between potentially matching points. With learning based algorithms, high level features are extracted for pixel comparison leading to much better performance than using pure image intensity. However, suitable training data is not always available to supervise a well-performed model, especially in the field of remote sensing.

Therefore, a self-supervised strategy is promising to train a learning based matching cost algorithm, without the need of cumbersome ground truth data collection.

ForDroughtDet is a research project (FKZ: 22WB410602), which concentrates on detecting the physiological and morphological status of trees under drought stress according to the geometric deformation. The research target is wild trees, thus common 3D detection technique such as LiDAR is not feasible for a continuous laser scanning to acquire the point cloud, due to possible shaking of the leaves caused by wind. Hence, the project raises a practical challenge, to construct 3D models of wild trees, for which a self-supervised stereo matching strategy is appropriate.

#### ◇ **Learning based cost regularization**

In addition to measuring the similarity of the target pixels, the disparity distribution within the neighborhood should also be considered to guarantee a smooth result. Semi-Global Matching (SGM) well approximates a 2D Markov Random Field (MRF) via multiple 1D scanline optimizations. However, the quality of each scanline's prediction varies a lot, depending on the specific scene structure. SGM empirically sums up the energy of all the scanlines without a theoretical background, which can result in inaccurate depth prediction, e.g. at slanted surfaces, disparity discontinuities, etc. Hence, a learning based algorithm should be proposed to adaptively select better performing scanlines and ignore the others, for further disparity estimation.

#### ◇ **Learning based end-to-end stereo matching**

The whole stereo matching pipeline can be approximated as a differentiable and trainable deep neural network, to directly predict a disparity map from a stereo pair. With every single stereo module supervised, the performance is further enhanced, even achieving accurate depth estimation in ill-posed regions. Nevertheless, most state-of-the-art end-to-end methods are only tested on close-range data, for which the great performance is not well generalized in remote sensing for airborne and spaceborne stereo tasks. Considering the large data amount and wide stereo baselines, the stereo matching can be memory-hungry and time-consuming. Therefore, an efficient end-to-end neural network is promising to handle large scale remote sensing data with reasonable runtime consumed.

Considering the above mentioned problems of lacking training data for learning strategies, a self-supervision scheme, a simple model with low training data demand, and a highly efficient neural networks trained by synthetic or automatically annotated ground truth are proposed. In addition, a remote sensing project is used as a case study to support the thesis from a practical point of view.

## 1.3 Dissertation Organization

The dissertation is based on the Ph.D's research work in dense stereo matching, which is tested and applied in the fields of remote sensing and computer vision. 3 published peer-reviewed journal papers support the dissertation:

- ◇ Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2019. Self-supervised convolutional neural networks for plant reconstruction using stereo imagery. *Photogrammetric Engineering & Remote Sensing*, 85(5), pp.389-399.
- ◇ Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2020. Multi-label learning based semi-global matching forest. *Remote Sensing*, 12(7), p.1069.
- ◇ Xia, Y., d'Angelo, P., Fraundorfer, F., Tian, J., Fuentes Reyes, M. and Reinartz, P., 2022.

---

**GA-Net-Pyramid: An efficient end-to-end network for dense matching. Remote Sensing, 14(8), p.1942.**

The publications are attached in the appendix. In Chapter 2, the theoretical background of dense matching is provided, with an emphasis on binocular stereo, and a brief introduction of monocular and multi-view stereo. Chapter 3 reviews the state-of-the-art stereo matching techniques and the corresponding stereo datasets for benchmarking the algorithms. Then the Ph.D's work is summarized in Chapter 4, which keeps promoting the development of dense matching, from matching cost optimization, adaptive cost aggregation, to full end-to-end depth estimation based on neural networks. At last, Chapter 5 concludes the thesis and points out the future research.

## 2 Basics

This chapter introduces the basics in dense matching, from theoretical processing steps to concrete schemes in practice. Classical stereo techniques are introduced, followed by recent learning based research. The fundamentals guide the Ph.D's work to propose new methods and strategies for dense matching. At the end, monocular and multi-view stereo are briefly reviewed, as the main focus is on binocular methods.

### 2.1 Binocular Stereo

In computer vision, stereo images could be obtained using two cameras with a certain displacement in between as the stereo baseline. Binocular dense matching basically locates the stereo correspondences between two images, which are essentially the projections from the same object points. Based on assumptions of camera calibration and epipolar rectification, the term disparity is proposed as the horizontal relative displacement of corresponding pixels. Thus, a disparity map is calculated containing the disparity values of all the pixels, which can be easily transformed to the depth of the scene as Equation 2.1 (Hartley and Zisserman, 2004). Accordingly, a virtual 3D view could be obtained for better scene understanding as shown in Figure 2.1.

$$Depth = \frac{Baseline \times Focal\ Length}{Disparity}. \quad (2.1)$$

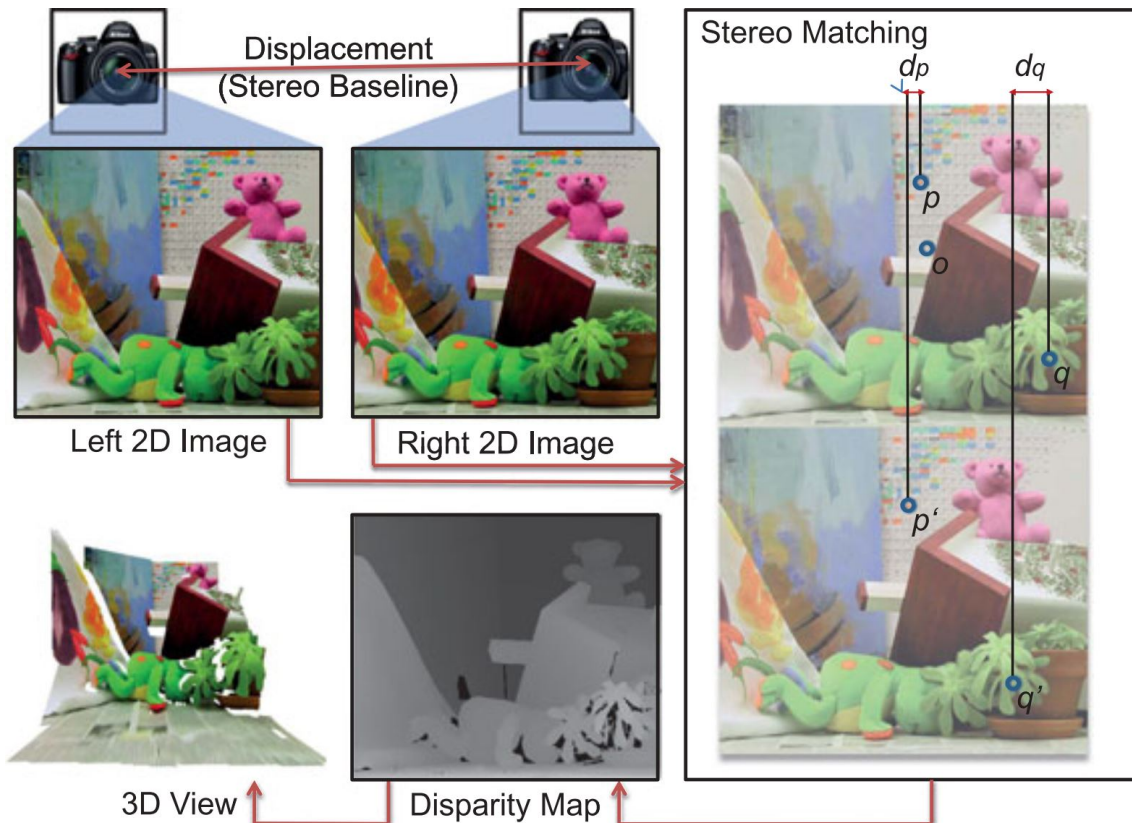


Fig. 2.1. Binocular stereo 3D reconstruction (Bleyer and Breiteneder, 2013). Stereo matching locates the corresponding pixels from two 2D images. The disparity is thus calculated as the relative displacement in between, from which a 3D view of the scene could be recovered.



Specifically, a disparity space image (DSI) is defined in which a probability is assigned to each element  $(x_0, y_0, d_0)$  within the disparity space, to represent the likelihood of a match between  $(x_0, y_0)$  in the reference frame and  $(x_0 - d_0, y_0)$  in the target frame. Dense stereo matching aims at estimating a surface within the DSI to best fit the geometric state of the object as displayed in Figure 2.2, either based on the lowest matching cost locally or additionally considering a spatial (piecewise) smoothness globally. However, a series of stereo rules should be assumed and obeyed to pursue the theoretically optimal disparity arrangement (Intille and Bobick, 1994). For each point from a frame, its corresponding point in the other frame should be located on a certain line and vice versa as shown in Figure 2.3. The line is termed as epipolar line ( $L_1$  and  $L_2$ ), which is the intersection of the image plane and the epipolar plane containing the object point (T) and the baseline ( $O_1O_2$ ). Thus, the correspondence search is simplified from 2D to 1D. Depending on the scene depth, the disparity is limited to a certain range. In addition to this hard constraint, there are several soft constraints in stereo matching, which should be followed in general but could be violated in special situations. Firstly, corresponding points should have similar pixel intensities as they originate from the same object point. However, this rule is not valid if different illumination or exposure conditions are applied on each image, or noise exists. Secondly, neighboring pixels should have the same or similar disparities to ensure the smoothness of the reconstructed surface, except for regions with depth discontinuities. Thirdly, each image point should have up to one corresponding point in the other image, with the exception that occlusions occur that lead to no matches. Besides, this rule cannot hold when several object points lie along the same projection axis. In Figure 2.3, the pixel  $q_1$  could be matched to any pixel among  $q_2$ ,  $p_2$  and  $t_2$ . Finally, points from a certain surface should be projected onto both image frames in the same order. In Figure 2.4 (a), the projections of Q, P and T on the left image  $q_1$ ,  $p_1$  and  $t_1$  have the same order as on the right image  $q_2$ ,  $p_2$  and  $t_2$ . This rule, nevertheless, cannot be applied on transparent targets as in Figure 2.4 (b) or violated by occlusions resulting in missing image points.

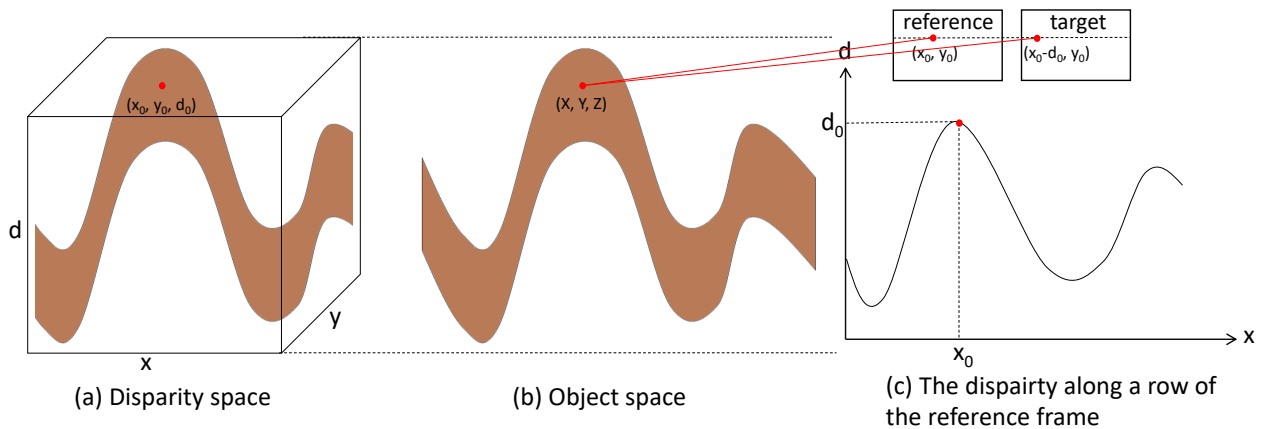


Fig. 2.2. Disparity determination through the disparity space for locating the correspondences. The element  $(x_0, y_0, d_0)$  in the disparity space should have the highest probability within the DSI, assuming  $(x_0, y_0)$  and  $(x_0 - d_0, y_0)$  are the corresponding projections from the object point  $(X, Y, Z)$  on the reference and target frame, respectively.

With the above stereo rules or constraints, diverse strategies have been designed to locate the dense correspondences between the stereo pair. For a better study of the stereo problem and to facilitate the algorithm design, the dense matching procedure is detached as a series of standard processing units (Scharstein and Szeliski, 2002), from which the overwhelming majority of stereo matching methods could find its own prototype.

### 2.1.1 Matching Cost Computation

The first step is to calculate the matching cost, which measures the photo consistency or similarity between potentially matched points to locate the corresponding pair. An intuitive

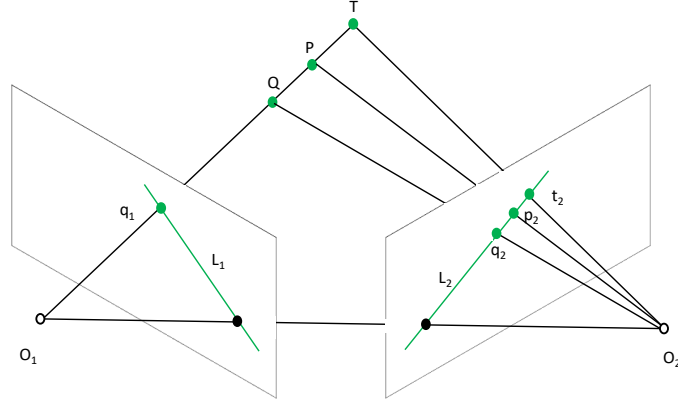


Fig. 2.3. Stereo epipolar geometry. The corresponding point for  $q_1$  could be located exclusively through the epipolar line  $L_2$  in another image. Thus, the stereo matching is highly simplified from 2D to 1D.

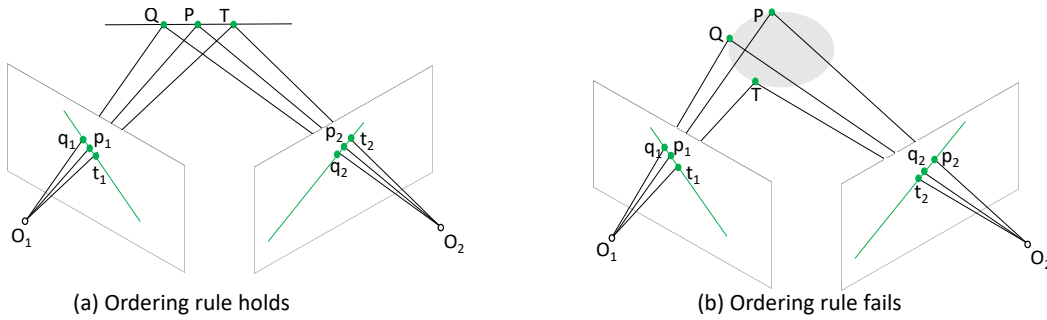


Fig. 2.4. The ordering rule in stereo matching. Normally, the order of the image points from the same targets should be consistent in each of the image, as (a). However, the rule is not valid for transparent objects, e.g. in (b).

measurement could be absolute intensity differences (AD) (Kanade et al., 1995), squared intensity differences (SD) (Hannah, 1974; Anandan, 1989; Matthies et al., 1989; Simoncelli et al., 1991), or normalized cross-correlation (Hannah, 1974; Ryan et al., 1980; Bolles, 1993). Computing the similarity metrics using the above standards is straightforward, nevertheless, the performance is drastically affected in practice by non-Lambertian reflectance, radiometric inconsistency, imaging noise, etc. A simple modification is to calculate the gradient of the intensity or color before comparing the pixels, or applying more advanced metrics such as mutual information (MI) (Viola and Wells, 1995) to compare the image entropy, Rank and Census transforms (Zabih and Woodfill, 1994) using robust non-parametric measures, etc. For example, Census models the intensity varieties within the neighborhood surrounding each pixel for comparison. The neighbors with lower intensities than the central pixel are assigned with a number of 0, otherwise with 1. A bit string is thus constructed for each pixel from the surrounding patch, to be compared with the string of other pixels for matching cost computation which is more insensitive to radiometric differences.

### 2.1.2 Cost (Support) Aggregation and Disparity Computation

For each pixel  $p$  from the reference frame, a naive strategy to find its corresponding point would be searching within the target frame pixel by pixel along the epipolar line, until the pixel resulting in the lowest matching is located, as Equation 2.2.

$$d_p = \underset{d_{\min} \leq d \leq d_{\max}}{\operatorname{argmin}} C(p, p-d). \quad (2.2)$$

In the above equation,  $C$  measures the matching cost between  $p$  in the reference frame and  $p-d$  in the target frame, under the disparity constraint as  $d \in [d_{\min}, d_{\max}]$ . This strategy is named as winner-take-all (WTA). However, the resultant disparity map could be very noisy,

since the information from a single pixel cannot exclusively locate its corresponding pixel from another image. Given the data ambiguity, matching outliers may occur as more than one pixel could result in a local minimum of the matching cost.

Hence, a stereo matching pipeline also extracts the neighboring information within a surrounding region for each pixel, to aggregate the cost and execute a region or window based comparison. It should be noted that a matching cost algorithm normally considers this spatial aggregation already, to explicitly include the neighboring pixels for the cost calculation as Equation 2.3, or propose a feature to represent the surrounding patch for further processing, e.g. Census.

$$d_p = \operatorname{argmin}_{d_{\min} \leq d \leq d_{\max}} \sum_{q \in N_p} C(q, q - d). \quad (2.3)$$

In the above equation, all the pixels  $q$  contained by the window  $N_p$  centered at  $p$ , as a support region, are taken into consideration for the cost aggregation. However in practice, it should be noted that the neighboring pixels cannot equally contribute to the central pixel to aggregate the matching cost, since they may originate from multiple depth planes leading to contradictive disparity determination. Regarding to this problem, a solution could be designed via two strategies. Firstly, setting an appropriate size and shape for the support region to include pixels from the same category, limiting pixels from other categories. A small surrounding window can limit the influence from uncorrelated points, nevertheless, resulting in a noisy depth estimation as the naive strategy in Equation 2.2. On the other hand, a large support region might lead to a blurred disparity map, with edge fattening or lost details. Hence, Fusiello et al. (1997) directly check nine square windows at different image locations, in order to avoid covering pixels from a different category through at least one window. Hirschmüller et al. (2002) cut the support window into 5 (or 9) sub-regions, with one of them at the center and the other 4 (or 8) sub-windows outside. At last, only the central sub-window and 2 (or 4) outer sub-windows with the best correlation scores are used for the cost aggregation. In Veksler (2002), the size and shape of the support window are simultaneously optimized using a large amount of compact windows, with a low ratio of perimeter to area.

Besides determining a suitable size or shape for the support window, it is also feasible to assign a weight to each neighboring pixel for the cost aggregation as:

$$d_p = \operatorname{argmin}_{d_{\min} \leq d \leq d_{\max}} \sum_{q \in N_p} \omega_q \cdot C(q, q - d), \quad (2.4)$$

in which  $\omega_q$  indicates the contribution from a neighboring pixel  $q$  to determine the central pixel's disparity estimation. Thus, the cost aggregation within a homogeneous region can be satisfied by adjusting the weight of each neighbor from the support window. The support weight of each neighboring pixel is defined according to its relationship with the center. An intuitive clue could be the pixel intensity (or color) difference and spatial distance (Yoon and Kweon, 2005). It is natural to believe that homogeneous pixels would share similar appearances and close coordinates within an image, as they belong to a common target from the scene. Hence,  $\omega_q$  is formulated as:

$$\omega_q = \exp\left(-\left(\frac{|I(p) - I(q)|}{\rho_I} + \frac{\|p - q\|}{\rho_D}\right)\right). \quad (2.5)$$

In Equation 2.5,  $|I(p) - I(q)|$  measures the intensity difference between the central pixel  $p$  and one of its neighbor pixels  $q$ , while  $\|p - q\|$  represents the Euclidean distance in between.  $\rho_I$  and  $\rho_D$  can be additionally defined by users to balance the two terms. From the equation, the

spatially more close neighboring points owning similar intensity values will acquire higher importance to determine the central pixel’s disparity.

Hosni et al. (2009) argue that the rule on the pixel intensity and spatial distance might be violated, if a neighboring point originates from another object on a different depth plane but with a similar appearance, e.g. the neighboring leaves in Figure 1.1. Therefore, they propose their geodesic stereo by selecting homogeneous neighbors via the spatial connectivity. Homogeneous neighboring pixels would be ensured, if a path, constituted by pixels with similar intensity or color, exists to connect the central and the neighboring pixel. Accordingly, the weight of each neighbor is calculated as:

$$\omega_q = \exp\left(-\left(\frac{\min_{G \in \mathcal{G}_{p,q}} I(G)}{\rho_G}\right)\right) \quad \text{with} \quad I(G) = \sum |I(q) - I(q')|. \quad (2.6)$$

In the above equation,  $G$  is one of the possible geodesic paths from  $\mathcal{G}_{p,q}$ , which connect the center  $p$  and the neighbor  $q$  within the support region in all possible cases.  $\rho_G$  is also a user-defined parameter for controlling the smoothness of the disparity map.  $I(G)$  computes the path cost, via summarizing the intensity differences between all the neighboring pixels  $q$  and  $q'$  along the path. The method is tested on the Middlebury data and proven to achieve a superior performance (Hosni et al., 2009).

Locating the corresponding pixel with the support from certain neighboring pixels is nominated as local stereo methods, since only one surrounding patch is considered for each pixel to determine the disparity. Based on the strategy to adaptively set the support window size and support neighboring weights, the local methods promote the stereo matching techniques and produce high quality 3D results on stereo benchmarks, e.g. Middlebury (Scharstein and Szeliski, 2002). However, local methods might achieve very poor results in practice, e.g. for large textureless area, due to that a support window can only provide local and regional information. In addition, occlusions which are hard to process in local methods, have to be handled in the post-processing. Therefore, another category of stereo methods, global methods, are proposed to globally sense the scene and estimate the disparity. Finding a disparity value leading to low matching cost is only part of the optimization target in global methods, besides, the spatially neighboring pixels should also acquire similar disparity predictions in order to achieve the spatial smoothness. The two targets are named as data term and smoothness term, respectively, which are jointly considered in global stereo methods so that a global cost or energy minimization is obtained from all the pixels. Local methods essentially also consider the smoothness term implicitly within the support window, via assuming a common disparity value shared among the neighbors. Global methods, nevertheless, emphasize the smoothness explicitly in the optimization function as:

$$E(D) = \sum_{p \in I} C(p, D(p)) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} S(D(p), D(q)), \quad (2.7)$$

in which  $D$  denotes a disparity map for the reference frame  $I$ . The first term on the right side of the equation is the data term, which calculates the accumulative matching cost of all the pixels  $p$ , assuming  $D$  as the disparity results. The second term, smoothness term, penalizes (increases) the energy function if the smoothness requirement is disrupted by a pairwise neighboring pixels  $\{p, q\}$ . Similar to the scheme of support weights setting in local methods, the penalization can be adjusted according to the difference between the neighboring points’ intensity. Thus, the optimization is capable of encouraging varying disparities over potentially heterogeneous pixels, rather than an overall smoothing leading to blurred results.  $\lambda$  adjusts the influence of the smoothness terms. With  $\lambda = 0$ , the optimization degenerates to local methods.

A global method thus aims at computing a disparity map  $D$ , for which Equation 2.7 will find its minimum. In practice, the optimization considering the smoothness term in 2D is an np-complete problem (Boykov et al., 2001), since that the disparity determination of every pixel will affect every other pixel causing the "Knock-on" effect. A disparity map optimally satisfying Equation 2.7 cannot be solved in polynomial time, however, there are different strategies to approximate the global energy minimization.

Graph cuts (Boykov et al., 2001) is a commonly used technique to approximate the energy minimization. A graph is constructed with its nodes and edges formed by the image pixels and the corresponding relationship among them. In stereo matching, the optimization starts from an initial disparity map, which could be randomly initiated, and then iteratively converges towards the global minimum via a sequence of moves. Each move adjusts the disparity value of each pixel and has to result in a reduced or at least unchanged energy. By computing the minimum cut or maximum flow within the graph, the moves corresponding to the largest energy reduction can be found. Graph cuts achieves great performance in optimization problem, nevertheless, the calculation is time-consuming. Geman and Geman (1984) attempt to protect the disparity discontinuities via a Markov Random Fields (MRFs) based energy function. The global energy minimization is pursued via Maximum a Posterior estimate of the image. The global optimization can also be approached based on Belief Propagation (BP) (Felzenszwalb and Huttenlocher, 2004), which passes messages between neighboring pixels to express a pixel's belief for its neighbor's disparity assignment.

The np-complete problem in the general form of global methods (Equation 2.7) is caused by mutual influence among pixels for the respective smoothness requirement. Hence, an intuitive solution to approximate the energy minimization would be simplifying the optimization from 2D to 1D, by only considering neighbors along a directed 1D path as a scanline optimization task (Scharstein and Szeliski, 2002). Thus, each pixel can be processed independently through the scanline it belongs to for the smoothness term. However, this strategy also leads to an independent disparity estimation line by line, which breaks the 2D spatial relationship and causes the streaking problem.

Since that 1D scanline optimization is computationally possible, researchers attempt to apply it in multiple directions, which doesn't increase the theoretical complexity but better approximates the 2D smoothness expected by global methods. This idea led to the development of Semi-Global Matching (SGM) (Hirschmüller, 2008), a milestone of stereo matching algorithms. The algorithm constructs the energy function in form of global stereo methods containing both data and smoothness term, however, summarizes the energy along multiple 1D scanlines with different directions, typically 8 or 16 canonical scanlines in horizontal, vertical or diagonal direction, to determine the disparity results according to the minimum summarized energy as WTA in local stereo methods. Hence, a good compromise is obtained between accuracy and efficiency.

The schematic visualization of SGM is shown in Figure 2.5. For each pixel  $p$ , the disparity is determined according to the corresponding matching cost together with the disparity result from its previous neighbor along a scanline. Thus, 1D smoothness is satisfied via scanning through the path, pixel by pixel, from the image border to the target point. With multiple 1D scanlines visited, the 2D smoothness is approximated. In Equation 2.8, the energy of traversing a scanline in direction  $r$  for a pixel at image location  $p$  is defined.

$$\begin{aligned} L_r(p, d) = C(p, d) + \min ( L_r(p - r, d), L_r(p - r, d - 1) + P_1, \\ L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2 ). \end{aligned} \quad (2.8)$$

$L_r(p, d)$  denotes the energy for a disparity candidate  $d$ .  $C(p, d)$  is the data term under the current parallax, which could be computed using different matching cost algorithms, e.g.

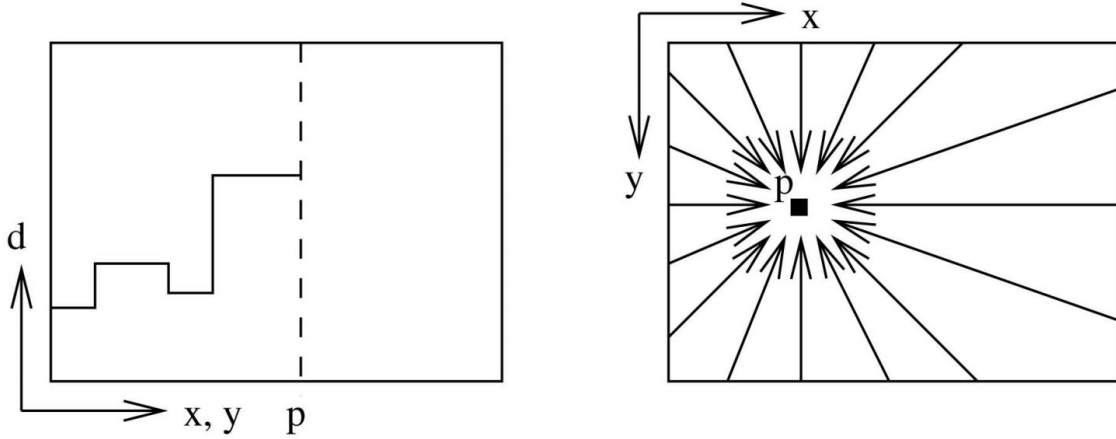


Fig. 2.5. The cost aggregation along a single scanline and the strategy in SGM to visit each pixel through multiple scanlines, assuming 16 scanlines are used (Hirschmüller, 2008). From the image border, the disparity is continuously predicted for each pixel to support the next pixel’s estimation along a directed path for smoothness. With the same procedure repeated along multiple scanlines, the 2D smoothness is approximated for each pixel.

Census or MI, and adjusted according to the specific scenario. The rest of the equation represents the smoothness term. With different options to aggregate the previous pixel  $p - r$ , the minimum energy is calculated from either a consistent disparity estimation, or a conflicting disparity selection with an additional penalty term for smoothness. Depending on the magnitude of the disparity inconsistency,  $P_1$  or  $P_2$  is applied for punishing only 1 or larger disparity difference, respectively ( $P_2 > P_1$ ). By summarizing the energy via multiple 1D scanlines, the disparity corresponding to the minimum summarized energy will be the final result, using the WTA strategy.

$$d_p = \underset{d_{\min} \leq d \leq d_{\max}}{\operatorname{argmin}} S(p, d) \quad \text{with} \quad S(p, d) = \sum_r L_r(p, d). \quad (2.9)$$

SGM is widely applied in computer vision and photogrammetry, thanks to its high robustness and its good accuracy-efficiency balance. However, there is still space to improve the algorithm. For example, the penalty terms applied in Equation 2.8 prefer fronto-parallel surface, which is not always present in the scene and might lead to a biased estimation. Therefore in Banz et al. (2012), an adaptive strategy to adjust the penalty functions is suggested, according to the gradients of the image intensities. Facciolo et al. (2015) construct a more compact 2D scanline aggregation, by additionally using the 1D scanlines visited already. Rothmel (2017) proposes tSGM, which uses a dynamic search range of disparity based on a pyramid architecture, achieving higher efficiency. Lu et al. (2021) design an efficient architecture for real-time SGM implementation.

### 2.1.3 Post-Processing

The matching cost computation provides an initial dissimilarity measurement between potentially matched pixels under a disparity candidate. Then, before further disparity estimation, neighboring pixels are included in the calculation so that the context around the target pixel is perceived to avoid mismatch due to ambiguous pixels. In addition, the spatial smoothness is guaranteed by limiting the disparity differences between neighboring points. The obtained disparity map, however, may still need refinement via a series of post-processing steps (Scharstein and Szeliski, 2002).

Firstly, the disparity prediction is essentially a selection from a set of pre-defined candidate values, which satisfies the WTA principle leading to the lowest matching cost, or globally achieves the best balance between matching cost and depth consistency. Thus, the corresponding disparity map is discretized. In order to obtain continuous depth values with sub-

pixel accuracy, a quadratic curve can be used to fit the cost values with respect to certain disparity candidates, at least 3, and locate a more precise disparity value at the curve's bottom.

Besides, occlusions could happen when only one of the stereo images captures the target point. The occluded pixels can be detected via a left-right consistency check (Cochran and Medioni, 1992; Fua, 1993). A cross-check between the disparity results regarding the left and right image as the reference frame, respectively, would indicate the occluded regions with inconsistent estimation. Hence, the disparity prediction of the corresponding regions is removed. An interpolation using neighboring pixels with valid disparity estimation can be implemented to compensate these "disparity holes". In addition, median or bilateral filters are also used as post-processing to correct noisy estimation, e.g. caused by mismatches.

## 2.2 Machine/Deep Learning Assisted Stereo

As the development of machine and deep learning (Mitchell, 1997; Goodfellow et al., 2016), the dense matching technique has boosted its performance with a better understanding of the data, a deeper representation of the features, a improved parameterization of the optimization function under a trainable framework, etc. The learning capability can be applied to stereo matching in two forms: replacing a conventional matching step with a supervised module as Figure 2.6 or pure learning based end-to-end architecture as Figure 2.7 (see details in Chapter 3).

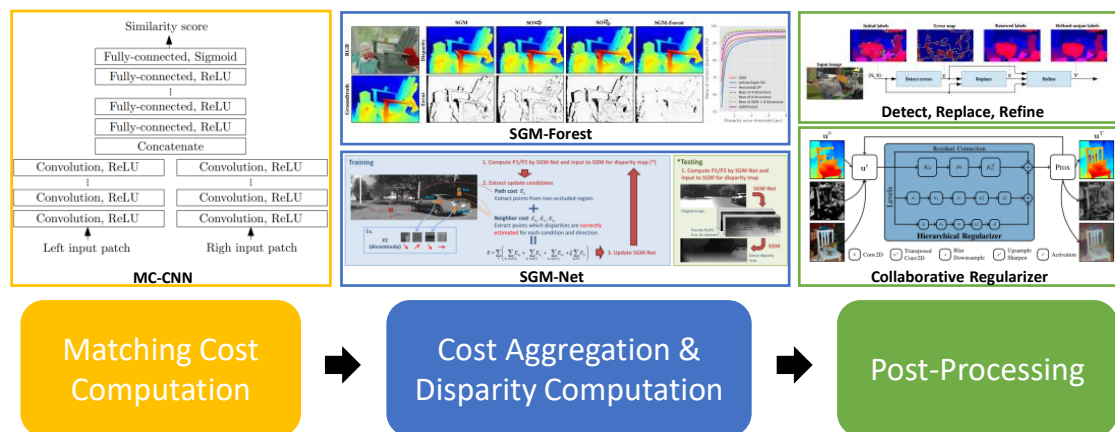


Fig. 2.6. Deep learning assisted stereo matching (Zbontar and LeCun, 2016; Gidaris and Komodakis, 2017; Seki and Pollefeys, 2017; Schönberger et al., 2018; Knöbelreiter and Pock, 2019). Along the conventional processing pipeline, each module can be supervised for better feature representation to calculate the matching cost, smarter strategy to aggregate neighboring pixels and penalize disparity inconsistency, and learning based post-processing to refine the results.

In the first category, certain stereo matching steps can be substituted by a learning unit and integrated into a conventional algorithm. For example in Zbontar and LeCun (2016), the pixel comparison is built on features extracted by a network for matching cost computation. Thus, a trained model is capable of automatically locating an appropriate surrounding region of the target pixel for similarity measurement and providing high level features to calculate the matching cost. Afterwards, SGM is applied for cost aggregation and disparity estimation. A very important advantage for this hybrid of conventional wisdom and deep learning is that less data are needed to train a well-performed model, thanks to the relatively simple structure focusing on a certain sub-task. For the same reason, overfitting is less likely to happen, e.g. in Xia et al. (2018) the pre-trained MC-CNN model on Middlebury dataset is directly used for plant stereo reconstruction.

Recently, state-of-the-art deep learning techniques (He et al., 2016; Chen et al., 2018) and

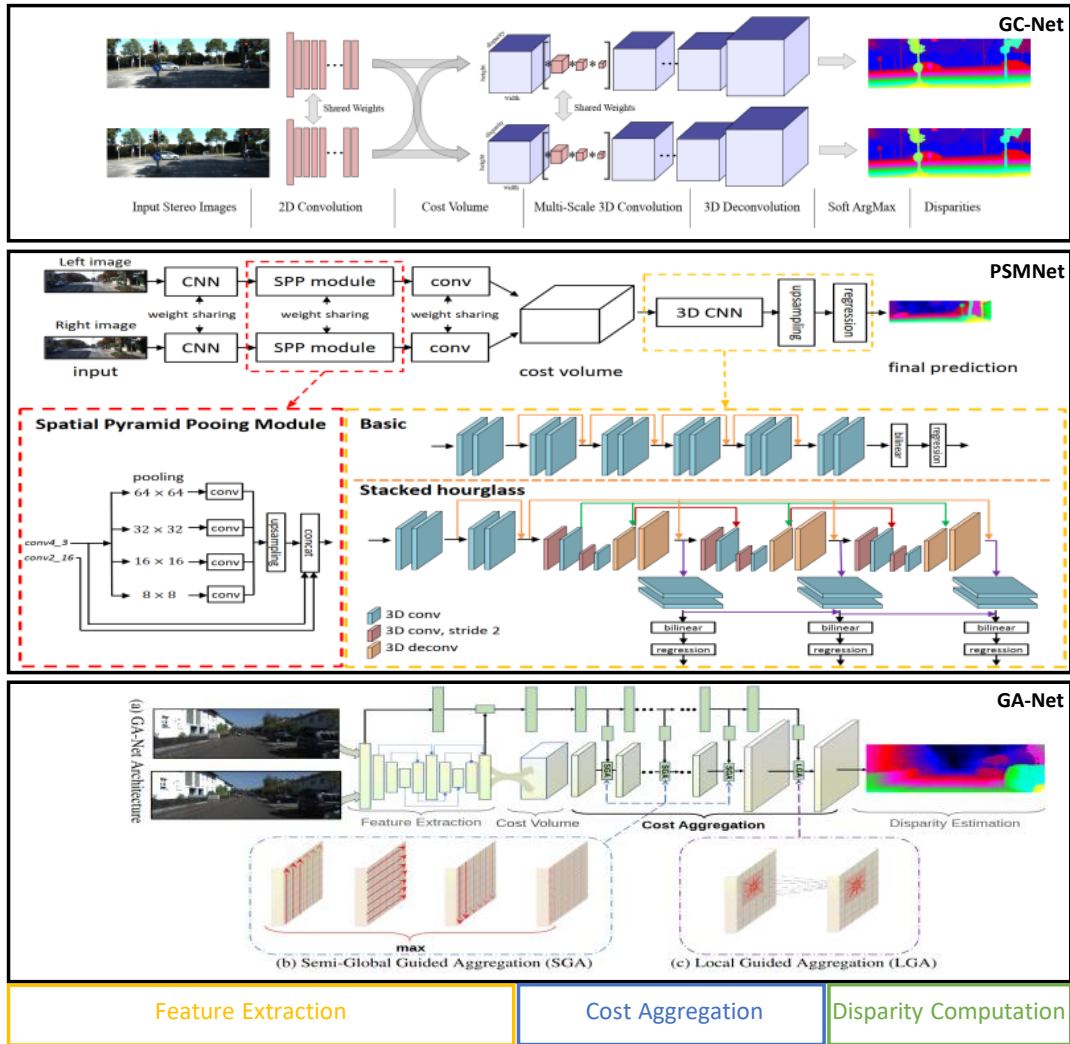


Fig. 2.7. Representative end-to-end neural networks for stereo matching (Kendall et al., 2017; Chang and Chen, 2018; Zhang et al., 2019). The entire stereo matching procedure is fully differentiable and trainable, from feature extraction, cost volume generation and regularization, to disparity prediction at the end. Thus, a disparity map can be output directly from a stereo pair.

abundant well-annotated training data (Mayer et al., 2016) have built a solid basis to construct a full end-to-end neural network for stereo matching. From an input stereo pair, a network can be supervised to predict the disparity values directly. In Figure 2.7, we list some representative end-to-end algorithms, including GC-Net (Kendall et al., 2017), PSM-Net (Chang and Chen, 2018), GA-Net (Zhang et al., 2019). These neural networks mainly include a feature extraction module, a cost regularization module and a disparity regression module. From each stereo pair, the features are learned and compared to generate a cost volume. The cost can roughly indicate the probability of each disparity candidate to be correct, for which an encoder-decoder is usually exploited to further regularize the cost volume so that a larger receptive field is obtained and the spatial relationship within the neighborhood is utilized. Thus, the disparity is calculated as the weighted summation of each disparity candidate according to its corresponding probability. It should be mentioned that the post-processing can also be included within an end-to-end architecture, such as Pang et al. (2017), with a disparity map predicted and refined together.



## 2.3 Monocular and Multi-View Stereo

Binocular stereo matching simulates human eyes to perceive the scene depth with two cameras pointing towards the object, from which a stereo image pair is acquired and a disparity map indicating the displacement of corresponding pixels is calculated to represent the depth. However, the stereo input could vary when only one or multiple images of the scene are available, depending on the specific imaging situation in practice as shown in Figure 2.8. Accordingly, the technique of monocular and multi-view stereo matching is developed.

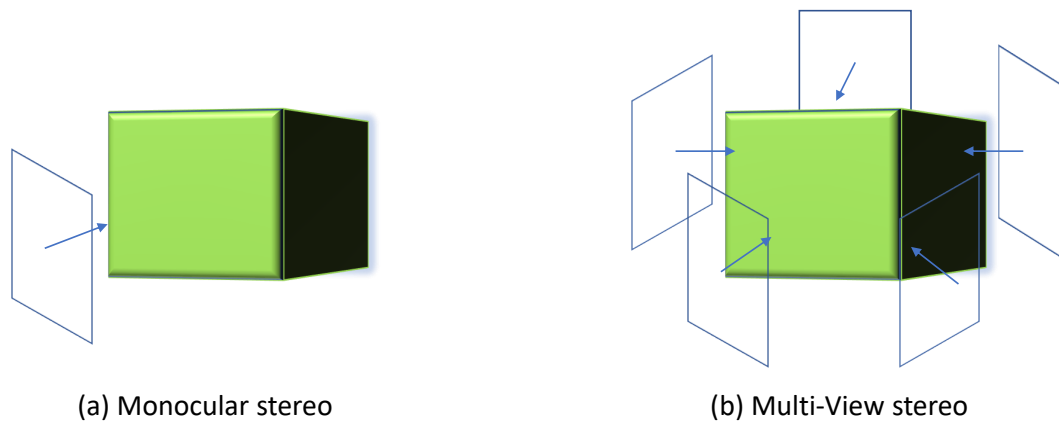


Fig. 2.8. Monocular and multi-view stereo acquisition. In monocular stereo, only a certain view of the scene is obtained, which makes the depth sensing an ill-posed problem. In multi-view stereo, a sequence of images (at least two) are captured around the target scene or object. Detailed 3D reconstruction can be achieved with fewer occlusions and reduced stereo errors thanks to additional views.

Monocular stereo matching is theoretically an ill-posed problem, since that the geometric setup to triangulate the target is broken with one of the stereo images removed. As displayed in Figure 2.3, the image point  $q_1$  might correspond to any of the object points  $Q$ ,  $P$  and  $T$  along the optical axis, without the support from the second frame. Monocular depth inference is traditionally realized with the aid of active depth sensors, such as RGB-D cameras and LiDAR scanners (Zhao et al., 2020). The practical use of these sensors, nevertheless, can be harmed due to complex environmental factors such as varying illumination, large depth ranges as in aerial and satellite altimetry, or dense depth estimation requirement, etc. Hence, current researches focus on solving the problem via a data-driven strategy (Garg et al., 2016; Fu et al., 2018; Chakravarty et al., 2019; Facil et al., 2019; Wang et al., 2019). A network can be trained to perceive the monocular depth according to the scene logic, structure information, relative positions and orientations among objects, etc. (Hoiem et al., 2005). The supervision clues could be obtained either from the differences between the predicted and ground truth depth values, or the geometric constraints on the camera pose when the ground truth is not available or expensive to collect (Zhao et al., 2020). In addition, multiple sub-tasks can be constructed to assist the depth prediction according to the relationship among the tasks, such as the joint estimation of monocular depth, optical flow and ego-motion in Yin and Shi (2018).

Given that multiple images of a certain scene or object, it is natural to believe that a higher quality of 3D reconstruction can be achieved with additional knowledge acquired from extra views. Besides, occlusions are less likely to happen. Thus, multi-view stereo matching is proposed to build a more complete 3D model with better details. Regarding one of the stereo images as the reference frame, an intuitive scheme would be computing the matching cost of a reference pixel with respect to the pixels from the other images according to each depth candidate (Okutomi and Kanade, 1991). The final prediction can be the depth corresponding to the minimum sum of matching cost scores. Also, a series of depth planes can be defined, on which each input image is projected. The optimal depth is thus the one

leading to the most harmony stack of projections (Collins, 1996), e.g. acquiring the lowest pixel value variance (Yang and Pollefeys, 2003). In order to better organize and study multi-view stereo, Seitz et al. (2006) extract the core properties of different algorithms to construct a guideline. Firstly, a format should be decided to represent the multi-view reconstruction, e.g. via voxels, meshes, level-sets, or simply depth maps as binocular stereo. Afterwards, an optimization target is defined to check the quality of the resultant 3D models, such as photo-consistency and silhouette-consistency (Kutulakos and Seitz, 1999). The visibility is modelled to consider certain views for measuring the consistency. Similar to binocular dense matching, some prior requirements are imposed on the shape of the reconstructed models, e.g. minimal surfaces preference (Tasdizen and Whitaker, 2004), together with the consistency check. Thus, a more precise geometry can be recovered even for low-textured regions. At last, an optimization strategy should be designed to minimize the cost/energy function, so that an optimal surface is extracted (Seitz and Dyer, 1997; Kolmogorov and Zabih, 2002; Vogiatzis et al., 2005) or iteratively fitting to the constructed 3D volume (Kutulakos and Seitz, 1999; Slabaugh et al., 2004). It should be mentioned that, many algorithms enforce some pre-requirements on the geometry of the objects or scenes for a better reconstruction, e.g. a bounding box (Kutulakos and Seitz, 1999; Slabaugh et al., 2004), foreground and background segmentation (Vogiatzis et al., 2005) or limited depth ranges (Kolmogorov and Zabih, 2002; Zitnick et al., 2004), etc.

## 3 State-of-the-Art in Learning based Dense Matching

This chapter reviews the state-of-the-art dense matching techniques. As stated in Chapter 2, SGM provides a balance between the quality of the reconstructed stereo images and the computational complexity, so this algorithm (and the corresponding variants) is used in numerous scenarios. However, with the continued introduction of machine learning and deep learning into the computer vision field, much higher performance is being achieved than was the case a few years ago. The combination of classical algorithms with deep learning based pixel comparison lead to the first breakthrough (Zbontar and LeCun, 2016). In the last few years, learning based stereo methods evolved from replacing parts of the traditional matching pipeline with learning based methods to end-to end networks. The data driven end-to-end approach enables the networks to learn the processing strategy and automatically adapt to the training data. Hence, this chapter mainly focuses on machine/deep learning based methods. Finally, several benchmark datasets for training and testing the AI-based dense matching algorithms are presented.

### 3.1 Enhancement of existing algorithms through Machine/Deep Learning

Early use of machine and deep learning still follows the traditional dense matching pipeline, in which only certain individual steps (as described in Chapter 2) are replaced by a learning module. With the aid of a supervised processing unit, the stereo information flow is better transferred leading to higher reconstruction quality. The subsequent sections follow the traditional pipeline and introduce the state-of-the-art learning based modifications.

#### 3.1.1 Learning based Matching Cost

Through the pipeline, the similarity measurement in pixel- or patch-level is the first step to determine the correspondences between the stereo pair. A good similarity measurement provides the basis to estimate the matching cost, for following cost optimization and disparity computation. As the method only targets on extracting the feature of the patches for comparison, a simple network architecture is often applied, as shown in Figure 3.1.

Since in stereo matching, each input image or patch of the pair is processed using the feature extraction method to allow a uniform representation and feature comparison. Thus, a Siamese structure (Bromley et al., 1993) is appropriate, which is constructed by sub-networks containing the same layer composition, to symmetrically extract feature from patches for following comparison. At the end, a similarity score is calculated and supervised by the prepared labels. The hinge loss and the cross-entropy loss are normally used (Zbontar and LeCun, 2015; Luo et al., 2016; Zbontar and LeCun, 2016; Shaked and Wolf, 2017). The former expects a larger similarity of a positive match than a negative match that appears in pair by a certain margin, while the latter requires higher/lower similarity score of the positive/negative matching example.

Zbontar and LeCun (2016) propose a representative model for early CNN based matching cost algorithms, MC-CNN. Up to five convolutional layers, with each of them followed by a rectified linear unit, constitute the feature extraction module. Afterwards, the outputs of each sub-network are transmitted to a comparison sub-network for the similarity score calculation. Regarding the comparison module, a lightweight structure is designed for a

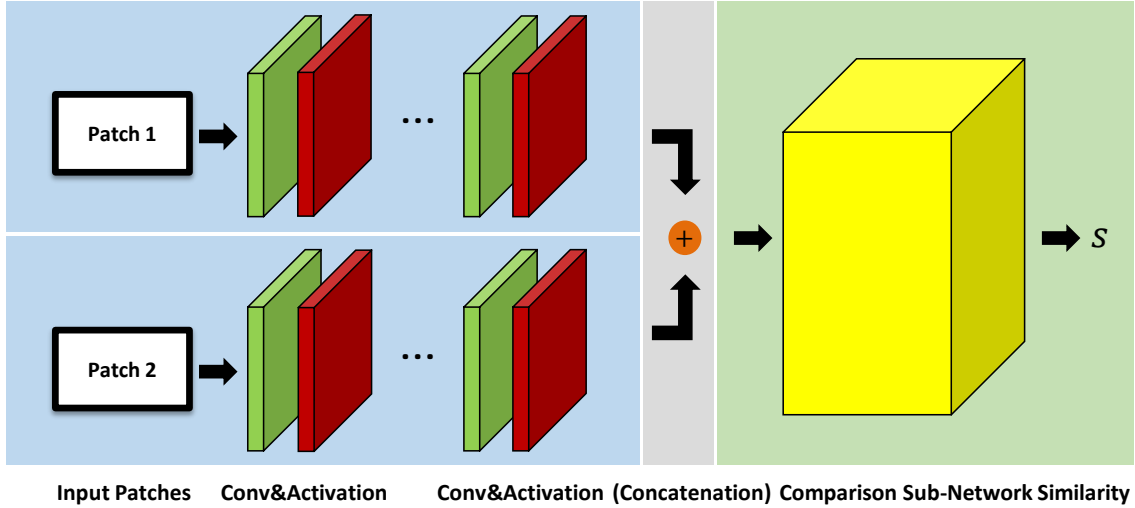


Fig. 3.1. Basic network architecture for matching cost computation. A Siamese network is usually used as a backbone, with two branches sharing the same architecture and weights to extract features from each of the patches. Afterwards, the feature can be directly compared for matching cost calculation, e.g. via a correlation layer, or concatenated together to feed a comparison sub-network to adaptively learn a similarity measure.

faster prediction, which uses only the dot product of the normalized feature vectors as the similarity score. Besides, the feature vectors can also be concatenated and passed through a sequence of fully-connected layers (as a trainable unit) to adaptively learn a similarity measure, leading to another architecture with higher accuracy. Meanwhile, other matching cost methods are also proposed, in which a deeper module is built based on ResNet for a better performance. The feature extraction and comparison units are merged to directly output a similarity score for each patch, etc. (Han et al., 2015; Zagoruyko and Komodakis, 2015; He et al., 2016; Shaked and Wolf, 2017).

Luo et al. (2016) adopt the same idea. Instead of training a model on a series of patch pairs to measure the degree of similarity, they regard the stereo matching problem as a multi-class classification task. Specifically, they aim at directly executing an overall comparison among all the disparity candidates and finding the best one, rather than determining the similarity between patches with certain displacements in sequence. Hence, the forward process is applied only once for each point to process all the disparities, thus achieving better efficiency.

Shaked and Wolf (2017) attempt to deepen the feature extraction module of the matching cost network, in order to unravel more profound information when comparing pixels and patches. However, it is shown in (Zbontar and LeCun, 2016) that simply adding more layers is not capable of enhancing the performance. Therefore, they resort to residual blocks with skip connection to enable deeper networks. Through their experiments, unfortunately, stacked residual blocks increase the difficulty level to make the training converge, while not achieving improvement. Hence, a weighted skip connection is introduced as:

$$Y = F(X) + \mu \cdot X. \quad (3.1)$$

Compared with the normal skip connection,  $\mu$  is additionally defined to determine the contribution of the input, so that the stacked residual blocks can adaptively adjust the influence of the shallower or deeper feature representation. As for the comparison sub-network, they design a hybrid measurement using a simple dot product of the feature vectors, and a more complex similarity learning through a fully connected network, which is basically a combination of the fast and accurate architecture of MC-CNN.

Zhang and Wah (2018) focus on the two most important factors, consistency and distinctiveness, for matching cost computation. The former expects a constant representation of a point from different views, while the latter emphasizes the discrepancy for non-matched

pixels. Therefore, they build up a multi-objective frame to simultaneously optimize the two targets based on the principle of Pareto optimality (Zitzler and Thiele, 1999). The Pareto optimality seeks an optimal resource allocation state, so that no reallocation could be made to improve any objective without influencing others. Good features could be extracted on the Pareto frontier for matching cost calculation.

Traditional local methods suffer from selecting a suitable window size, for which a small window cannot include enough context, while a large window may blur the details. Accordingly, Chen et al. (2015) propose a deep embedding model to extract multi-scale features. They design a "double" Siamese network, which processes the original and a downsampled image, respectively. Thus, the corresponding feature contains multi-scale information, leading to a more reasonable expression for untextured regions, depth discontinuities, etc. Finally a similarity score is estimated for each scale, which are then merged based on an ensemble voting system as the result. Similarly, Zagoruyko and Komodakis (2015) also exploit a multi-scale feature extraction, in order to combine the feature of each scale to determine the similarity measure.

Ye et al. (2017) apply multi-size and multi-layer pooling to obtain a feature pyramid across multiple scales. Thus, a variant receptive field is acquired with both rich context contained, and local details preserved. Park and Lee (2017) also leverage spatial pyramid pooling layers, so that their proposed matcher is able to see a wider neighborhood around each point with more hints to search the corresponding point. However, they use the pyramid pooling in the comparison sub-network (see Figure 3.1), causing multiple computations of the pooling operations for each pixel, depending on the number of disparity candidates.

Moreover, Schuster et al. (2019) attempt to enlarge the receptive field via dilated convolutions and propose their stacked dilated convolutions (SDC) module. The definition of dilated convolution originates from signal processing (Holschneider et al., 1990; Chen et al., 2018), for which holes (zeros) are inserted into the convolutional kernels in order to increase the receptive field. SDC stacks multiple modules, each containing several parallel dilated convolutional layers with different dilation rates. Thus, multi-scale features can also be generated. Besides, with the same kernel and shared weights among dilated convolutional layers in parallel, the network is highly efficient. Similar strategies are adopted in (Fu et al., 2018).

Instead of designing a purely learning based algorithm, Batsos et al. (2018) resort to classical methods and propose their "Coalesced Bidirectional Matching Volume (CBMV)". They train a random forest (Breiman, 2001) to collect all the important evidence from four conventional matchers, normalized cross correlation (NCC), census (Zabih and Woodfill, 1994), zero-mean sum of absolute differences (ZSAD) on intensities and SAD on filtered intensities via a horizontal Sobel operator. As the matching cost for a certain disparity hypothesis may differ, when the left or right image is regarded as the reference frame, the feature vector for feeding the random forest includes bidirectional similarity or likelihood from the basic matchers, with both left-to-right and right-to-left matching considered. The random forest predicts the confidence of each disparity candidate resulting in a robust matching volume, that is invariant to affine intensity transformations, camera gain or bias, image sampling etc.

The above methods require supervision by labeled data, which is not always available or costly to collect. Hence, Tulyakov et al. (2017) argue that task-specific constraints could be used to apply a weak supervision on the model, with no need of a well annotated dataset. Furthermore, even when training on labeled data, the constraints are able to provide extra knowledge to make robust feature metrics. As shown in Figure 3.2, five terms are considered, namely epipolar constraint, disparity range constraint, matching uniqueness constraint, smoothness constraint and ordering constraint.

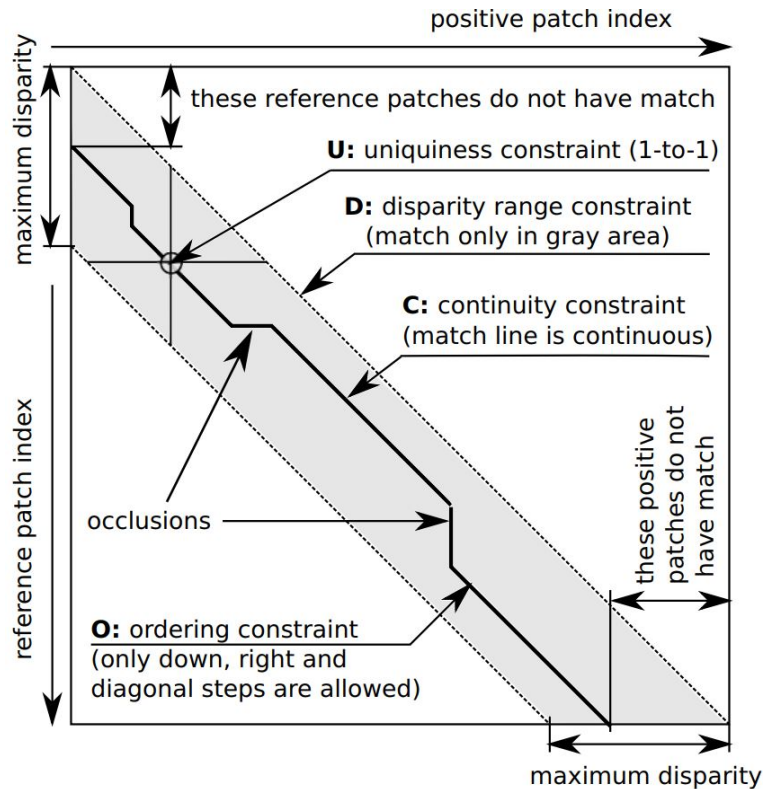


Fig. 3.2. Matching constraints along an epipolar line (Tulyakov et al., 2017), including epipolar constraint, disparity range constraint, uniqueness constraint, continuity constraint and ordering constraint.

According to the figure, correspondences should be searched along the epipolar line within a certain disparity interval. The best match should surpass all the other alternatives. Moreover, neighboring pixels should have consistent disparities and the order of the points on the reference frame should keep the same on the matching frame. Based on the above constraints, the corresponding weakly trained matching cost measurement can achieve comparable performance with a supervised model.

As the matching cost network mainly targets at image patches to measure the similarity, a small and sparse dataset can already provide enough training samples, e.g. in (Zbontar and LeCun, 2016) 25 million, 17 million and 38 million training examples could be extracted from KITTI-2012 (Geiger et al., 2012), KITTI-2015 (Menze and Geiger, 2015) and Middlebury (Scharstein and Szeliski, 2002; Scharstein and Szeliski, 2003; Scharstein and Pal, 2007; Hirschmuller and Scharstein, 2007; Scharstein et al., 2014) dataset, respectively.

### 3.1.2 Learning based Cost Aggregation and Regularization

With the above learning based similarity measurements, an initial matching cost is obtained, which is essentially still a local region based comparison, despite that certain techniques are applied to extend the receptive field. This initial cost cube could be noisy, especially for occlusions, low-texture or non-texture areas, and repetitive patterns. Therefore, the cost volume should be optimized, in order to carry out a global regularization to better guide the disparity estimation.

SGM is typically used to regularize the cost volume, e.g. in (Chen et al., 2015; Luo et al., 2016; Zbontar and LeCun, 2016), as it can provide a good trade-off between accuracy and efficiency. However, the method contains manual setting of penalty terms and empirically accumulates the cost of all the scanlines before WTA, which cannot satisfy different scene structures. Hence, some researchers attempt to determine a smart scheme with the aid of

machine/deep learning techniques (Michael et al., 2013; Park and Yoon, 2015; Poggi and Mattoccia, 2016; Seki and Pollefeys, 2016; Seki and Pollefeys, 2017; Schönberger et al., 2018; Xia et al., 2020).

Park and Yoon (2015) explore a strategy to modulate the data term in SGM and global methods, so that the influence of unreliable pixels is confined. Hence, they train a random forest to select the most trustworthy measures from twenty-two basic matching costs, including image gradients, matching scores, left-right differences, etc. Afterwards another random forest is trained to estimate the confidence of each pixel, according to the selected measures. Based on the confidence, reliable pixels keep a similar matching cost as before, while the other pixels with low confidence will be flattened, so that the following disparity estimation is dominated by confident pixels, leading to more robust results even in challenging regions.

SGM tends to avoid inconsistent disparities among neighboring points, via penalizing the current disparity candidate that is different from the previous pixel's estimation. This fronto-parallel smoothness assumption causes inappropriate estimation for slanted plane, particularly in untextured regions. Scharstein et al. (2017) propose their SGM-P to alleviate the problem. They compute the surface orientation priors based on a fast stereo matching on downsampled image, so that the determination of the penalty terms could be guided by the surface shape and adjusted according to the specific situation. The surface's geometry could also be perceived by drawing a surface normal map via Manhattan-world priors (Lee et al., 2009). Compared with the pure SGM, SGM-P achieves a noticeable improvement towards the theoretical upper bound of the performance, for which the surface orientation priors is directly acquired from the ground truth. SGM-P is essentially not a supervised algorithm, however, the surface orientation is promising as an extra information source to be integrated into other learning based strategies for more reasonable depth estimation.

Similarly, SGM-Net proposed by (Seki and Pollefeys, 2017) is the first neural network based SGM optimization. A normalized image patch and its position are fed to the network, in order to predict the penalty terms for each pixel along different scanline directions. The network is trained with an aim to minimize two costs, path cost and neighbor cost. The former searches a path, along which the traversing cost should be the smallest, while the other one requires that the correct disparity pass through two consecutive pixels must have smaller cost than any other options. Different situations are considered, including depth border, flat and slanted plane. Moreover, they design a signed parameterization to differentiate positive and negative disparity change, e.g. the disparity is more likely to increase along a road towards the camera. With the aid of CNNs, SGM-Net outperforms the basic SGM with handcrafted penalties. SGM-Net is a more complete and automatic version of their previous work (Seki and Pollefeys, 2016), which simply exploits a CNN to estimate a confidence for manually adjusting the penalty terms.

After the cost aggregation along the scanline, a smart strategy is expected to fuse each scanline's estimation more reasonably instead of a simple summation. For this purpose, Michael et al. (2013) assign a weight to each scanline for deriving a weighted summation. They are inspired by the fact that the disparity map estimated by each single scanline presents varying qualities, depending upon the specific global structure of the scene. Due to the favor on fronto-parallel smoothness, the scanline direction may influence the disparity prediction. In Figure 3.3, the raw estimation of the "left to right" scanline is better than the "top to bottom" one (see the roof and the truck marked by the red rectangle), because only the former satisfies the fronto-parallel smoothness. Therefore, they apply the covariance matrix adaptation evolution strategy (CMA-ES) (Beyer, 2007) to automatically predict a weight and a couple of penalty terms for each scanline. The algorithm accomplishes better accuracy without hurting the efficiency.

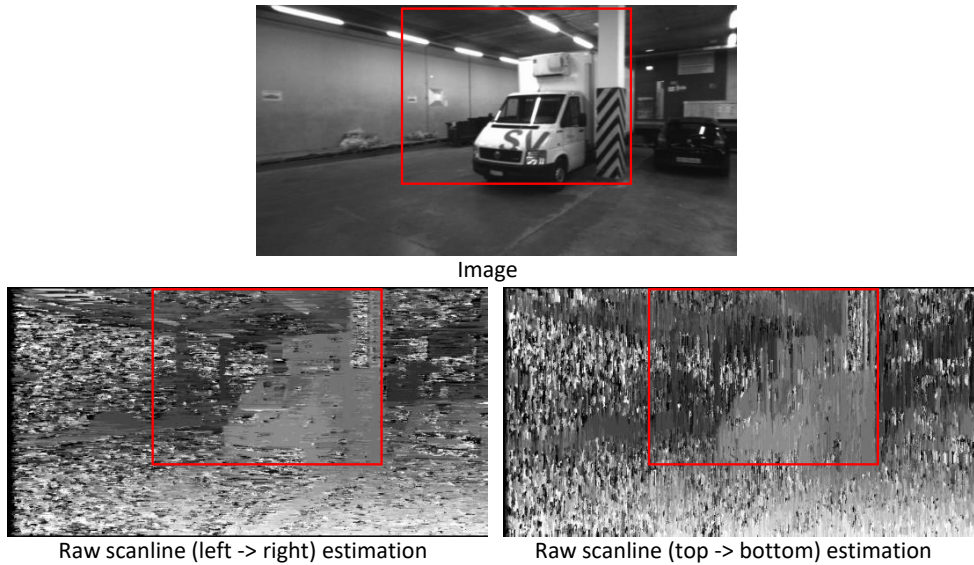


Fig. 3.3. The disparity estimation using a single scanline. Although the raw estimation from each single scanline is noisy, it can be found that the results from the "left to right" scanline is slightly better than "top to bottom", within the marked region. The reason is that the depth distribution is more stable horizontally than vertically.

Poggi and Mattocchia (2016) also focus on applying a weighted sum of the cost along each scanline. A disparity map is firstly estimated using each single scanline, from which a feature vector is extracted for each pixel and the corresponding surrounding patch to represent the statistical dispersion of the depth. Five metrics constitute the feature, including the disparity agreement, disparity scattering, median disparity, variance of the disparity and median deviation of the disparity. The disparity agreement is quantized by counting the number of neighboring pixels with the same disparity estimation as the center, while the disparity scattering records the number of different disparity hypotheses within the patch. A random forest classifier is then trained to predict a confidence value for the corresponding scan line, which is used to determine the aggregate cost value through weighted summation

Some researches directly discard scanlines with bad estimations. Schönberger et al. (2018) attempt to approach the theoretical upper bound accuracy of SGM by always selecting the best scanline for disparity prediction. Therefore, they also train a random forest to find the best scanline, named SGM-Forest. Instead of using a handtuned feature to feed the random forest as (Poggi and Mattocchia, 2016), they simply use the disparity proposal of each scanline together with the cost for applying the disparity on all the scanlines to construct the feature vector. Thus the feature is built more efficiently with a straightforward delivery of each scanlines' estimation. At last, only the selected scanline and others with close prediction are accumulated for a confidence based weighted average of disparity. Xia et al. (2020) enhance the algorithm's robustness by setting the goal of the random forest as selecting all possible good scanlines, since in many cases there can be multiple well behaved choices. A single best target may confuse the forest's decision.

Besides SGM, there are also methods directly assisting the optimization of Markov Random Field (MRF) to balance the data and smoothness terms. Spyropoulos et al. (2014) attempt to guarantee the reconstruction density by correcting bad matches with the aid of reliable matches, instead of simply removing them. Thus, they train a binary classifier based on random forest to differentiate between good and bad matches, according to how close the estimation is to the ground truth value. Regarding the feature composition to feed the random forest, eight metrics are measured, which are the minimum matching cost among all the disparity candidates, the difference between the lowest and second lowest cost, the attainable maximum likelihood conversed from the cost curve, the distance to the nearest



image border, the distance to the depth discontinuity, the difference between the disparity of the center and the median disparity of the surrounding patch, and two left-right consistency measures. In the test phase, the selected reliable matching points are regarded as Ground Control Points (GCPs), and integrated as constraints into an MRF to correct wrong matches and optimize the target energy function. They impose soft constraints by setting GCPs' cost on non-selected disparities as a constant and large number (instead of infinity), which is experimentally proven to be more effective.

Due to the fact that the WTA strategy in traditional algorithms suffers from challenging situations, Shaked and Wolf (2017) design a Global Disparity Network (GDN) to replace WTA and estimate the disparity. The matching cost from their highway network is firstly regularized by a Cross based Cost Aggregation (Mei et al., 2011) and SGM. Afterward the cost volume is fed to GDN, which composes of several convolutional and fully-connected layers to predict the disparity. Besides, two extra fully-connected layers are added to construct another path for measuring the confidence of the prediction. Thus, GDN is jointly supervised by a cross-entropy loss by comparing the prediction with the ground truth, and a newly defined reflective loss which determines the quality of the prediction according to how close it is to the ground truth. The confidence can be used to support the following disparity refinement in left-right consistency check.

### 3.1.3 Learning based Post-Processing

After matching cost calculation and aggregation, an initial disparity map is obtained, via assigning an appropriate disparity value to each pixel so that an attainable minimum cost is acquired with spatial smoothness. Then some post-processing steps are added to refine the disparity results, e.g. classical methods may apply the sub-pixel enhancement to accomplish continuous depth prediction, detect occlusions via a cross-check between left-to-right and right-to-left matches, correct mismatches or invalid points using interpolation or plane fitting, etc. These steps can also be aided or substituted by learning based techniques.

Schönberger et al. (2018) improve their results obtained by SGM-Forest based on a confidence based spatial filtering. According to the prediction of the selected scanline by the random forest, other scanlines with a close estimation are also considered to fuse their disparity proposals via a confidence based average, and the sum of their confidence will be regarded as the correctness of the estimation. In a circular neighborhood with a radius of 5 pixels, disparity values of pixels with similar intensity and high correctness are selected. The disparity is determined by the median of these values. Thus, a spatially smoother disparity map is acquired.

Ye et al. (2017) argue that the WTA selected disparity could be a sub-optimal estimation, while the ground truth may lie closer to the disparity candidate associated with the second minimum cost. Therefore, they take the two disparity maps, which are corresponding to the minimum and second minimum cost, respectively, as input to a network for refining the raw estimation. The two disparity maps are firstly fused using convolutional layers without bias, and then (together with the master epipolar image) passed to an error detection module, which assigns a probability to each pixel indicating the likelihood of wrong matching. Afterwards, a parallel-replace module is designed to replace the erroneous estimation with new disparity labels, according to the previously computed probability. At here, two parallel networks are constructed inspired by ensemble learning (Liu et al., 2009), so that the spatial smoothness and local details are simultaneously considered. Finally, an overall refinement is applied based on residual learning.

Gidaris and Komodakis (2017) categorize the existing architectures for labeling refinement as hard refinement and soft refinement, which directly provides new estimation or correct the current results incrementally via adding an residual, respectively. However, they believe

that both solutions are sub-optimal. The hard refinement is essentially more complex, due to that a new label should be predicted. Moreover, the samples with correct initial estimation already, have to be considered in addition and revised with an identity transformation. On the other hand, the residuals based soft refinement can barely fix larger errors. Therefore, they subdivide the refinement procedure into the following subtasks: error detection, bad estimation replacement, and overall refinement finally. At first, an error map indicating the probability of each pixel with wrong prediction is obtained, which guides the following replacement task to assign new labels to erroneous regions (high error probability) with the aid of reliable estimation (low error probability). Then a refinement is further applied on all the points to enhance the overall labeling quality. The whole detection, replacement and refinement pipeline is fully trainable as end-to-end, which is compatible with multiple labeling scenarios. In case of stereo matching, a noticeable improvement is achieved by their proposed architecture, using the master image and an initial disparity estimation as input.

Instead of decomposing the refinement into a sequence of sub-tasks, Batsos and Mordohai (2018) exploit a residual based recurrent neural network (RNN), called RecResNet, to progressively modify the initial estimation. Given the reference image and a disparity map, RecResNet is able to correct large errors as the prediction is recursively improved. Furthermore, the residuals are estimated at different scales, which entitles the network to handle ill-posed regions in both large scale, e.g. large untextured areas, and small scale, e.g. fine structures, depth discontinuities, etc. Similarly, Jie et al. (2018) also use RNNs to iteratively refine the matching results. However, they integrate the disparity estimation and refinement within a common framework, which takes the matching cost calculated by (Shaked and Wolf, 2017) as input, and directly outputs a high quality disparity map without further post-processing. They focus on simulating the conventional left-right consistency check with neural networks, to avoid handcrafted setting, and propose their Left-Right Comparative Recurrent (LRCR) model. Concretely, a symmetrical architecture constructed by stacked convolutional Long-Short Term Memory (ConvLSTM) networks, is exploited to process the left-to-right and right-to-left matching cost, respectively. Within each ConvLSTM unit, rich context is encoded to obtain a disparity map for the left/right view and compared with the other (right/left) in order to calculate an error map. The error map (and the matching cost) for each view is then passed to the next ConvLSTM module for another iteration, which provides a soft attention map indicating the mismatched left and right regions, which is in turn used to recursively improve the estimate.

Knöbelreiter and Pock (2019) develop a collaborative regularizer to denoise the rough stereo matching results. Only a cost volume is needed as input, from which an initial disparity map is calculated simply via WTA, and an initial confidence map is acquired by transforming the cost to a matching probability distribution. Afterwards, the model learns a joint statistics among the image, the disparity and the confidence, so that a clean disparity map is generated. Multi-scale information is considered in their work based on a pyramid architecture.

## 3.2 End-to-End Neural Networks

Aided by machine learning, conventional stereo matching algorithms achieve higher performance with a supervised module to better deliver information through the pipeline, and avoid purely utilizing empirical strategies in each step which is essentially a shallow function with rigid definition. However, these methods still suffer from ill-posed regions, e.g. under occlusions, in large homogeneous areas without textures or only with repetitive patterns. Learning based matching cost algorithms are normally using a patch based similarity measure, requiring multiple forward passes to consider each disparity candidate. A smart feature extraction is applied, while only the surrounding neighbors are compared leading to

limited receptive fields. The following cost aggregation introduces more global clues, nevertheless, a well performed learning algorithm depends on a thorough training that could be time-consuming. Regarding the post-processing, modern CNN/RNN techniques could replace the hand-engineered steps, however, extra efforts are involved and the refinement relies more or less on the initial estimation.

Subsequent research focus on integrating each stereo matching step into an end-to-end network, including jointly extract feature from images, regularize the corresponding cost volume, and directly create a high quality disparity map. Instead of predicting a similarity score for each patch separately, the feature maps with the same size as the stereo pair are obtained, and a cost cube is created by shifting the feature map of the slave image along the epipolar line and concatenating it with the master feature map. The disparity range determines the number of horizontal displacements, assuming rectified stereo, and the size of the cost cube. Afterwards, a regularization strategy is applied to optimize the cost and predict the disparity map. The network is trained end-to-end to efficiently deliver the intermediate results to each module, and automatically adapt itself to produce a more reasonable solution.

The first end-to-end neural network for dense matching is FlowNet (Dosovitskiy et al., 2015). The network is not designed exclusively for dense matching, but for the more general optical flow estimation, which detects both the scene depth and the object motion. However, it represents the prototype for the following methods as it combines feature extraction, similarity measurement, and depth/motion estimation in a unified framework. Dosovitskiy et al. (2015) propose two architectures, FlowNetSimple and FlowNetCorr. The former directly stacks the input stereo pair as a "thicker" feature map to feed a generic network for an adaptive motion prediction. The latter uses a sub-network with two branches to extract feature from two images separately, and applies a correlation layer to compare the feature as:

$$c(p_1, p_2) = \sum_{m \in [-k, k] \times [-k, k]} \langle f_L(p_1 + m), f_R(p_2 + m) \rangle, \quad (3.2)$$

in which  $p_1$  and  $p_2$  are the reference and target point from the left  $f_L$  and right  $f_R$  image, respectively.  $m$  denotes the neighboring points to be considered within a  $(2k + 1) \times (2k + 1)$  patch. This correlation resembles a convolution, but it basically "convolves" a patch with another, without a trainable filter. Afterwards, a series of convolutions are used to extract high-level information, and then the feature is passed to a refinement module containing so called "upconvolutions", each of which consists of an unpooling and a convolution, to recover the resolution. Among them, an intermediate flow map is estimated and upsampled to be concatenated with the upconvolved feature map, together with the corresponding encoder's feature at the same resolution, so that the context is delivered with local details provided in next refinement. Finally, a full resolution flow map is obtained.

Regarding the subsequent development of end-to-end dense matching networks, 3D or 4D cost volumes could be generated, depending on each algorithm's specific strategy to compare the feature of the left and right image, e.g. feature correlation, feature concatenation and feature distances/differences. For example, FlowNetCorr produces a 3D cost volume as the stack of the correlation maps according to each disparity candidate, considering that the search of 2D optical flow is limited to 1D disparity between rectified image pair in dense matching.

### 3.2.1 3D Cost Volume (2D Convolution based Networks)

Following FlowNetSimple, DispNet (Mayer et al., 2016) is proposed specifically for dense matching. The network also exploits an encode-decoder structure, which they call contractive-expanding part, in order to enlarge the receptive fields through convolutions

with strides to observe larger disparities, and then upsample the features via upconvolutions to recover the resolution. Extra convolutions are applied in the expanding part (decoder) achieving smoother results than FlowNet. Besides, inspired by FlowNetCorr, DispNetCorr is designed to extract feature from each input image separately, and use a correlation layer to compare the extracted feature maps for further disparity prediction. As only 1D correlations are necessary to measure the horizontal displacements, DispNetCorr is computationally more efficient and allows finer disparity sampling rate. Mayer et al. (2016) initiate a new era for end-to-end dense image matching.

Knöbelreiter et al. (2017) also add a correlation layer in their network to build a 3D cost volume. However, the cost regularization is carried out using the conditional random field (CRF) (Lafferty et al., 2001) instead of a convolution based encoder-decoder. The calculated matching cost naturally becomes the unary term of the CRF, while the pairwise term is parametrized as the edge weights by a contrast sensitive model or a pairwise CNN. The structured support vector machine (SSVM) (Tsochantaridis et al., 2004) is used to train this hybrid of CNN and CRF as an end-to-end architecture. Comparable performance is achieved with higher efficiency, as the network is shallow without additional post-processing.

Pang et al. (2017) extend DispNet with an additional stage to learn a residual in order to improve the initial estimation. They apply DispNet to predict a full resolution disparity map as the first stage, from which an error map is computed by comparing with the ground truth. In addition, a synthesized left image is obtained via warping the right image according to the initial disparity estimation. Afterwards, they concatenate the left image, right image, disparity map, error map and the synthesized left image together to feed the following residual block. The residual block is also an hourglass (encoder-decoder) structure, which calculates multi-scale residuals. Each residual is added to the corresponding downscaled initial disparity map, so that multiple disparity maps are acquired for a joint supervision across scales. The two stages could be cascaded for end-to-end training, hence the algorithm is called cascade residual learning (CRL). With an extra residual learning to modify DispNet's estimation, better results are achieved especially for ill-posed areas, e.g. textureless regions, occlusions, etc.

Liang et al. (2018) define a feature constancy term to determine the correctness of the disparity estimation for further refinement. The feature constancy contains two metrics: feature correlation and reconstruction error. The feature correlation is basically the matching cost computed by a correlation layer. The reconstruction error is computed as the absolute difference between the left feature and the warped right feature according to the disparity estimation using a DispNet. Thus the initial disparity map and the feature constancy term are passed to their disparity refinement module, so that a residual is calculated to improve the estimation. Since the refinement could be performed iteratively until the expected small differences are acquired between two consecutive iterations, the network is named as iResNet (iterative residual prediction network).

In order to better handle occluded areas, Ilg et al. (2018) propose a network to jointly estimate disparity and detect occlusions. Based on an encoder-decoder structure as FlowNet and DispNet, the disparity (or optical flow depending on the demand) together with the occlusions and depth boundaries are estimated, which are refined continuously by the following refinement network. They define several network modules. For example, a residual module is designed to improve the estimation following (Pang et al., 2017). The occlusions are detected by applying the correlation layer twice, which matches the feature from the left frame to right and right to left, respectively. Thus, the inconsistent matches could be regarded as occlusions. The proposed algorithm is capable of achieving much better results in occluded regions with good efficiency. Besides, the scene flow can be estimated via combining a flow and a disparity network.

Some researches focus on unsupervised learning to avoid cumbersome data annotation for applications without training data. Inspired by the traditional left-right consistency check to remove outliers, Zhou et al. (2017) design a network to adaptively select confident predictions to supervise the training with no need of ground truth. They start from a randomly initialized network to extract features, create a correlation based cost volume, and predict disparity. It should be mentioned that they concatenate the cost volume with the feature maps, in order to incorporate the information from input for better cost regularization. Afterwards, disparity maps are predicted for both left and right frames, in which a left-right consistency check is applied to exclude outliers and use the inliers to train the network iteratively. As more iterations are finished, the selected confident set becomes larger until a well-performed model is obtained. Finally, a left-right consistency check and a median filter are used to refine the network estimation. The proposed unsupervised architecture achieves comparable performance with other supervised models, e.g. DispNet, MC-CNN, etc. The left-right consistency check can also be applied to calculate an error map, so that the network is guided to focus more on unreliable regions for refinement (Zhang et al., 2019).

The above networks are built based on elaborate architecture design. As the development of Automated Machine Learning (AutoML) (Hutter et al., 2019), Saikia et al. (2019) attempt to automatically determine an optimized network structure for higher performance. Hence, they regard DispNet as the backbone and optimize the encoder-decoder architecture using an AutoML technique, DARTS (Liu et al., 2018). A search space is firstly defined, so that a series of cells (meta structures) are available to compose the network. Then the continuous relaxation is implemented on the search space to make the cells differentiable, in order to enable the gradient descent based training. Thus, a training and a validation dataset could be generated to alternatively adjust the network parameters and find the final optimal setting. Finally, the BOHB (Falkner et al., 2018) method is used to tune the hyperparameters. Their AutoDispNet outperforms the baseline achieving SOTA performance.

### 3.2.2 4D Cost Volume (3D Convolution based Networks)

The above methods mainly use the correlation layer to construct 3D cost volumes, which somehow still relies on the traditional correlation metrics to measure the similarity between the left and right feature. Moreover, the feature dimension of the input is flattened in the cost volume, and the following 2D convolutions only aggregate the local context along the height and width of the volume, rather than the disparity dimension. Hence, the subsequent researches attempt to concatenate the feature maps of the stereo input to build up the cost volume with absolute representation (instead of an explicit definition: correlation), so that more freedom is left to the network for adaptively learning the correspondences. The feature maps from the left and right image are concatenated along the channel dimension, with a horizontal shift along the row indicated by a disparity candidate, resulting in a 4D cost volume with a dimension of  $height \times width \times N_{disp} \times 2C$ . In addition to the height and width of the volume,  $N_{disp}$  denotes the number of disparity candidates and  $C$  is the channel length. Correspondingly, a 3D convolution based encoder-decoder is used to regularize the cost volume, in order to learn the stereo across the height, width and disparity, respectively, for better learning the geometry and context. GC-Net (Geometry and Context Network) is proposed accordingly (Kendall et al., 2017). As the extra dimension of the cost volume and the corresponding 3D convolutions bring additional computation burden, the cost volume is downsampled by a factor of 32 in the encoder. Thus, the receptive field is also enlarged to perceive a larger context. Afterwards, a series of deconvolutions are used to gradually recover the resolution. A skip connection is added between feature maps at the same resolution from the encoder and decoder, respectively, in order to include detailed structures for a better estimation. At last, they define a soft argmin to regress the disparity using the regularized cost, which achieves sub-pixel accuracy and enables end-to-end training as a

differentiable operation. The soft argmin is defined as:

$$\hat{d} = \sum_{d=0}^{D_{max}} d \times \sigma(-C_d), \quad (3.3)$$

in which  $\sigma$  transforms the cost to a normalized probability of each disparity candidate. Then the final result  $\hat{d}$  is computed as the probability based weighted summation within the disparity range  $[0, D_{max}]$ . GC-Net achieves SOTA performance by leveraging both geometry and context. In following content, the soft argmin is utilized as the default disparity regression method by all the introduced networks, unless otherwise specified.

The pyramid architecture is also exploited to generate 4D cost volumes for including multi-scale information. Chang and Chen (2018) propose their pyramid stereo matching network (PSMNet) using a spatial pyramid pooling (SPP) module, in order to extract region-level features across scales. Thus, each target is better understood especially for ill-posed regions. PSMNet uses residual blocks to extract unary features from the input images, with dilated convolutions added to further extend the receptive field. Afterwards, a SPP module with four average pooling blocks in fixed sizes,  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$  and  $8 \times 8$ , are used to obtain multi-scale features. The acquired feature from each pooling block is upsampled to a unified resolution, based on bilinear interpolation, and concatenated together to generate a cost volume. Two architectures are designed for the cost aggregation. The basic one simply applies residual blocks with twelve 3D convolutional layers to regularize the cost, while the other advanced structure uses three encoder-decoder modules to repeatedly learn the global context in a top-down/bottom-up manner. Besides, a disparity map is computed for each hourglass unit to allow for intermediate supervision. The use of multi-scale feature and the stacked hourglass cost regularization grant PSMNet the best performance on KITTI-2012 and KITTI-2015 benchmarks in 2018.

PSMNet has been further improved by many researches in accuracy and efficiency. For example, based on PSMNet’s SPP module to merge multi-scale features, Nie et al. (2019) design a multi-level context ultra-aggregation (MCUA) strategy to produce a more discriminative representation of the matching cost. As for the cost aggregation, the previous techniques for fusing multi-level context information, such as DenseNets (Huang et al., 2017) and DLA (Yu et al., 2018), only accomplish intra-level combination of features without enough low-level details. Hence, the proposed MCUA scheme adds an independent child branch, which additionally takes a downsampled input via average pooling to enlarge the receptive field, so that inter-level features could be fused explicitly with the main branch. With MCUA introduced into PSMNet, the performance is increased by a notable margin. Cheng et al. (2020) modify the spatial propagation networks (SPN), which is a SOTA linear propagation module, by executing the propagation with a series of recurrent convolutions. Thus, a convolutional SPN (CSPN) is achieved to better learn the affinity within each patch to aggregate features. The combination of CSPN and PSMNet dominate the KITTI-2012 and KITTI-2015 benchmarks for quite a certain period. Also, Chabra et al. (2019) utilize vortex pooling (Xie et al., 2018) to replace the SPP in PSMNet, which increases the utilization ratio exponentially to perceive more pixels for feature extraction. Besides, the dilated convolution is used to enlarge the receptive field in cost regularization. They also design a disparity refinement module. By warping the right image to the left frame, according to the initial disparity estimation, an image reconstruction error map can be computed as the difference between the left and warped right image. Similarly, a geometric consistency error map is obtained from the left and warped right disparity map. They feed the left image, left disparity map, image reconstruction and geometric consistency error map to the refinement module, which outputs a disparity residual map for refinement. Guo et al. (2019) improve PSMNet by constructing a group-wise correlation cost volume and propose the group-wise

correlation stereo network (GwcNet). As for the group-wise correlation, the channels of the feature map are divided into several groups, after which a correlation map is calculated for each group only using the contained features. Afterwards, the correlation map for each group is concatenated along the channel dimension, thus, leading to a 4D cost volume with the channel length equal to the number of groups. The reason for this design is to make a compromise between a full correlation and a concatenation based cost generation. The former loses too much information for producing only one correlation value for each disparity candidate, while the latter cannot explicitly support the similarity measurement. They prove that, the group-wise correlation based cost volume requires less parameters for the following cost aggregation, to achieve similar performance as PSMNet using the concatenation based volume. Furthermore, several modifications are applied on PSMNet’s stacked hourglass structure for cost aggregation. First of all, a pre-hourglass unit is added at the beginning, which consists of four 3D convolutions, in order to let the network learn features at shallower layers via the intermediate supervision. Besides, the residual connection across different hourglasses is removed, so that certain hourglasses can be excluded in test for saving time. At last, a 3D convolution with kernel size as one is added to the residual connection within each hourglass module. Experiments show that GwcNet performs better than previous methods on several close-range benchmarks.

Duggal et al. (2019) also exploit SPP to extract multi-scale information. However instead of constructing the cost volume using all the disparity candidates, they prune the full search range to a narrower but more confident range to save memory and runtime. By referring to the PatchMatch method (Barnes et al., 2009), a particle sampling layer is designed to predict  $k$  random disparity candidates as an initial estimation. It is mentioned in the paper that, the full disparity search range is divided into  $k$  intervals and each initial selection is forced to happen in a certain interval to keep the selected disparity candidates covering the full range. Afterwards, an iterative processing is applied between a propagation and a evaluation layer, in order to continuously refine the selection. For every evaluation, a small subset of disparities is defined around each selected disparity candidate. A similarity score is calculated for each disparity within the subset, which is used for a weighted summation as a refined disparity. Thus,  $k$  new disparity candidates are available from each subset for further estimation. The selected disparity candidates can be propagated to the target pixel’s four neighbors. Based on the above strategy, a narrow disparity range is predicted to confidently include the ground truth, from which a pruned cost volume is generated for efficient disparity estimation.

Tulyakov et al. (2018) find that the concatenation of the left and right image descriptors for constructing the cost volume, may allow more context to be learned in following 3D convolutions based encoder-decoder. However, this causes costly memory usage which should be avoided in handling higher resolution data. Therefore, they refer to (Zagoruyko and Komodakis, 2015; He et al., 2016; Mayer et al., 2016; Pang et al., 2017) and design a “bottle-neck” architecture to compress the concatenated feature into a more compact matching signature for further regularization. Moreover, they argue that the normally used soft argmin for disparity regression returns a sub-optimal estimation, if the posterior disparity distribution from the regularized cost is multi-modal. Thus, a weighted average through the whole disparity range could be far from the ground truth. Hence, they empirically set a small range centered at the disparity candidate with maximum posterior probability, and apply the soft argmin only on the disparities contained by the range. Regarding the training, a sub-pixel cross-entropy loss is proposed to substitute the typically used  $L_1$  loss, for faster convergence and higher accuracy.

Also aiming at reducing the computational effort, Khamis et al. (2018) regularize a very low resolution cost volume (typically 1/8 or 1/16 of the original resolution) to directly predict a disparity map. They believe that most runtime and memory are spent on processing the

higher resolution cost, while usually lower resolution matching provides the largest performance gain, as it contains richer context information relevant to the task. Besides, they suspect that other methods with a decoder to gradually recover the resolution, may overfit to the training data, as the stereo matching is essentially a pixel-to-pixel mapping. Hence, they determine to use the input image for the refinement of the low resolution disparity map, with the high frequency details added along with the recovered resolution. Their implementation achieves the first real-time architecture for dense matching.

As the current SOTA networks either require long runtime and large memory to produce a high quality disparity map, or increase the algorithm’s efficiency with reduced accuracy, Wang et al. (2019) decide to grant the users the freedom to make a trade-off. They propose AnyNet, which generates improved disparity results as time goes by and provides the temporally best estimation. Thus, the algorithm is able to adaptively adjust itself for better efficiency or accuracy according to the specific scenario, e.g. a self-driving car in a open or crowded field. The network exploits a pyramid architecture, in which the top utilizes the downsampled feature maps for a coarse depth estimation as the first available stereo result. Afterwards, if time permits, the disparity map is passed to the next higher resolution pyramid level for refinement. AnyNet is the first network providing anytime estimation. In addition, thanks to its hierarchical estimation from coarse to fine, the algorithm is highly efficient even always pursuing the best results. Similarly, Yang et al. (2019) also propose a network for anytime on-demand prediction, named hierarchical stereo matching (HSM) network. However, they focus more on high resolution stereo data processing for more precise depth estimation, since the same disparity error would cause quadratically larger depth error for farther objects. Accordingly, a pyramid architecture for coarse-to-fine estimation is designed. Multi-scale features are firstly extracted from the input stereo pair. Afterwards, a cost volume is constructed for each scale, by computing the differences between the corresponding feature maps. The coarsest volume is firstly processed by their volume decoder via a series of 3D convolutions and a volumetric pyramid pooling, which is an extension of SPP to additionally consider the disparity dimension. Thus, the cost volume is regularized, for predicting an intermediate disparity map for practical use. Besides, the volume could also be upsampled and merged with the volume on the next pyramid level with higher resolution, for finer disparity estimation. The anytime setting of HSM allows relatively accurate depth estimation for closeby targets, with low latency up to 30 ms. Moreover, they deliver two high resolution stereo datasets, one synthetic and one real data, for training and test.

SGM is widely used to aggregate the cost volume, with neighboring points considered for smoothness along multiple scanlines. This inspires Zhang et al. (2019) to learn the context in a "semi-global" manner as SGM, instead of the normally used encoder-decoder as a pure learning module. Thus, the success of SGM naturally becomes the foundation to design a new model. Specifically, they propose the guided aggregation network (GA-Net), which approximates the scanline optimization as a differentiable layer (semi-global guided aggregation (SGA) layer) to automatically learn the penalty imposed on disparity discontinuities. In this way, no human intervention is needed to define the penalty terms and each pixel can be treated independently according to the specific situation. Moreover, the use of the computationally expensive 3D convolutions is reduced, leading to higher efficiency. They also design a local guided aggregation (LGA) layer to further filter the cost volume, such that thin structures are recovered. GA-Net can be trained end-to-end, achieving SOTA performance. Besides, it brings a new thought to build a learning architecture by referring to the conventional algorithms which has been proven effective through decades.

Yu et al. (2018) argue that more attention should be paid on the cost aggregation step in an end-to-end stereo system, thus propose an embedding sub-architecture to generate multiple aggregation proposals and adaptively select the most reasonable one. In this way, the cost regularization is reformulated as an explicit learning task, rather than a black box solely



from the encoder-decoder. Their network owns a similar backbone as GC-Net for feature extraction and concatenation to form the cost volume, and a 3D convolution based encoder-decoder is utilized for an initial regularization. Afterwards, a sub-network is added to generate potential aggregation proposals. 3D convolutions with rectangle kernels are applied on the cost volume, which are in size of  $3 \times 1 \times 1$ ,  $1 \times 3 \times 1$  and  $1 \times 1 \times 3$ , to aggregate the information along the disparity, height and width, respectively. Thus, each dimension is processed separately, reducing the computational complexity compared to full 3D convolution. The resultant cost volume is in 4D with its channel length representing the number of aggregation proposals. On the other hand, a series of convolutions is applied on the reference input image to create a guidance map containing low-level structured information. A matrix multiplication is carried out between the aggregated cost and the guidance map, and the maximum calculated value is selected for each disparity candidate as the best proposal. A soft argmin could then be applied on the cost to predict a disparity map. Their learning based cost aggregation strategy obtains support from the reference frame as a global view guidance, leading to SOTA results on several benchmarks.

### 3.2.3 Multi-Task Learning

In the field of artificial intelligence, we normally guide a model towards a fixed target according to the specific tasks. It is, however, gradually demonstrated that some related tasks may provide clues to better supervise the model to generalize on our original task, which leads to multi-task learning (Ruder, 2017). Multi-task learning is also applied in stereo matching, for example adding a sub-task to extract the semantic information for locating the depth edges.

(Zhan et al., 2019) attempt to combine the semantic segmentation and stereo matching into a unified framework. Since the two tasks could support each other, a common semantic encoder is shared to extract both semantic and disparity features, achieving a lightweight model named DSNet. The training is separated into three steps. Firstly the semantic sub-task is learned which could provide auxiliary feature for the stereo sub-task. In order to balance among classes, a weight is assigned to each class according to the corresponding pixel quantity. Afterwards the semantic network is frozen to train the disparity network specifically, in which DispNet is referred to estimate the disparity. It should be noted that both feature correlation and concatenation are exploited to generate the cost, with an attention mechanism (Xu et al., 2018) introduced to combine the cost volume. Finally, a joint supervision is applied to train both tasks. As only one (left) image owns semantic labels, the calculated disparity is used to warp the right image into the left frame so that both images are included within the semantic loss for a more robust estimation. The combination of the two tasks promotes both scene parsing and stereo matching.

Likewise, SegStereo (Yang et al., 2018) also suggests to incorporate semantic feature into stereo matching pipeline. With ResNet-50 (He et al., 2016) as the backbone to extract features from the input image pair, a cost volume is constructed via a correlation layer. In addition, a segmentation network is used to acquire semantic feature maps. Afterwards, the disparity feature from the left image is transformed by a convolution block to preserve details, and then concatenated with the cost volume and left semantic feature, which is then feed into the encoder-decoder based disparity estimation. Thus, the semantic clues are embedded as a high-level guidance, to provide more robust representation for better locating the stereo correspondence in ill-posed regions. Furthermore, the calculated disparity map can be used to warp the right semantic feature map, so that the ground truth of the segmentation labels for the left image could supervise both frames. SegStereo supports both unsupervised and supervised training, based on photometric loss and regression loss, respectively, achieving SOTA performance. In the algorithm, however, only unidirectional

support is obtained from semantics to stereo. Similar work is found in SegFlow (Cheng et al., 2017) for joint estimation of video segments and optical flow.

Inspired by PSMNet Net and SegStereo, Wu et al. (2019) propose their SSPCV-Net to build up a pyramidal cost volume to contain multi-scale features, and simultaneously carry out a semantic segmentation sub-task to assist the disparity boundary estimation. Compared with PSMNet which produces a single cost volume from multi-scale feature maps, however, SSPCV-Net constructs a cost volume for each scale separately. Meanwhile, the highest resolution features are also used to generate a semantic cost volume, via concatenating the semantic feature maps of the left and right image which are acquired based on PSPNet (Zhao et al., 2017). From here, a 3D multi-cost aggregation module is applied to fuse the spatial cost volumes from the lowest resolution to the highest, together with the semantic cost volume. As for each fusion, the cost volume at lower resolution is upsampled to the same size as its immediately higher scale for a summation in between. Afterwards, a 3D feature fusion module (FFM) is designed, which consists of a 3D adaptive pooling and a fc-ReLU-fc-sigmoid structure (Hu et al., 2020), in order to predict a weight for a weighted summation of the two cost volumes. Thus, all the cost volumes could be aggregated together for the final disparity estimation. Their FFM well integrates multi-scale spatial information and semantic clues, achieving much better results than previous SOTA methods.

The stereo correspondence search is usually hampered by textureless areas and boundaries. Therefore, the edge information is promising to guide the matching procedure for more reasonable disparity assignment. Accordingly, EdgeStereo is proposed in (Song et al., 2020b) to construct a multi-task network for both disparity estimation and edge detection. The network consists of two branches for the two tasks, which share the shallow part for feature extraction. Afterwards, the features from both images are connected by a correlation module to build a cost volume, while the left feature goes deeper for edge detection. Then the detected edge feature is incorporated into the stereo matching branch, as the first support, by concatenating the edge map with the cost volume. Thus, the geometric information is considered in further disparity estimation. After a convolution based encoder for deep representation and context perception, a residual pyramid decoder is designed to recover a full resolution disparity map from coarse to fine. A disparity map at the lowest resolution is firstly computed, which is continuously upsampled and refined in higher resolution pyramid level via a residual learning. At here, the second support from edge information is acquired, via defining an edge-aware smoothness loss to penalize the disparity discontinuities, such that a more semantic meaningful optimization is realized. Regarding the training, three stages are carried out. The first stage solely trains the edge detection branch based on a class-balanced cross-entropy loss (Liu et al., 2019). Then the stereo branch is trained with fixed weights of the other branch. At last, both branches are optimized, except the shared shallow backbone. In the paper, they prove that the two target tasks could promote each other. The disparity map acquires sharper boundaries, while the model after multi-task training predicts better edges than previous model purely trained on edge detection dataset.

The data labelling remains to be a problem in the field of computer vision, as cumbersome annotation is needed to provide dense and accurate labels for training. Jiang et al. (2019) propose a multi-task network, Shared Encoder Network for Scene-flow Estimation (SENSE), to simultaneously estimate disparity and optical flow, detect occlusions and segment the scene. Thus, the interaction among different tasks can support each other even with partially labelled ground truth. For example, the semantic information is able to assist the disparity or optical flow estimation, which in turn help to locate the occlusions. Accordingly, three losses are defined as supervised loss, distillation loss and self-supervised loss. The supervised loss is naturally based on pixels with ground truth labels, while the self-supervised loss is measured by the photometric consistency or semantic consistency via warping the

images or semantics onto a common frame. The distillation loss uses the model pre-trained on well labelled data, e.g. synthetic data, to train the network on real data with scarce labels. Furthermore, one encoder is shared by all the tasks for a compact network design and interaction.

El-Khamy et al. (2019) design an interesting end-to-end network to estimate the scene depth, which is captured by two cameras with different field of views (FOVs). One camera uses a tele zoom lens, with its FOV contained within the other camera's FOV equipped with a wide angle lens. The overlapped FOV between the cameras could provide the stereo disparity results. However, predicting the depth of the surrounding area outside the narrow FOV but within the wide FOV is difficult. Accordingly, they propose three architectures, named tele-wide stereo matching network (TW-SMNet), single image inverse depth estimation network (SIDENet) and multi-task tele-wide stereo matching network (MT-TW-SMNet). TW-SMNet is able to process the overlapping FOV, which is basically from the tele zoom lens, however, obtains bad results on the outer region with incomplete binocular stereo information. As SIDENet is trained to understand the scene for monocular disparity prediction, better results are acquired on the outer region than TW-SMNet. Hence, it inspires the authors to combine TW-SMNet and SIDENet to construct MT-TW-SMNet. The two branches (TW-SMNet and SIDENet) could assist each other, leading to further improved estimation. For example, the disparity estimation from TW-SMNet refines the prediction of SIDENet, by providing the disparity results of the overlapped FOV, as prior knowledge of the scene's depth. The proposed methods can be applied to blur the background for esthetically better effects in image Bokeh.

### 3.3 Confidence Measurement

Although the SOTA deep learning based methods have achieved high quality stereo results, which outperform the conventional methods by a large margin, a confidence measure should be available to indicate the correctness of the estimation. Besides acting as a self-evaluation about the results, the confidence is particularly important in applications aiming at providing accurate prediction for reliable machine guidance, e.g. self-driving, in order to avoid disastrous consequences (Kendall and Gal, 2017). Therefore, some studies focus on the confidence measurement to complement modern algorithms, so that a more complete stereo product can be delivered.

Some algorithms design a module to specifically estimate the confidence of the disparity map, which is obtained from an independent pipeline. Fu and Fard (2018) propose several CNN based models for stereo confidence estimation, which take the initial disparity map and the corresponding reference RGB image as input. The designed models are categorized into two types, basic models and effective models. The former alternatively merges the initial disparity map and the RGB image as a four channel input, or organizes two branches to process the disparity and RGB, respectively, before a further decision sub-network to estimate the confidence. The latter adjusts the two-branches version, by utilizing dilated convolutions to increase the receptive field in either of the two branches or both, or in another separate branch to additionally process RGB information. Both basic and effective architectures achieve SOTA performance, while the latter could generalize better as larger receptive field is observed. However, the algorithm is patch based and can only consider a small neighborhood around each pixel for confidence measure. Hence, Tosi et al. (2018) add a ConfNet in addition, which absorbs global information from the disparity map and the reference image, such that a far larger receptive field is perceived and a smoother confidence map is obtained. The ConfNet uses an encoder-decoder structure to process the concatenated feature extracted from the disparity map and the image, and outputs a confi-

dence map with the same resolution. Then they cascade the ConfNet and a local approach such as (Poggi and Mattochia, 2016) and (Fu and Fard, 2018), so that local and global information is fused for the final confidence estimation. They nominate the method as Local Global Confidence Network (LGC-Net), which outperforms the previous local approaches. Kim et al. (2017) emphasize the spatial consistency in confidence estimation by extracting confidence features from both pixel-level and superpixel-level. Thus, unreliable estimation on textureless or occlusion areas is alleviated. The pixel-level confidence feature vector is constructed using a series of metrics from (Park and Yoon, 2015), according to the matching cost volume and the disparity map. On the other hand, the confidence measure within each superpixel is obtained via a GMM clustering model (Biernacki et al., 2000). A superpixel map could be generated from the color image based on an off-the-shelf superpixel segmentation algorithm, e.g. the SLIC method (Achanta et al., 2012) used in the paper. Afterwards, the combination of pixel-level and superpixel-level confidence features is fed to a random regression forest to predict a confidence map. In addition, they design a hierarchical confidence map aggregation (HCMA) strategy to further enhance the confidence estimation in test period, based on K-Nearest Neighbor (KNN). The proposed method achieves SOTA performance.

The work from (Kim et al., 2017) is extended in (Kim et al., 2019) via a CNN to estimate the stereo confidence, according to the cost volume and initial disparity map. The network consists of two sub-networks, a matching probability construction network (MPCN) and a confidence estimation network (CEN). The former transforms the matching cost volume to a matching probability volume, which can better handle the scale variation problem. The latter is capable of learning a mapping from the matching probability and disparity estimation, to a confidence measurement, by searching the top-k matching probability and the corresponding disparity. In order to better train the network with sparse ground truth, a semi-supervised strategy is designed to use confident pixels via the image reconstruction loss. The network is proven effective using three post-processing algorithms: cost modulation (Park and Yoon, 2015), GCPs-based propagation (Xu et al., 2013) and aggregated GCPs-based propagation (Min et al., 2014).

There are also works which measure the confidence of the estimation as a byproduct, together with the stereo matching results. SGM-Forest (Schönberger et al., 2018) summarizes the confidence of all the hypotheses close to the selected disparity hypothesis, as the confidence measure of the random forest disparity prediction. This confidence measure could support their confidence based spatial filtering, which selects neighboring pixels with confident prediction to compute the disparity median as the final estimation. The median of each neighbor’s confidence estimation is the corresponding correctness. Shaked and Wolf (2017) add two more FC layers in their network as a separated branch, which uses the feature for disparity estimation to additionally predict the confidence, in parallel with the disparity prediction branch. Mehlretter (2020) argue that most deep models only predict the confidence according to the uncertainty present within the data or tasks, instead of the stereo processing itself. Accordingly, they study both of the aleatoric and epistemic uncertainty in their network. The former could locate the difficult regions in matching, where the error mostly originates, e.g. depth discontinuities, reflective areas, occlusions, etc. The latter defines the limitation of the model which is trained on finite training samples, resulting in bad generalization in a new domain. The epistemic uncertainty can be theoretically explained away with sufficient training data, which is not possible for the aleatoric uncertainty. However, the latter doesn’t increase in test period. Concretely, a probabilistic neural network is proposed based on the GC-Net backbone, from which both disparity and confidence are estimated. The aleatoric uncertainty is estimated via a separate branch after the cost regularization, by comparing the predicted disparity with the ground truth. Regarding the epistemic uncertainty, a Bayesian method is applied to create a probability distribution of the network’s parameters, in which the weights of the network are sampled rather than directly

learned. Hence, different disparity maps are calculated from a single stereo pair, by sampling different sets of the network weights, and the epistemic uncertainty is approximated according to the variance of the prediction. The training could thus optimize the probability distribution via the mean and variance of the network's parameters. In test period, the disparity is predicted multiple times for each stereo pair, from which the mean of the disparity predictions is regarded as the final estimation. Images from a different domain may lead to bad estimation, however, the variation of multiple predictions (of the same image) already well represents the model's weakness in the confidence map. The experiments prove that the estimated confidence considering both uncertainties is more consistent with the error map.

### 3.4 Cross Domain Estimation

Although SOTA networks continuously promote the development of dense stereo matching in both accuracy and efficiency, overfitting on the training domain is an unavoidable issue as there is no datasets covering all possible situations. The network may suffer from achieving good predictions, when fed with images owning different color styles, lighting or exposure conditions, view angles, resolutions, etc., from the data used for training. Besides, a synthetic dataset is widely utilized to pre-train the models as it could provide huge amounts of training samples with precise and dense annotation. However, discrepancy exists between simulated and real environment making the finetuning on real data indispensable. Therefore, some studies focus on designing a robust network which can generalize well cross domains, so that a good estimation is acquired even for unseen scenes and targets. Two main categories of the network adaption are proposed, offline and online.

Finetuning is regarded as a normal offline adaption, which allows a network to better estimate stereo correspondences for unfamiliar scenes. The ground truth for practical scenarios could be hard or expensive to collect, however, the stereo images in the target domain could be easier to obtain. Hence, extracting appropriate stereo knowledge from the available images only is naturally considered as a good direction for a pre-optimization. Tonioni et al. (2017) propose an unsupervised adaption strategy to leverage conventional stereo matching algorithms, e.g. SGM, for disparity estimation on the test data as they are generally more robust for different scenarios, in order to supervise a pre-trained network. To exclude unreliable predictions of traditional approaches, a confidence measurement CNN (Poggi and Mattocchia, 2016) is used to adapt the network only on highly confident estimations. Accordingly, a confidence guided loss is defined, in which the network's disparity prediction is guided to approach the traditional algorithm's prediction if its confidence is above a pre-defined threshold. Moreover, another loss function is designed to enforce the smoothness term, by minimizing the disparity difference within the neighborhood. Thus, the highly confident estimation can be propagated to low confident ones. Experiments indicate that after the proposed adaption, the predicted disparity map from DispNet is smoother and more accurate. Pang et al. (2018) obtain empirical findings that a pre-trained network achieves blurred boundary reconstruction and erroneous estimation on ill-posed areas in the target domain, which however could be improved when fed with an upsampled version of the same stereo pair. Hence, they propose their "zoom and learn" (ZOLE) model, in joint supervision of synthetic data with ground truth and stereo pairs from the target domain without ground truth. Regarding the former, a normal  $L_1$  loss is applied according to the comparison between the prediction and ground truth. As for the latter, the network is fed with stereo pairs from the target domain in the original resolution and properly upsampled resolution, respectively. The disparity prediction on each higher resolution input is downsampled to the original resolution, which is found to still own finer details compared with the stereo results on original resolution input. Therefore, another loss is introduced by using the dis-

parity map of higher resolution input to guide the lower resolution stereo matching, based on graph Laplacian regularization (Elmoataz et al., 2008; Milanfar, 2013). The pre-trained ZOLE model is tested on images collected using smart phone cameras on a driving car in street view. The predicted disparity maps own sharper edges and more local details after the adaption. It is mentioned in the paper that the resolution should not be increased excessively for supervising the network in the target domain, to keep a reasonable receptive field.

Although the most effective domain adaption should be offline before bringing the model to practical scenarios, the pre-trained model can also be adjusted online as each new image is observed in test period, to conduct early prediction. Tonioni et al. (2019) offer a framework to adapt the pre-trained model online towards the target domain. Considering the practical use, good efficiency should be acquired, since the model stays in training mode during the test period to optimize the network parameters after each backward propagation. Thus, a pyramid structure is applied, which generates multi-scale feature maps. The stereo correspondences are firstly located at the lowest resolution, with a correlation layer used to measure the matching cost and a decoder to estimate the disparity. Then the disparity map is upsampled to match the resolution of the next higher resolution pyramid level, and acts as the initial estimation from which only a small disparity range is needed for further correcting the results softly. Hence the disparity of each pixel is refined by searching the corresponding point within a narrow range through the pyramid, achieving a high efficiency. The model could be adapted by unsupervised loss functions, e.g. a reconstruction error which measures the difference between the left image and the warped right image according to the disparity prediction. They name the network as Modularly Adaptive Network (MADNet). Besides, they also design a strategy to partially tune a subset of the network, in order to achieve a faster adaption. A portion of the network is sampled in each iteration, which would acquire a higher superiority in next sampling if the optimization on the portion brings positive effect to the overall performance. Specifically, positive effect represents that the current two consecutive iterations enable a decrease of the loss more than the previous ones. The proposed method leads to the first stereo system with real-time self adaption. Similarly, Zhong et al. (2018) propose an online network adaption scheme to process continuous stereo video, which is capable of improving the depth estimation gradually as more frames are observed. Since the strategy enables their stereo network to be directly applied in unseen open world environment, it is nominated as "OpenStereoNet". Specifically, OpenStereoNet consists of feature extraction, encoder-decoder based cost regularization, and a projection layer (soft argmin) to predict the disparity. The online adaption is realized based on a photometric warping error via comparing the right and the warped left image according to the disparity estimation. In addition, two Long Short-Term Memory (LSTM) units are added to the bottleneck of the feature extraction and cost regularization module, respectively, so that the network could provide a smooth disparity sequence along the temporal axis according to its past experiences. Tonioni et al. (2019) provide a strategy to search a base model for faster online adaption to the unseen target domain, based on Model Agnostic Meta Learning (MAML) (Finn et al., 2017). Besides, a confidence measure is applied for a weighted unsupervised loss calculation, thus the noise from doubtful estimation is alleviated during the adaption.

In addition to the adaption strategy, online or offline, to finetune the pre-trained model towards the target domain, the data from the training domain can already be transformed to own similar appearance, e.g. color style, with the test data, such that the domain shift is relieved when transplanting the pre-trained network to practical scenarios. Also, the extracted low-level feature can be normalized, such that the domain property is limited before building up the cost volume. Based on the two findings above, Song et al. (2020a) propose AdaStereo to preprocess the input stereo pair and the output of the feature extraction module, in order to obtain a more robust and general representation for disparity estimation

across domains. During the pre-training of the network on synthetic data, a color transfer module is applied to transform each synthetic stereo pair towards a randomly selected test stereo pair, e.g. from KITTI or Middlebury, before feeding to the network. Thus, a similar color style is acquired between the two different domains. The color unification is carried out in the LAB color space, according to the mean and standard deviation of the color values. Regarding the matching cost, the extracted feature is normalized along the spatial and channel dimension successively, and then used to construct the cost volume via correlation or concatenation. Thus, the cost volume can be regularized within a fixed range to narrow the domain gap. Their experiments demonstrate that the pre-trained AdaStereo network on synthetic data achieve comparable performance with other SOTA domain-invariant methods. Besides, the two preprocessing steps could be embedded into any SOTA stereo networks without incurring extra trainable parameters. However, the color transfer for every single stereo pair still increase the computation burden. Furthermore, some researches aim at training a robust network on, e.g. synthetic data, and keeping an acceptable performance for real data without finetuning or adaption. Zhang et al. (2020) design a domain-invariant stereo matching network, DSMNet, for robust cross domain estimation without the need of explicit adaption. In order to reduce the domain differences, the common batch normalization (BN) layer is replaced by a newly designed domain normalization (DN) layer after each convolution. Traditional BN normalizes the feature within each channel through the whole batch, which can amplify the domain property. On the other hand, another conventional normalization layer, instance normalization (IN), uses the feature of each channel from each input image independently for normalization, thus limiting the image level variation and obtaining a better cross domain estimation. Nevertheless, the feature vector (along the channel dimension) of each pixel is not fully considered, leading to susceptible similarity measure when inconsistent feature norms or scaling exist. Hence, DN is designed which comprises an IN for the first normalization, and then normalizes the feature along the channel again, so that a robust feature measure is accomplished considering both spatial and channel dimensions. Also, a non-local aggregation layer is proposed, which is a general form of their SGA layer (Zhang et al., 2019), for better structural and geometric representation in both feature extraction and cost aggregation. Based on the two modifications, DSMNet outperforms all the SOTA networks pre-trained on the same synthetic data, when applied on real data, and even surpasses some finetuned models. Cai et al. (2020) argue that a network can acquire better generalization ability, if it is fed with feature from the matching space (MS) instead of image appearance, i.e. pixel intensity or RGB values. Therefore, they replace the feature extraction module from an end-to-end stereo network, e.g. GC-Net and PSMNet, with a domain-invariant MS representation based on CBMV features (Batsos et al., 2018). Thus, the training starts from the cost regularization step using conventional matching cost measurement, without exposing the network to the image appearance. Their MS feature expression makes the pre-trained network suffer from marginal accuracy loss in the training domain, however, generalize much better in the unseen target domain.

## 3.5 Datasets

Through the years, a sequence of stereo datasets are available to evaluate the SOTA approaches, for which dozens of stereo pairs with ground truth are sufficient for a fair comparison among competitive methods. The data are acquired based on elaborate campaign planning, precise camera calibration, and expensive ground truth collection with technical support needed using LiDAR, structured light, etc. Besides, each dataset represents a certain situation, e.g. Middlebury datasets (Scharstein and Szeliski, 2002; Scharstein and Szeliski, 2003; Scharstein and Pal, 2007; Hirschmuller and Scharstein, 2007; Scharstein et al., 2014) mainly focus on indoor scenes with different challenging cases for object reconstruction,

KITTI datasets (Geiger et al., 2012; Menze and Geiger, 2015) are more interested in street view imagery to better perceive the scene depth for smart driving, ETH3D dataset (Schöps et al., 2017) contains both indoor and outdoor images to supply a more comprehensive evaluation standard, etc. As the development of modern stereo applications, machine and deep learning are utilized in dense matching for processing high definition stereo data and accomplishing 3D reconstruction with high accuracy and rich details. Accordingly, a huge amount of stereo images with ground truth are required to supervise learning based algorithms, especially for data-hungry end-to-end methods. Thus, some synthetic datasets are proposed (Wulff et al., 2012; Mayer et al., 2016; Dosovitskiy et al., 2017) for an initial learning phase, which could provide thousands of training samples with ground truth disparity maps, without the need of cumbersome data collection and annotation. In this chapter, real and synthetic stereo data are summarized, with a main focus on the commonly used datasets and a brief introduction of less popular or newly proposed data. The described datasets include both indoor and outdoor imagery, covering from close-range, to airborne and satellite data.

### 3.5.1 Real Stereo Data

**Middlebury Data Series:** The Middlebury Stereo Vision project provides the early software platform for an online stereo algorithms evaluation since 2002 (Scharstein and Szeliski, 2002). They also make their first multi-view stereo dataset with ground truth disparity maps public, so that each method contributor is able to pre-test the algorithm before submitting them for evaluation. The dataset consists of six sequences with each of them containing nine images. A digital high-resolution camera is placed on a horizontal translation stage for taking stereo pairs with regular baselines. The images only include piecewise planar objects, e.g. newspapers and posters, for a simple scene composition, which enables the segmentation and direct alignment technique on each planar target (Baker et al., 1998) to estimate the affine motion and the corresponding sub-pixel disparity values for ground truth. In the paper, the limited scene complexity is a compromise for high precision ground truth disparities, which is extended in their following study (Scharstein and Szeliski, 2003) with another two sequences provided, Cones and Teddy, for more complex surfaces. Each sequence also owns nine images taken from viewpoints with equal space in between, from which the view-2 and -6 have precise ground truth disparity maps. To handle the depth perception of more complex objects, the structured light technique is utilized to uniquely label every pixel with special light patterns, so that a dense range map of the scene is obtained. The Middlebury data series is further extended in (Scharstein and Pal, 2007; Hirschmuller and Scharstein, 2007), in order to contain more challenging situations including a larger disparity range, more untextured regions, and radiometric variations. Accordingly, 30 and 6 new datasets are published in each paper, with each dataset containing seven images acquired from equidistant viewpoints along a line. From each viewpoint, three exposure settings are applied under three lighting conditions, leading to nine radiometrically different images in total. The ground truth disparity maps are produced also based on the structured light technique from (Scharstein and Szeliski, 2003), which is improved in (Scharstein et al., 2014) for more precise disparity measurement up to 0.2 pixels, via a robust interpolation of lighting codes. Besides, 33 new datasets are added with much higher resolution (6 megapixels), under different exposure and ambient illumination conditions. The high resolution requires better camera calibration and rectification strategy using bundle adjustment. Meanwhile an "imperfect" version of each dataset is generated to simulate the practical rectification errors. Accordingly, 2D ground truth disparity values are available including y-disparities. The datasets bring a new challenge into the community of stereo matching.

**KITTI Data Series:** As the development of robotics navigation and self-driving, the indoor stereo datasets cannot meet the demands of evaluating depth estimation approaches for



driving scenarios. Geiger et al. (2012) find that many algorithms owning a high rank on, e.g. Middlebury benchmark, struggle to achieve acceptable performance when processing street-view stereo pairs including cars, pedestrians, etc. Compared with the laboratory data collection, the images from outdoor are more likely to involve non-lambertian surfaces, various materials, and challenging lighting conditions, which may increase the difficulty level for dense matching. Therefore, they propose the first realistic non-synthetic stereo and optical flow dataset, KITTI-2012, for driving scenarios, with ground truth disparity maps available to train and evaluate the corresponding approaches. The data is collected with two stereo camera pairs (one grayscale, one color) mounted on a car driving around urban and rural areas, which are synchronized with a laser scanner and a positioning system for ground truth generation. The ground truth disparity maps are acquired by projecting the LiDAR point cloud onto each frame, for which Iterative Closest Point (ICP) is used for data registration. Due to the occlusion during the laser scanning, the ground truth disparity maps are semi-dense. 389 stereo pairs are published, with 194 of them provided together with ground truth for training. The rest is for test. Moreover, KITTI-2012 also includes datasets for 3D visual odometry/SLAM and object detection. In their subsequent research (Menze and Geiger, 2015), a new dataset for 3D scene flow, KITTI-2015, is designed also for autonomous driving. Based on the fact that an outdoor scene can be decomposed into a static background and a sequence of moving objects, 400 KITTI raw images are annotated to create a dynamic scene, for which the background is firstly reconstructed with all moving objects removed, and then elaborate CAD models are added into the frame as dynamic objects. The reason for not directly annotating the scene flow with the LiDAR points, originates from the hardware limitation that the rolling shutter and the low frame rate of the used laser scanner cannot support the flow measurement for all the moving objects. Hence, the 3D CAD models of 16 representative vehicles are selected and sampled as point clouds to be fitted into each frame. KITTI-2015 provides half of the 400 stereo images with ground truth as training data, with the other half to test the SOTA optical flow and scene flow methods.

ETH3D Data: Schöps et al. (2017) contribute their ETH3D stereo datasets, containing both indoor and outdoor views. The imaging targets include man-made buildings and natural scenes with vegetation, which are rarely considered in previous data. 47 low resolution two-view stereo pairs are provided with 27 of them for training and the rest for testing with ground truth withheld. For some scenes, the images are in two different sizes, one of which owns a larger FOV to fully cover the other. The images are collected via a Digital Single-Lens Reflex (DSLR) camera synchronized with a multi-camera rig capturing varying FOVs. The ground truth is obtained using a high precision laser scanner. In addition to the binocular stereo data, 25 high resolution multi-view stereo pairs and 10 low-resolution many-view stereo video data are also publicly available. Since various scene categories are included in the datasets, ETH3D expects a robust stereo algorithm to reduce overfitting.

EuroSDR Image Matching Benchmark: Triggered by the development in the field of dense matching and the modern elevation data collection from advanced airborne imaging sensors, an aerial image matching benchmark project (Haala, 2014) is launched by the European Spatial Data Research Organisation (EuroSDR). The project mainly aims at setting a public platform to evaluate dense matching algorithms for airborne stereo data processing and DSM generation. Thus, the potential of the normally used photogrammetric softwares can be assessed according to the quality of generated 3D products. Two test areas from, Vaihingen/Enz and Munich, are covered to include different landuse and ground geometry. The Vaihingen/Enz subset concentrates on a semi-rural region, with the flight height set as 2900m and the GSD as 20cm. For stereo reconstruction, the image pairs acquire from one folded to nine folded overlap, leading to 63% in flight and 62% cross flight overlap. The other subset from Munich is a representative urban area, in which the central part of the city in size of 1.5km  $\times$  1.7km is covered. Higher resolution is obtained with the GSD as 10cm. An overlap up to fifteen folded regions is achieved, from which the overlap for both

in flight and cross flight directions is 80%.

**Urban Semantic 3D (US3D) Data:** To assist the previously mentioned multi-task learning models which combine semantic and stereo information for better joint optimization, Bosch et al. (2019) propose their US3D dataset including both semantic and disparity labels. The dataset is composed of WorldView-3 target-mode panchromatic and VNIR images, with 26 of them collected between 2014 and 2016 over Jacksonville, Florida and 43 collected between 2014 and 2015 over Omaha, Nebraska. The GSD of panchromatic and VNIR data is 30cm and 1.3m, respectively, covering around 100 square kilometers area. Both the stereo and semantic ground truth are generated based on LiDAR products published by the Homeland Security Infrastructure Program (HSIP) (Brown et al., 2018). US3D contains multi-view and multi-band satellite imagery, which promotes the development of pairwise semantic stereo.

### 3.5.2 Synthetic Stereo Data

Since Dosovitskiy et al. (2015) have proven that the optical flow estimation problem can be solved via a deep neural network based on supervised learning, many end-to-end neural networks are proposed for stereo reconstruction, the performance of which is largely dependent on the quality and quantity of available training data. As the real stereo data is often in small amount with a specific situation concerned, a large synthetic dataset using automatic ground truth annotation is promising, to thoroughly train a model from scratch and reduce the requirement of real training data for finetuning.

**Scene Flow Data Series:** Mayer et al. (2016) propose the first large scale synthetic stereo video datasets with accurate ground truth for scene flow estimation, which own sufficient realism and variation to properly model the real-world situations, and provide a huge amount of data to supervise deep networks. Three subsets are customized using a 3D creation suite, Blender, named as FlyingThings3D, Monkaa, and Driving, including stereo color images with bidirectional ground truth of disparity and optical flow. In addition, the disparity change, motion and object boundaries are also available. FlyingThings3D is rendered using everyday objects sampled from ShapeNet (Savva et al., 2015), which randomly fly over static background constructed by choosing from 200 shapes of cuboids and cylinders. Around 25000 stereo pairs are created, with a main focus on randomness to realize a large variety for training deep neural networks, instead of concentrating on a specific task with rigid rules. Monkaa is essentially a short film, from which nonrigid motions are simulated to generate stereo pairs. Besides the selected film scenes, the virtual camera's orientation is adjusted to manually create new scenes for more animated images. Driving is designed to mimic KITTI datasets, with naturalistic street scenes from a driving car's viewpoint. The models of cars, trees and street lights are arranged in an appropriate manner, so that comparable scene settings to KITTI is obtained. The emergence of Scene Flow datasets promotes the development of deep learning based stereo methods to a great extent. Most of the supervised algorithms described in this chapter pre-train the models using the datasets.

**MPI-Sintel Flow Data:** The data originates from an open source CGI movie, Sintel, containing various image sequences degraded by motion, defocus and atmospheric blur for more realistic appearance (Wulff et al., 2012). There are 1628 frames in total (564 frames for test), collected from 35 scenes, which enable appropriate supervision for flow estimation. Regarding the ground truth, the camera intrinsic and extrinsic parameters, scene depth information, occlusion masks and motion boundaries are all available.

**CARLA Simulator:** CARLA (Dosovitskiy et al., 2017) is an open source platform which simulates autonomous driving scenes, with a variety of sensor suites and environment conditions to be selected by users. Numerous urban elements could be added including vehicles,

buildings, traffic signs, etc. In order to better train the driving scenarios, the speed and acceleration of cars can be modeled to animate different driving situations. In addition, the setting of illumination and weather conditions is possible. This engine can be used to create multiple virtual cases, enabling deeper research in self-driving.

Recently, more and more stereo datasets are proposed to assist the development of learning based matching algorithms. As for the real data, the ApolloScape dataset contains many more frames and labels for self-driving, compared with the popular KITTI data series (Huang et al., 2020). Geyer et al. (2020) release the Audi Autonomous Driving Dataset (A2D2), with both stereo and semantic information provided in the form of images and point clouds. Regarding the synthetic data, Li and Snavely (2018) utilize unlimited multi-view internet photos to generate a huge stereo dataset, MegaDepth, for monocular depth estimation. Yang et al. (2019) create a synthetic dataset together with a real dataset. The synthetic one, High-res virtual stereo (HR-VS), is simulated using the CARLA environment, with the images in size of  $2056 \times 2464$ . The real one, High-res real stereo (HR-RS), is collected using stereo cameras and LiDAR from an urban driving campaign, acquiring  $1918 \times 2424$  images and the corresponding ground truth. Both datasets are proposed for developing high resolution dense matching methods, with the target on autonomous driving.

## 4 Summary of the Contributions

In order to promote the development of dense matching, certain strategies are proposed in the thesis to optimize the stereo framework, in form of three published peer-reviewed journal papers. Regarding the aforementioned objectives, the stereo matching pipeline is improved from the matching cost computation, cost aggregation, to an overall enhancement as an end-to-end neural network. In this chapter, our contributions are summarized.

### 4.1 Self-Supervised Matching Cost Network with Case Study: Plant Reconstruction (ForDroughtDet)

Dense image matching is widely used in computer vision to construct high-quality geometric models, which recovers three-dimensional (3D) information from two-dimensional (2D) images. Among different techniques, SGM outperforms most existing classical approaches in accuracy and efficiency, which is widely applied in diverse scenarios such as building reconstruction, DSM generation, robot navigation, and self-driving, etc. (Hirschmüller, 2011; Kuschik et al., 2014; Qin et al., 2015). However, the performance of SGM varies with different matching cost calculation methods adopted. Along with the development of machine/deep learning (Lecun et al., 1998), supervised algorithms for learning the pixel similarity are proposed with better performance to extract feature from pixels and compute the matching cost. Among them, Matching Cost based on CNN (MC-CNN) (Zbontar and LeCun, 2016) belongs to the first generation. Composed of a simple architecture, MC-CNN is trained on pairs of small image patches with known true disparity, and outputs a similarity measure of pixels which is exploited by SGM for the data term. Thanks to a good extraction of the local image features and a trained similarity measure to compare the extracted feature descriptors, the integration of MC-CNN and SGM proven to outperform most previous algorithms. However, the ground truth collection is a bottleneck for neural network-based algorithms, considering that huge amount of labeled data is required to train the net (Krizhevsky et al., 2012; Knöbelreiter et al., 2018). For example, ground truth acquisition via LiDAR sensors is complicated in remote sensing. It could be difficult to capture a temporally-consistent dense point cloud as the reference data of stereo images, from a dynamic scenes in practice. Inspired by the work of Knöbelreiter et al. (2018), we propose a dense matching strategy combining SGM and a self-trained MC-CNN.

#### 4.1.1 Background

Dense matching attempts at establishing correspondences between every pixel in the image pair (Scharstein and Szeliski, 2002). Together with the known camera orientations, a dense point cloud can be obtained. Most dense stereo matching algorithms consist of four steps: Firstly, a similarity measure between two potentially matching pixels is computed to evaluate the matching cost. As the matching cost can be ambiguous, costs are then usually aggregated in a local neighborhood. Global stereo methods apply certain regularization to the aggregated costs, while local methods simply select the correspondence with the lowest matching cost. SGM combines local and global methods by regularizing the aggregated costs before determining each correspondence. Afterwards, for rectified stereo pairs, a disparity map containing the horizontal shifts between the images is obtained (Bolles et al., 1987; Okutomi and Kanade, 1993). Finally, sub-pixel interpolation, left-right consistency check, and outlier filtering are included in the post-processing by most stereo algorithms for disparity refinement.

On the other hand, CNNs (Lecun et al., 1998) have been used to solve several vision problems such as classification (Krizhevsky et al., 2012), recognition (Lawrence et al., 1997), etc. It is basically a feedforward artificial neural network constructed by a sequence of layers with learnable weights and biases. A volume of activations is transformed from one into another when going through the layers, and finally certain scores are obtained as output at the end of the network, which could be e.g. class scores for classification. Four types of layers are frequently used: (a) convolutional layers, in which each neuron is related to a local region of the input; (b) pooling layers, used to downsample the previous volume and enlarge the receptive field; (c) rectified linear units applying an element-wise activation function; and (d) fully-connected layers, which calculate the output by connecting each neuron to all the neurons of the previous volume for high-level reasoning. The network can be trained to reach its best performance when a sufficient amount of training samples are available.

### 4.1.2 Dense Matching based on MC-CNN and SGM

CNNs provide a new possibility in dense matching (Luo et al., 2016; Zbontar and LeCun, 2016). Zbontar and LeCun (2016) proposed a dense stereo algorithm using a CNN based matching cost measurement combined with SGM and additional post-processing steps, which outperformed most previous stereo matching algorithms. Specifically regarding the data term, a binary classification data set is constructed for training the net, based on either the KITTI (Geiger et al., 2012; Menze and Geiger, 2015) or the Middlebury (Scharstein and Szeliski, 2002; Scharstein and Szeliski, 2003; Hirschmuller and Scharstein, 2007; Scharstein and Pal, 2007; Scharstein et al., 2014) stereo data sets with available ground truth disparity maps. At each image location, a positive and a negative training example are extracted. The positive example is a pair of patches from the left and right image, respectively, with the central pixels projected from the same object point, while the negative example is from a pair of patches where this geometric condition is not satisfied.

Two network architectures are designed and trained on the extracted training examples. Both of them are Siamese networks with two sub-networks sharing the same weights (Bromley et al., 1993). The sub-network consists of several convolutional layers, each of which is followed by a rectified linear unit. Thus, each input image patch can be transformed into a feature vector. Afterwards, the rest of the network computes the similarity measure using the feature vectors. For that, two networks variants are proposed as shown in Figure 4.1. The first architecture uses the dot product of the normalized feature vectors as similarity measure. Therefore, it has a lower runtime and named as fast architecture. The second one is designed for more accurate matching cost calculation, which learns the similarity measure during training. The extracted feature vectors are concatenated and passed through a number of fully-connected layers with a rectified linear unit following each of them. At the end, a fully-connected layer followed by a sigmoid nonlinearity is used to produce the similarity score. In this research, the accurate architecture is adopted considering its better performance.

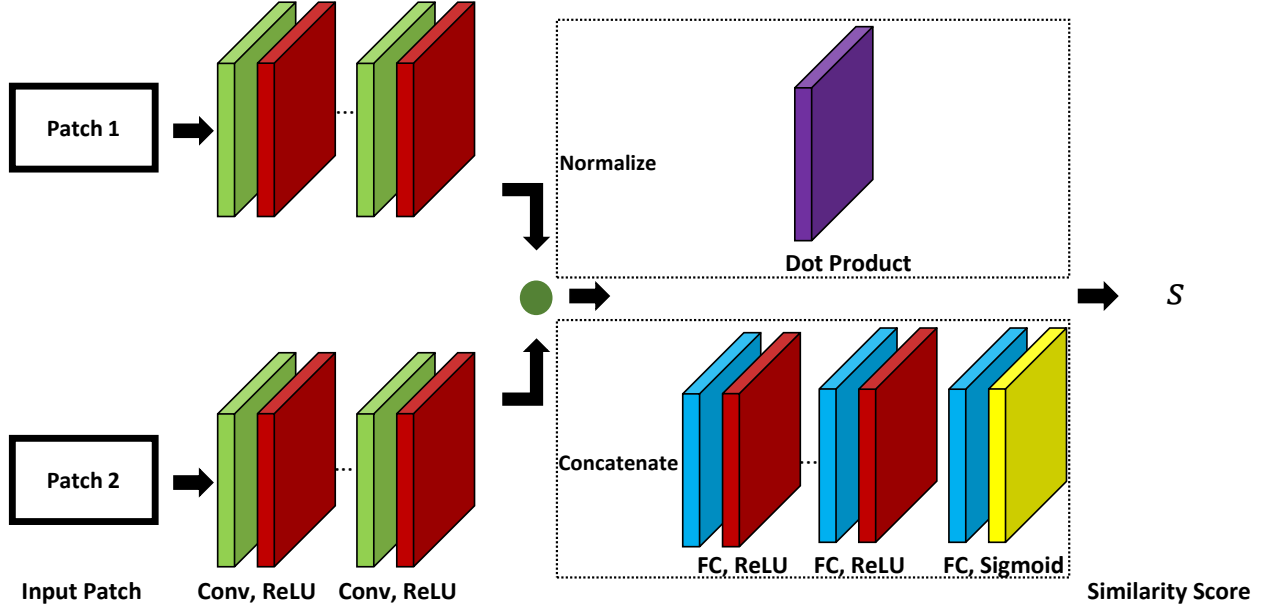


Fig. 4.1. The architecture of MC-CNN. With the same sub-networks, composed of a series of convolutional layers and rectified linear units, a feature vector can be generated for each of the input image patches. Then, a similarity score is computed at the end, either simply based on a dot product of the normalized feature vectors or another sub-network to learn the similarity measure during training. The latter architecture achieves better accuracy at the cost of relatively higher complexity.

Regarding the training, the binary cross-entropy loss is used as defined in

$$l = t \cdot \log s + (1 - t) \cdot \log(1 - s). \quad (4.1)$$

$l$  is the calculated loss, and  $s$  is the similarity score from the output of the net. The value of  $t$  depends on the category of the training example being used, which is equal to 1 for positive examples and 0 for negative ones.

Zbontar and LeCun (2016) acquire the hyperparameters of MC-CNN based on manual search, which include the number of convolutional layers in each sub-network (5), the number of feature maps in each layer (112), the convolutional kernel size (3), the number of fully-connected layers (3), the corresponding number of units in each full-connected layer (384), and the input patch size ( $11 \times 11$ ). The same hyperparameter setting is applied in this research.

After the CNN based matching cost computation, SGM is exploited to regularize the disparity estimation based on a piece-wise constant smoothness term. As mentioned in Chapter 2.1.2, SGM is a combination of local and global stereo methods (Hirschmüller, 2008) and approximates a global two-dimensional smoothness term by summation of one-dimensional smoothness constraints along 8 or 16 directions. For each direction, assuming the target pixel is at location  $p$ , the cost is computed as:

$$L_r(p, d) = C(p, d) + \min(L_r(p - r, d), L_r(p - r, d - 1) + P_1, L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2). \quad (4.2)$$

in which  $L_r(p, d)$  is the cost along the path traversed in direction  $r$  assuming  $d$  as the disparity.  $C(p, d)$  is the matching cost according to the output of MC-CNN.  $P_1$  is a penalty term added when the previous pixel has a disparity difference of 1. A larger value  $P_2$  is utilized to penalize larger disparity differences. For each pixel  $p$ ,  $S(p, d) = \sum_r L_r(p, d)$  is computed and the disparity with the minimum  $S$  is selected as the result using WTA strategy.

SGM is selected as the smoothness term in both Zbontar and LeCun (2016) and this research, thanks to its good performance and efficiency. The runtime is proportional to the reconstructed volume (d’Angelo and Reinartz, 2012; d’Angelo, 2016). It should be noted that,  $C(p, d)$  is calculated using MC-CNN and aggregated based on Cross-Based Cost Aggregation (CBCA) (Mei et al., 2011). Then,  $S(p, d)$  undergoes CBCA once more before the final disparity determination. Referring to Zbontar and LeCun (2016) and Mei et al. (2011), some post-processing steps are implemented to refine the quality of the disparity map, including interpolation, sub-pixel enhancement, a median filter, and a bilateral filter.

The following sections provide the details for training MC-CNN. Two schemes are designed, of which one utilizes the ground truth from a LiDAR scanner to construct training data. The other scheme is a self-training strategy, that directly uses the dense matching results of MC-CNN, pre-trained on the Middlebury datasets, to retrain the network. The performance of the two schemes is compared.

### 4.1.3 Training Strategy with Ground Truth

Zbontar and LeCun (2016) offer several models pre-trained on the KITTI 2012, KITTI 2015, and Middlebury datasets, respectively. As one option, we start from the pre-trained net on the Middlebury datasets which focus on static objects, and further train the net using our own data with ground truth from LiDAR. Thus, the network is finetuned before testing on our data. Also, the learning ability of MC-CNN for objects from a different category could be tested.

As for the LiDAR scanning, a point cloud is generated to obtain the ground truth disparity maps for our experiment images. As the image orientation and the point cloud use different coordinate systems, a co-registration step is needed before the point cloud can be used. Besides, the main target is to test the performance of MC-CNN trained with different strategies and compare with the classic matching cost algorithm Census to demonstrate the effectiveness of MC-CNN. Hence as shown in Figure 4.2, we first generate the disparity maps of the experiment images, based on SGM either with Census or MC-CNN pre-trained on the Middlebury data sets. A pixel-wise average of the two maps is computed and projected into 3D space to obtain a point cloud. Then, the point cloud from the laser scanner is registered to this newly generated point cloud. The ground truth disparity map is obtained by projecting the registered laser scanning point cloud onto the epipolar image planes. We use CloudCompare (Girardeau-Montaut et al., 2005) to roughly align the two point clouds first, by scale matching, rotation, translation, and manual point pair picking alignment. After the rough alignment, the objects that are reconstructed well by both dense matching and LiDAR, and aligned close to each other already, are selected for a further fine registration based on the Generalized Iterative Closest Point (GICP) method (Segal et al., 2009). GICP is more robust and performs better than the standard ICP without loss of efficiency. Afterwards, only well-registered objects are kept to generate the ground truth.

### 4.1.4 Training Strategy without Ground Truth

More and more data are gradually available to meet the need of CNN for training. However, in most cases, high performance is accomplished at the cost of substantial pre-processing workloads to label the training examples. Therefore, many self-supervised concepts have been proposed to avoid the time-consuming manual annotation (Joung et al., 2017; Zhou et al., 2017; Knöbelreiter et al., 2018). Joung et al. (2017) exploited the correspondence consistency between stereo images to pick samples during the training and guide the network to compute matching cost. Zhou et al. (2017) randomly initialized a network and adopted left-right consistency check to select suitable matching to train the net. Knöbelreiter et al. (2018)

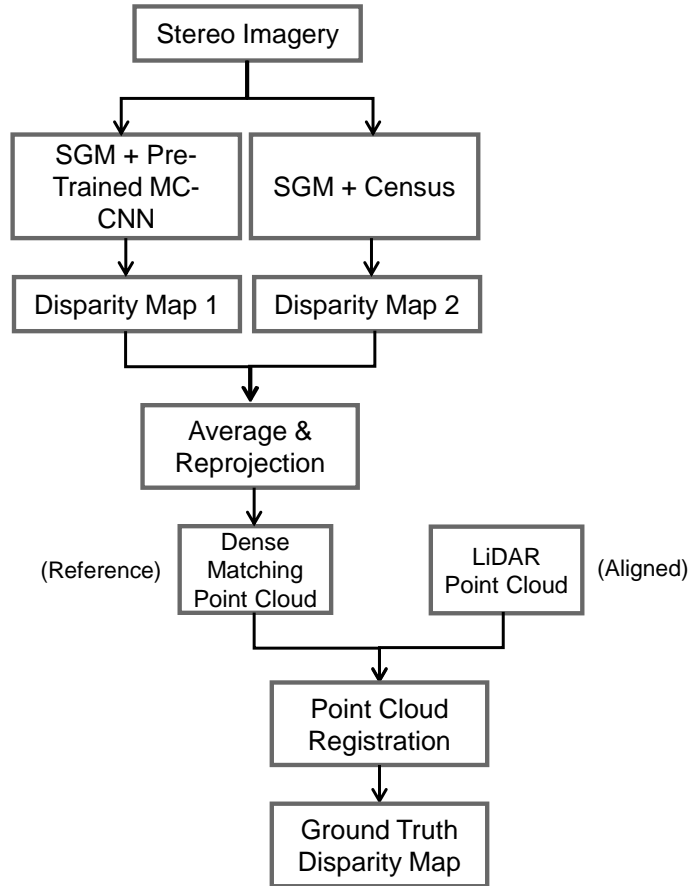


Fig. 4.2. The flow chart for ground truth disparity map generation using LiDAR point cloud. Starting from the experiment stereo images, the disparity maps are generated using SGM with Census and MC-CNN pre-trained on the Middlebury data sets, respectively. Afterwards, a pixel-wise average of the two maps is computed, and projected into the object space to obtain a point cloud. The laser point cloud is registered to this newly generated point cloud. Thus, the ground truth disparity map is acquired via projecting the registered LiDAR point cloud onto the epipolar image planes.

constructed the training data using a pre-trained version of their hybrid CNN-Conditional Random Fields (CRF) model followed by a conservative consistency check to reject most outliers. Based on that, their self-supervised network is able to improve the completeness and accuracy of the stereo reconstruction results on aerial imagery.

High-resolution LiDAR point clouds are very difficult and expensive to capture especially in an outdoor environment. It could be challenging to obtain perfectly matching image and LiDAR data considering the long scanning time and the dynamic attributes of objects in practice. Therefore, instead of using LiDAR data, a self-training procedure is applicable even to scenarios where ground truth acquisition is difficult or impossible. We use the MC-CNN pre-trained on Middlebury data, to generate disparity maps used for self-training. A left-right consistency check with a threshold of 1 pixel is used to filter most outliers as:

$$|d_p^L + d_q^R| \leq 1, \quad q = p - d_p^L. \quad (4.3)$$

In Equation 4.3,  $d_p^L$  is the disparity for pixel at location  $p$  in the disparity map regarding the left epipolar image as the master epipolar plane, while similarly  $d_q^R$  is calculated via dense matching regarding the right epipolar image as the reference epipolar plane. Only pixels where left-right matching differs by less than 1 pixel are used as ground truth to further train MC-CNN. The detailed procedure is shown in Figure 4.3



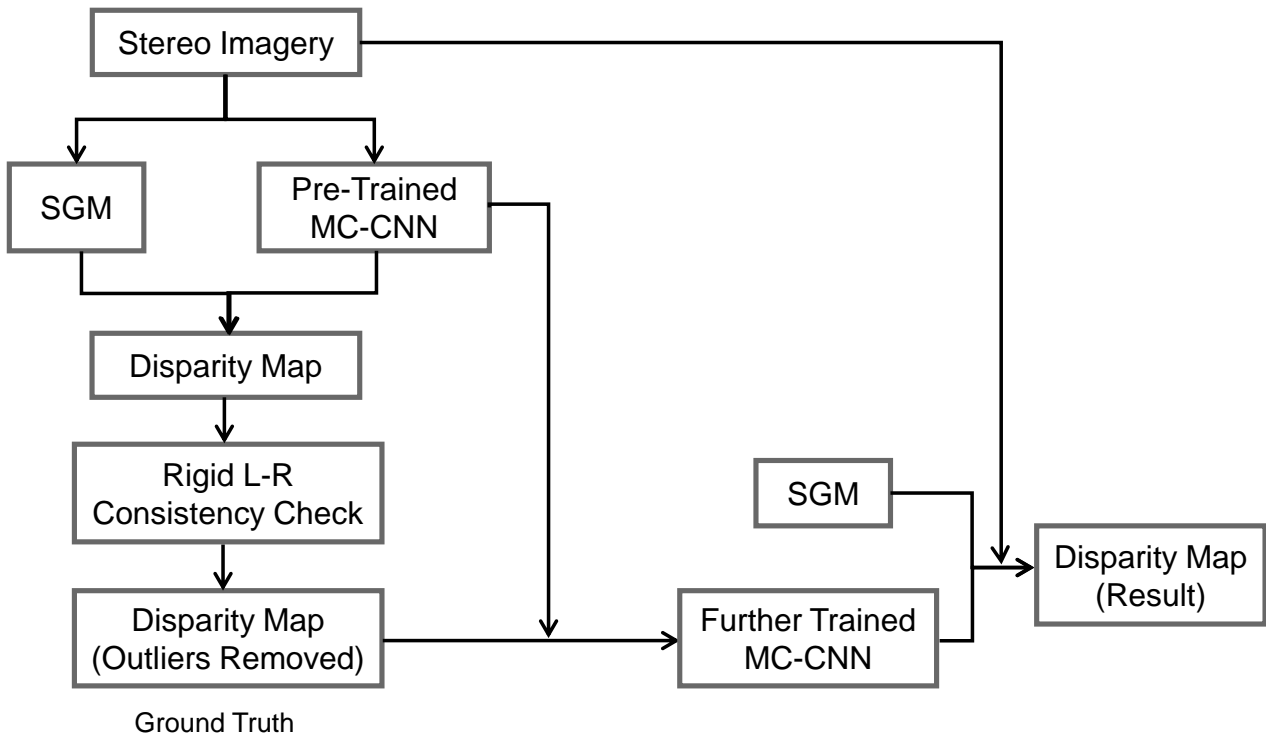


Fig. 4.3. The flow chart for the self-training strategy. Based on SGM and a pre-trained MC-CNN on Middlebury datasets, a disparity map is generated. Afterwards, a rigid left-right consistency check is applied to remove most outliers. Only the pixels left are regarded as accurate estimation (artificial ground truth) to further train the MC-CNN model, which is finally used to predict the disparity results.

#### 4.1.5 Case Study: Plant Reconstruction and Drought Detection

In this section, the proposed self-training strategy is validated on two experiments, aiming at reconstructing the 3D shape of a complex object, plants. The stereo matching remains very difficult, especially for leaves, due to the lack of unique features, many occlusions, and repetitive structure. Thus, the feasibility of self-trained MC-CNN is demonstrated. The first experiment was carried out in an indoor laboratory environment. In this experiment, an 8-meter high tree standing in the atrium of a building was photographed from above. At the same time, a LiDAR point cloud was captured from a similar position. The second experiment investigated stereoscopic images from the crown of a beech tree growing in a typical European forest.

The main objective of this work is the three-dimensional reconstruction of trees and their leaves in the forest. In order to minimize the influence of environmental conditions, the first experiment investigates an 8-meter high deciduous tree inside a building. A digital high-resolution handheld camera (NIKON D5500) equipped with an 18 mm lens is used to acquire images from a bridge over the crown of the tree. An exposure time of 1/20 seconds and an ISO speed rating of 400 was used. The acquired images are 4000 pixels in height and 6000 pixels in width. A stereo image pair with a baseline length of approximately 0.1 meters is taken from a distance of approximately 1 meter from the tree. Details about the image acquisition are available in Table 4.1. A Leica HDS7000 laser scanner is used to obtain a point cloud of the plant from a similar position. Capturing the point cloud with a point distance of 6.3 mm and a depth error of 0.4 mm RMS at a distance of 10 meters took about 10 minutes.

The proposed dense matching approach requires epipolar images, where corresponding pixels are located on the same image row. MicMac (Rosu et al., 2015) was utilized for camera calibration, relative orientation and epipolar image rectification. The epipolar images generated based on the stereo pair mentioned above are shown in Figure 4.4.

Table 4.1. The image acquisition parameters.

<b>Camera model</b>	NIKON D5500
<b>Height</b>	4000 pixels
<b>Width</b>	6000 pixels
<b>Exposure time</b>	1/20 sec
<b>ISO speed rating</b>	400
<b>Focal length</b>	18.0 mm
<b>Object distance</b>	~ 1 m
<b>GSD</b>	0.02 cm/pixel
<b>Baseline length</b>	~ 0.1 m



Fig. 4.4. The epipolar image pair for the first experiment. MicMac was utilized for camera calibration, relative orientation and epipolar image generation.

Disparity maps have been calculated using the strategies described above with four different matching costs:

- ◇ Census: Using Census as matching cost;
- ◇ MC-CNN-Pre: Using MC-CNN matching cost pre-trained on the Middlebury data sets;
- ◇ MC-CNN-LiDAR: Using MC-CNN further trained on the LiDAR ground truth for matching cost, as described in Chapter 4.1.3;
- ◇ MC-CNN-SelfT: Using MC-CNN further trained using the disparity maps of MC-CNN-Pre, as described in Chapter 4.1.4.

After the processing as described in Chapter 4.1.2 and applying the left-right consistency check as described in Chapter 4.1.4, the generated disparity maps for the epipolar image pair in Figure 4.4 are shown in Figure 4.5. For pixels with valid matching, the calculated disparity values from -91 to +42 are represented by the color from blue to yellow accordingly.

It should be noted that, the training and evaluation of the different methods are hampered by systematic differences between LiDAR and stereo pairs. Due to the automatic air conditioning of the building, there were small movements of the branches and leaves during LiDAR recording which took around 10 minutes. These led to slightly different leaf positions between LiDAR and stereo images. During the generation of the ground truth disparity map, some errors are included unavoidably when picking up point pairs to align the point clouds initially. The fine registration with GICP can improve the co-registration but errors still exist. Due to these problems, the point cloud registration is not perfect which influences the use of the ground truth disparity map generated from the LiDAR data. This is also the reason that we determine to only focus on some selected objects (leaves) after rough align-

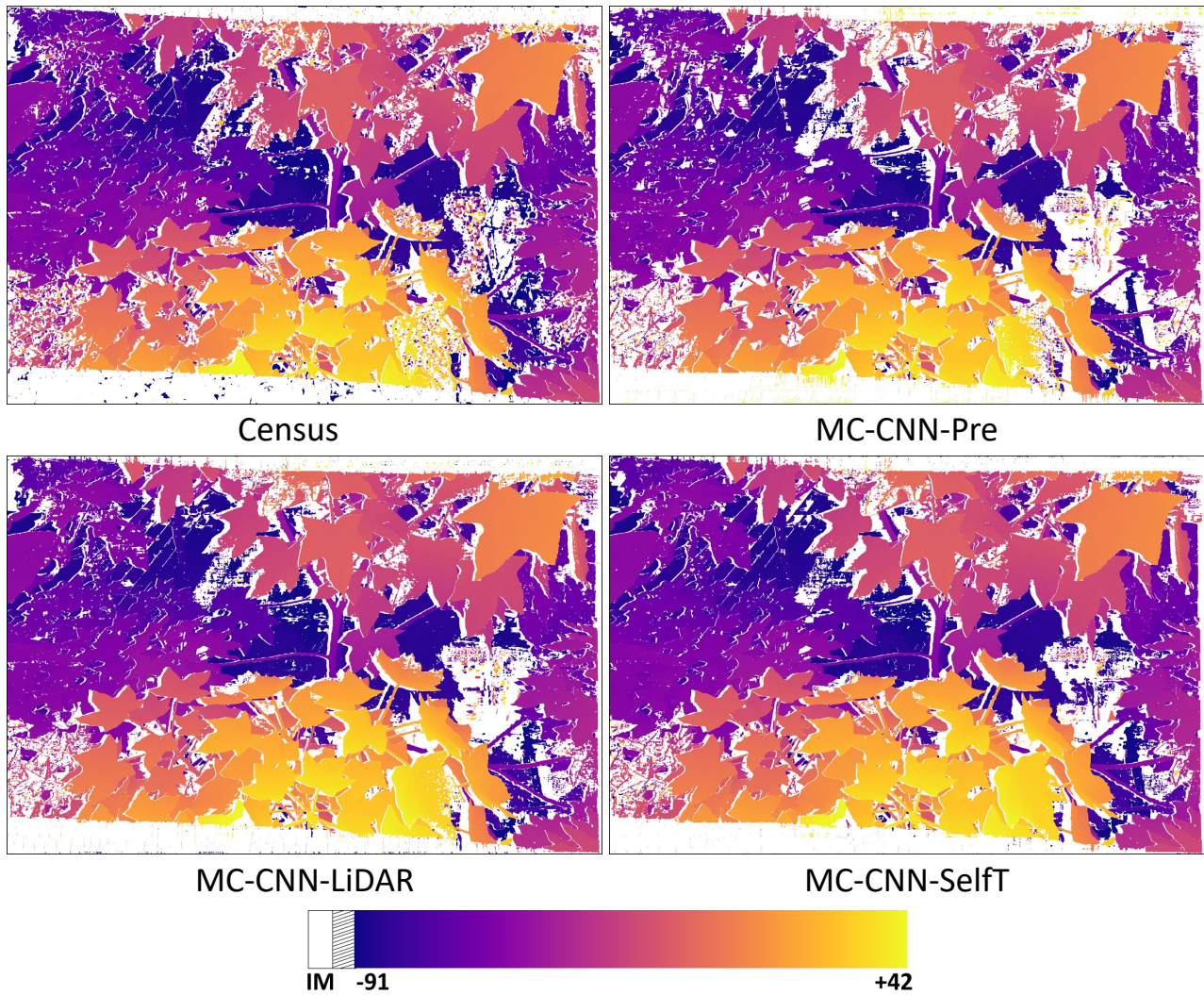


Fig. 4.5. The disparity maps generated based on SGM with different strategies, Census, MC-CNN-Pre, MC-CNN-LiDAR and MC-CNN-SelfT for matching cost. Inconsistent matching (IM) is represented by the color white.

ment to do GICP, as mentioned in Chapter 4.1.3. Afterwards the relatively well registered leaves by GICP, that visually show merely small shift between the point clouds, are utilized for training and evaluation of the methods, which alleviates the problem mentioned above. This is in accordance with our application, as the shape of the leaves is the major indicator of plant drought condition. Compared with images from the Middlebury data sets with sizes of around  $300 \times 200$  to  $3000 \times 2000$  pixels, our images are larger ( $6000 \times 4000$  pixels), and the masked leaves can still provide a good amount of application specific training data. Thus, we use 13 well registered leaves together with Jadeplant and Sword1 data (containing a plant, belonging to the Middlebury datasets 2014) as training data. The reason for adding the Middlebury data into the newly generated data sets is to increase the amount of training data from limited selected leaves.

A visual comparison of the results in Figure 4.5 shows that the tree was well reconstructed by all matching schemes. The results of five independent leaves not used during training on the LiDAR ground truth are shown in Figure 4.6. While most parts of the leaves are well reconstructed, some differences in completeness and amount of outliers are visible.

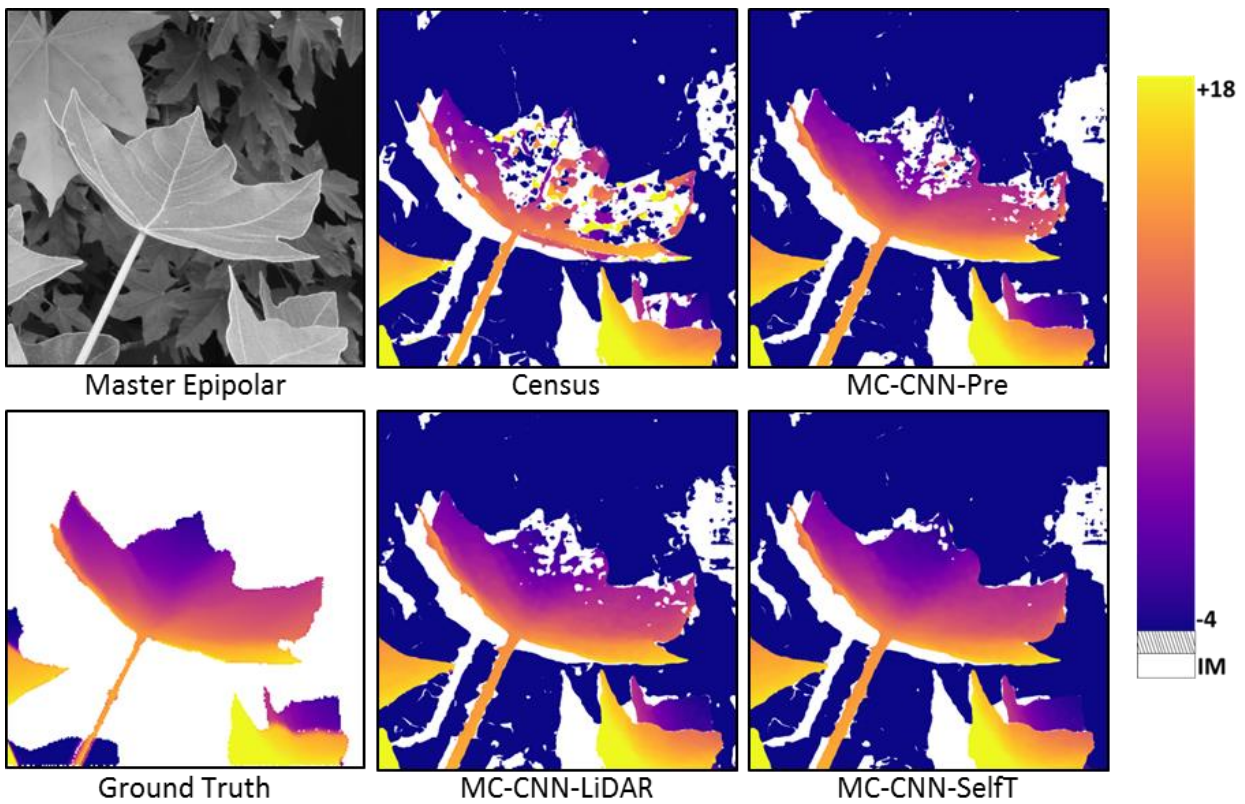
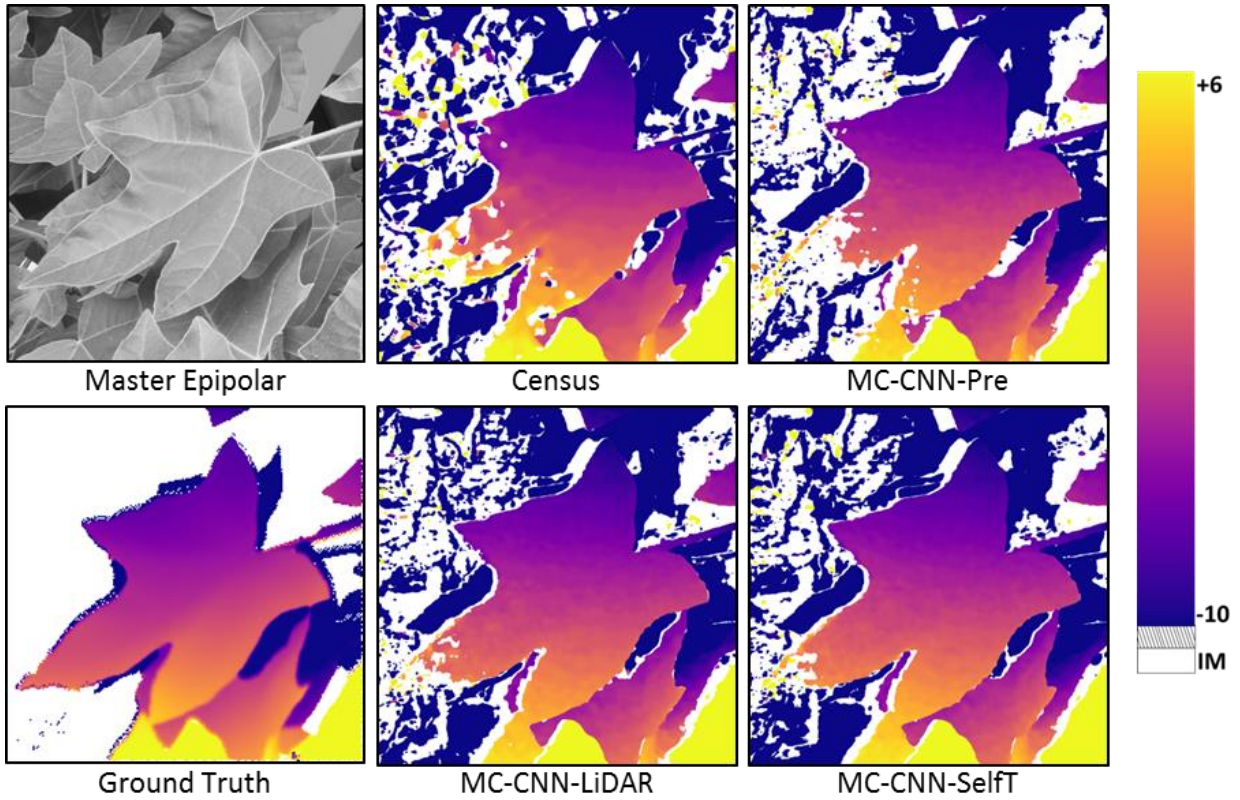
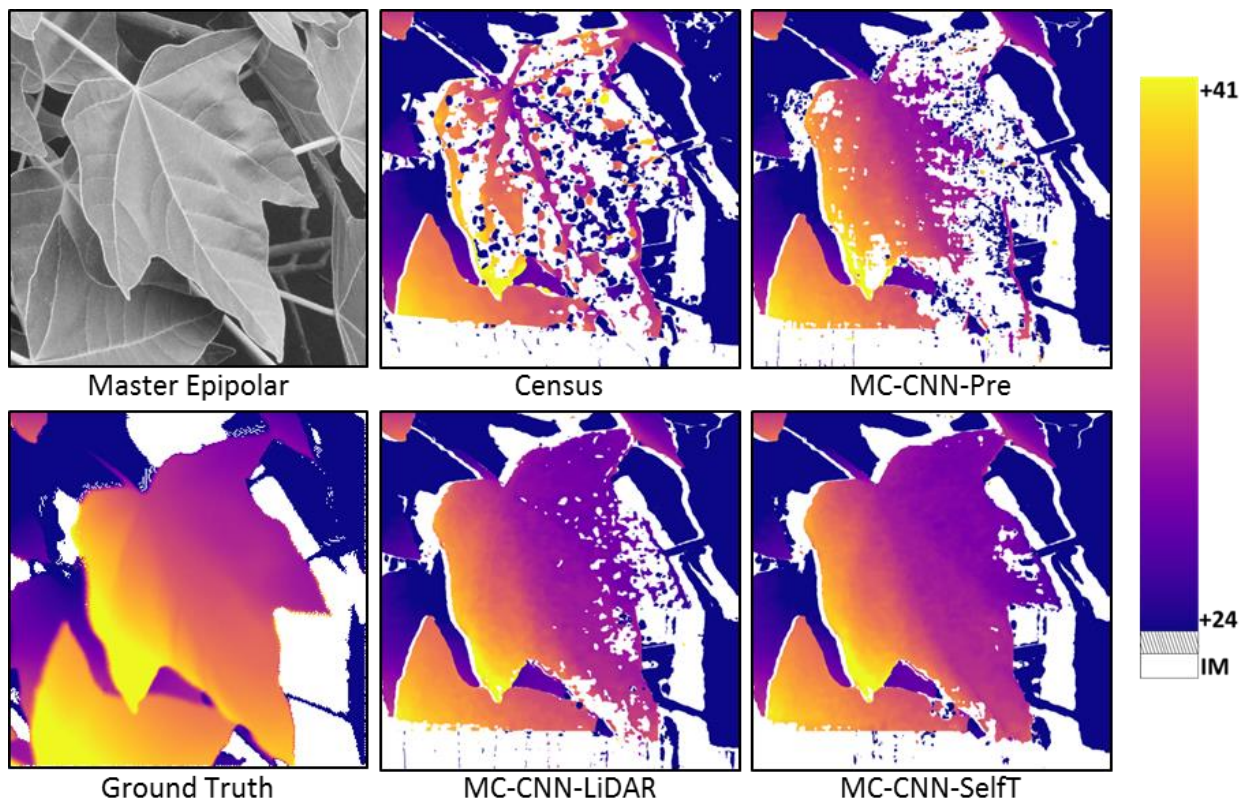
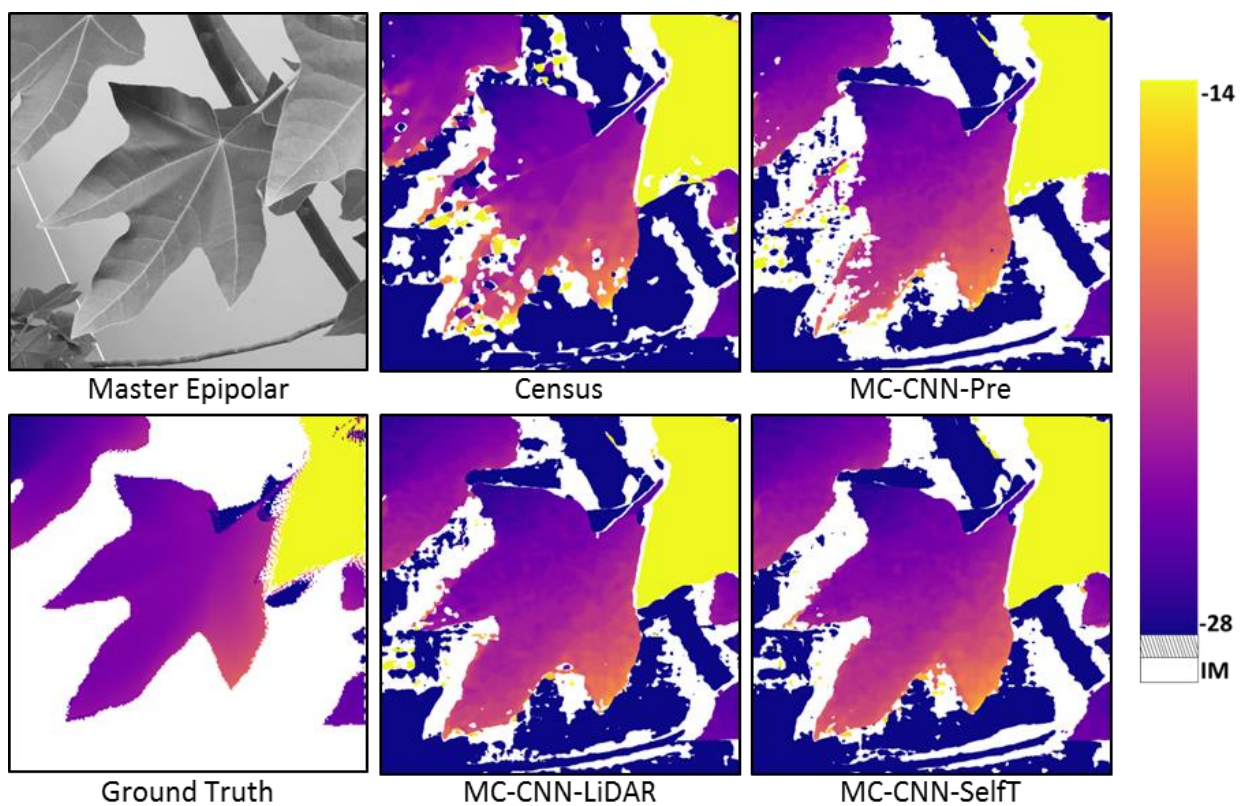


Fig. 4.6. The reconstruction details of several selected leaves. From left to right in each subset: the first row includes the master epipolar image and disparity maps for Census and MC-CNN-Pre. The second row includes the ground truth and disparity maps for MC-CNN-LiDAR and MC-CNN-SelfT. In order to enhance the contrast of the disparity within each single leaf, we have used a different color bar for each leaf. Pixels invalidated by the left-right check are shown in white.

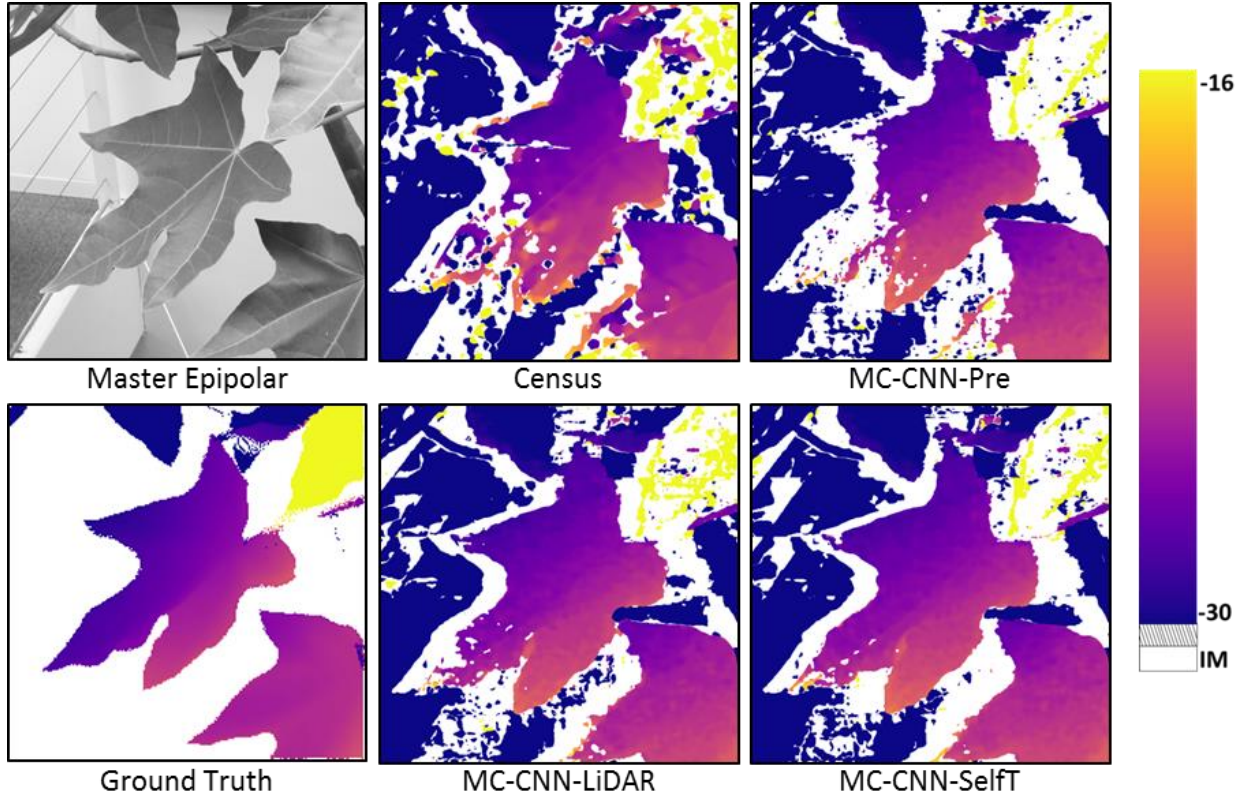


(c) Leaf c



(d) Leaf d

Fig. 4.6. The reconstruction details of several selected leaves. From left to right in each subset: the first row includes the master epipolar image and disparity maps for Census and MC-CNN-Pre. The second row includes the ground truth and disparity maps for MC-CNN-LiDAR and MC-CNN-SelfT. In order to enhance the contrast of the disparity within each single leaf, we have used a different color bar for each leaf. Pixels invalidated by the left-right check are shown in white. (cont.)



(e) Leaf e

Fig. 4.6. The reconstruction details of several selected leaves. From left to right in each subset: the first row includes the master epipolar image and disparity maps for Census and MC-CNN-Pre. The second row includes the ground truth and disparity maps for MC-CNN-LiDAR and MC-CNN-SelfT. In order to enhance the contrast of the disparity within each single leaf, we have used a different color bar for each leaf. Pixels invalidated by the left-right check are shown in white. (cont.)

From a visual inspection, it is found that the disparity values obtained by all four strategies match with the ground truth. With Census as matching cost, the main shape of the leaf is reconstructed but with considerable noise and low completeness. MC-CNN-Pre results in low completeness, cf. leaf (e), but shows less noise. However when fed with specific data for further training, MC-CNN-LiDAR and MC-CNN-SelfT achieve higher reconstruction completeness. MC-CNN-SelfT results in a slightly better leaf reconstruction than MC-CNN-LiDAR and fewer gaps. We would like to point out two reasons for this behavior: Firstly, in self-training more training samples are available for the net to develop the ability to learn new feature and calculate the similarity score. In Figure 4.5, it can be seen that all leaves are reconstructed or partially reconstructed in MC-CNN-Pre, which can possibly be used in MC-CNN-SelfT compared with only a few leaves used in MC-CNN-LiDAR. Hence, the further trained MC-CNN can learn from each single leaf during the training and recover more area. Besides, the rigid left-right consistency check, applied to the dense matching results of MC-CNN-Pre to construct training samples, guarantees a reasonable training procedure for MC-CNN-SelfT.

In addition to the visual comparison above, a quantitative evaluation is performed by comparing the generated disparity maps with the disparity maps obtained from LiDAR. The leaves a - e shown above are used for comparison. Firstly, the disparity difference  $D_p$  is calculated as below in units of pixels:

$$D_p = d_p - d_p^G \quad p \in N_p, \quad (4.4)$$

where  $d_p$  denotes the disparity value of a pixel at location  $p$  calculated using one of the four dense matching schemes.  $d_p^G$  is the corresponding ground truth disparity value.  $N_p$  is the set of pixels where both dense matching and ground truth provide disparity values. The mean ( $D_{\text{mean}}$ ), median ( $D_{\text{median}}$ ), standard deviation ( $D_{\text{STD}}$ ) and median absolute deviation ( $D_{\text{MAD}}$ ) of the disparity differences are computed for comparison. The results are reported in Tables 4.2, 4.3, 4.4 and 4.5.

$$D_{\text{mean}} = \text{mean}(D_p). \quad (4.5)$$

$$D_{\text{median}} = \text{median}(D_p). \quad (4.6)$$

$$D_{\text{STD}} = \sqrt{\text{mean}\left(\left(D_p - D_{\text{mean}}\right)^2\right)}. \quad (4.7)$$

$$D_{\text{MAD}} = \text{median}\left(\left|D_p - D_{\text{median}}\right|\right). \quad (4.8)$$

Table 4.2. Mean of the disparity difference between dense matching and ground truth.

Leaf	$D_{\text{mean}}$ (pixels)			
	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
a	0.28	-0.23	<b>0.05</b>	0.17
b	-6.78	-4.96	-2.32	<b>-1.88</b>
c	-13.88	-14.32	-3.73	<b>-3.13</b>
d	<b>0.35</b>	0.72	0.50	0.64
e	-0.15	<b>0.14</b>	0.30	0.46

Table 4.3. Median of the disparity difference between dense matching and ground truth.

Leaf	$D_{\text{median}}$ (pixels)			
	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
a	0.11	-0.11	-0.10	<b>-0.00</b>
b	-1.78	-1.72	-2.02	<b>-1.57</b>
c	-3.91	-3.30	-3.54	<b>-3.12</b>
d	<b>0.32</b>	0.48	0.40	0.57
e	<b>0.06</b>	0.29	0.28	0.40

Table 4.4. STD of the disparity difference between dense matching and ground truth.

Leaf	$D_{\text{STD}}$ (pixels)			
	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
a	4.49	4.48	<b>2.37</b>	2.76
b	19.61	15.02	1.29	<b>1.28</b>
c	25.53	30.65	7.86	<b>6.38</b>
d	2.73	3.16	<b>1.06</b>	1.13
e	5.35	2.84	<b>0.70</b>	0.86

By comparing the results in Table 4.2 and Table 4.3, it can be observed that the median is as expected more robust to outliers than the mean (e.g. for leaf c, all the  $D_{\text{median}}$  are around

Table 4.5. MAD of the disparity difference between dense matching and ground truth.

Leaf	Census	$D_{MAD}$ (pixels)		
		MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
a	0.76	<b>0.57</b>	<b>0.57</b>	0.63
b	3.03	0.51	0.42	<b>0.40</b>
c	3.49	0.64	<b>0.63</b>	<b>0.63</b>
d	0.73	0.67	<b>0.60</b>	0.65
e	0.50	0.46	<b>0.43</b>	0.51

3 pixels). Leaf b and c show a relatively large systematic disparity difference. This can be attributed to the systematic error caused by the shape change and imperfect point cloud registration of the ground truth disparity map.

The  $D_{STD}$  values in Table 4.4 show the robustness of MC-CNN-LiDAR and MC-CNN-SelfT, as they exhibit much lower  $D_{STD}$  than Census and MC-CNN-Pre.

$D_{MAD}$  has been widely used for depth map evaluation, as it is more robust to outliers than  $D_{STD}$ . The disparity map generated from Census has a relatively high  $D_{MAD}$  for the leaves b and c. This is due to the large amount of noise in the Census results, as visible in Figure 4.6.

In addition to the pixel-based direct comparison, the reconstruction completeness and the percentage of the accurately measured pixels are calculated. The reconstruction completeness is calculated using Equation 4.9.

$$Cpl = \frac{n_{DM/G}}{n_G} \times 100\%. \quad (4.9)$$

$n_G$  denotes the number of pixels with a valid disparity value provided by the ground truth in each leaf.  $n_{DM/G}$  denotes the number of pixels where both dense matching and ground truth provide disparity values. Thus the completeness  $Cpl$  will be the percentage of pixels in ground truth which are reconstructed by the dense matching as well.

However due to the systematic error, the disparity difference  $D_p$  between dense matching and ground truth cannot be directly utilized for evaluation. Therefore, we remove the systematic disparity shift for each leaf before computing the percentage of accurate pixels.

$$Acc = \frac{n_{pass}}{n_G} \times 100\%. \quad (4.10)$$

$$n_{pass} = \text{the number of pixels if: } |D_p - D_{\text{median}_{\text{mean}}} | \leq \varepsilon. \quad (4.11)$$

$$D_{\text{median}_{\text{mean}}} = \text{mean}(D_{\text{median}_{\text{scheme } i}}) \quad i \in \{1, 2, 3, 4\}. \quad (4.12)$$

In the above equations,  $D_{\text{median}_{\text{mean}}}$  is the mean of  $D_{\text{median}}$  calculated using each of the four matching schemes for each leaf.  $n_{pass}$  counts the number of pixels with the deviation below  $\varepsilon$ , a pre-defined threshold to evaluate the corresponding accuracy. In this research,  $\varepsilon$  is set as 0.5 and 1 pixel respectively for the test. The results are shown in Table 4.6.

MC-CNN-SelfT consistently obtains a slightly higher completeness than MC-CNN-LiDAR, while MC-CNN-LiDAR obtains slightly higher accuracy values for most leaves, except for leaves b and c, where MC-CNN-SelfT shows significantly better completeness and 1 pixel



Table 4.6. Evaluation of reconstruction completeness and accuracy for each dense matching scheme.

Algorithm	a			b			c			d			e		
	Cpl	Acc		Cpl	Acc		Cpl	Acc		Cpl	Acc		Cpl	Acc	
		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p
Census	92.0	31.8	57.0	63.0	14.8	23.9	49.7	7.6	14.0	92.0	36.4	56.9	89.7	43.3	71.0
MC-CNN-Pre	91.1	42.1	67.3	82.0	39.0	62.5	59.8	23.6	37.0	91.5	37.6	63.3	85.0	45.6	72.9
MC-CNN-LiDAR	96.9	<b>43.8</b>	<b>72.1</b>	89.2	<b>51.9</b>	70.7	86.4	34.5	60.5	<b>99.4</b>	<b>44.3</b>	<b>69.4</b>	97.1	<b>55.6</b>	<b>82.5</b>
MC-CNN-SelfT	<b>97.9</b>	41.0	67.0	<b>98.6</b>	51.0	<b>81.4</b>	<b>95.7</b>	<b>39.7</b>	<b>62.2</b>	<b>99.4</b>	41.9	67.8	<b>99.5</b>	47.9	77.4

accuracy values. Both re-trained methods consistently outperform Census and MC-CNN-Pre. This shows that especially MC-CNN-SelfT, which does not require additional LiDAR ground truth data, is a good approach for significantly improving the leaf reconstruction.

In this experiment, MC-CNN-LiDAR is handicapped due to imperfect ground truth, leading to disadvantages compared to the MC-CNN-SelfT method. We therefore assume that the scores for MC-CNN-LiDAR could be improved slightly by using a perfectly registered ground truth. However due to different registration errors for each leaf (cf. Table 4.3), the LiDAR trained network is not able to learn and correct for a systematic error between the LiDAR point cloud and the image data. We thus believe that the evaluation does not favor a specific method.

The second experiment was performed as part of our project "ForDroughtDet (FKZ: 22WB410602)" aiming at detecting the physiological and morphological status of trees under drought stress and studying the adaptation of forest areas to climate change. A major part of the project focuses on constructing a detailed and accurate 3D model of tree leaves in order to monitor the shape change when facing drought.

For this purpose, two nadir-viewing cameras are mounted on a crane system for stereo measurement. When the system is lifted above the trees, a stereo image pair of the tree crowns can be obtained. In order to test the feasibility of the stereo method described in this research, a stereo image pair above a beech tree subject to slightly artificial drought stress is collected. Some information about the images and the camera setting is shown in Table 4.7.

Table 4.7. The image acquisition parameters.

<b>Camera model</b>	SONY ILCE-5100
<b>Height</b>	4000 pixels
<b>Width</b>	6000 pixels
<b>Exposure time</b>	1/60 sec
<b>ISO speed rating</b>	125
<b>Focal length</b>	19.0 mm
<b>Object distance</b>	~ 3 m
<b>GSD</b>	0.06 cm/pixel
<b>Baseline length</b>	~ 0.25 m
<b>Acquisition date</b>	June 19 <sup>th</sup> , 2018

The corresponding epipolar image pair is shown in Figure 4.7. In this experiment, no LiDAR data is available, thus only Census, MC-CNN-Pre and MC-CNN-SelfT can be applied. The disparity map computed using MC-CNN-SelfT is shown in Figure 4.8.

Figure 4.7 shows that the large beech tree crown is much more complex, and has much smaller leaves than the indoor tree used in the first experiment. The slight drought stress

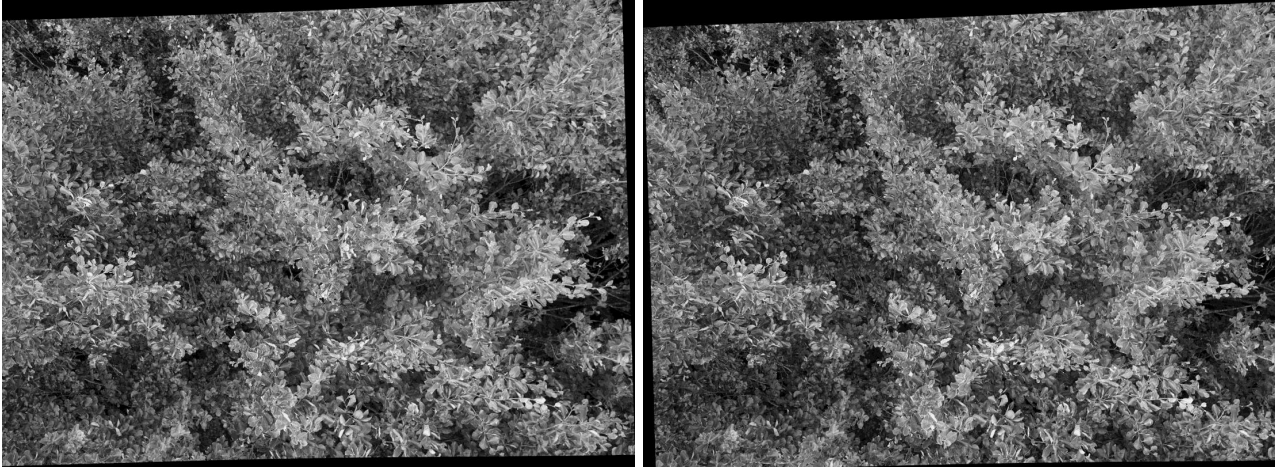


Fig. 4.7. The epipolar image pair for the second experiment, which is collected from the test region of our project.

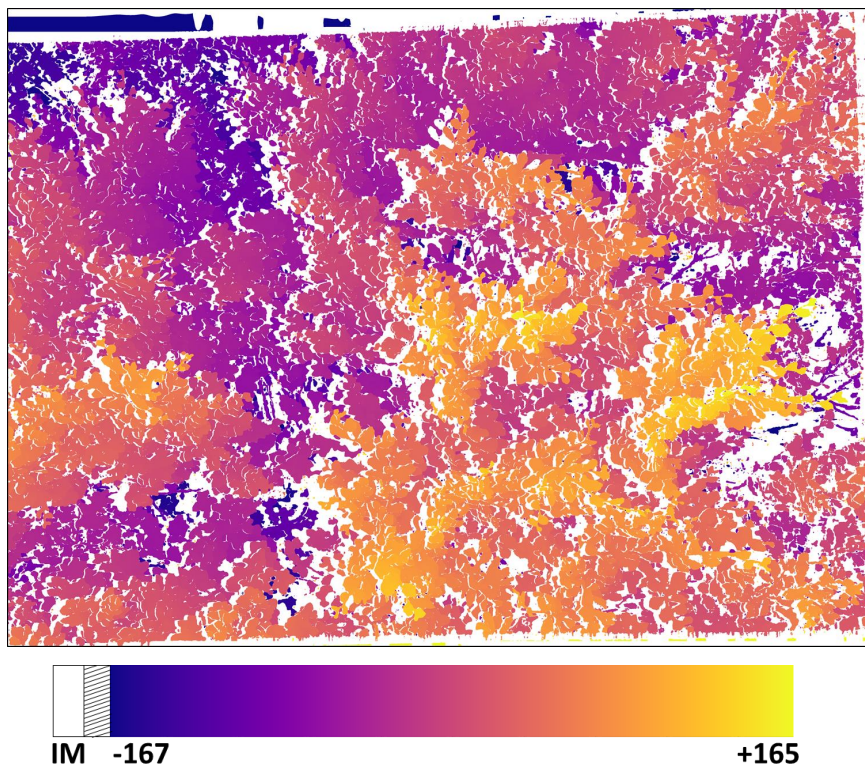


Fig. 4.8. The disparity map generated using self-trained MC-CNN. Inconsistent matching (IM) is represented by the color white.

leads to multiple different leaf shapes. Under the hypothesis that curved leaves are an indicator for drought stress, the stereo method should enable a clear separation of planar and curved leaves. The generated disparity map provides a dense reconstruction of the tree crown, and individual leaves are separable. The reconstruction completeness for MC-CNN-Pre and MC-CNN-SelfT, are 76.0% and 78.7%, respectively. Due to the lack of ground truth, the value is computed as the ratio of pixel passing the left-right check to the number of valid pixels in the rectified image. Some leaves under drought stress are selected for visual comparison. As shown in Figure 4.9, the curled shape of the leaves is clearly visible in the disparity image and the profile plot. It can be found that all the profiles are roughly U shaped, similar to the true shape of the leaves.

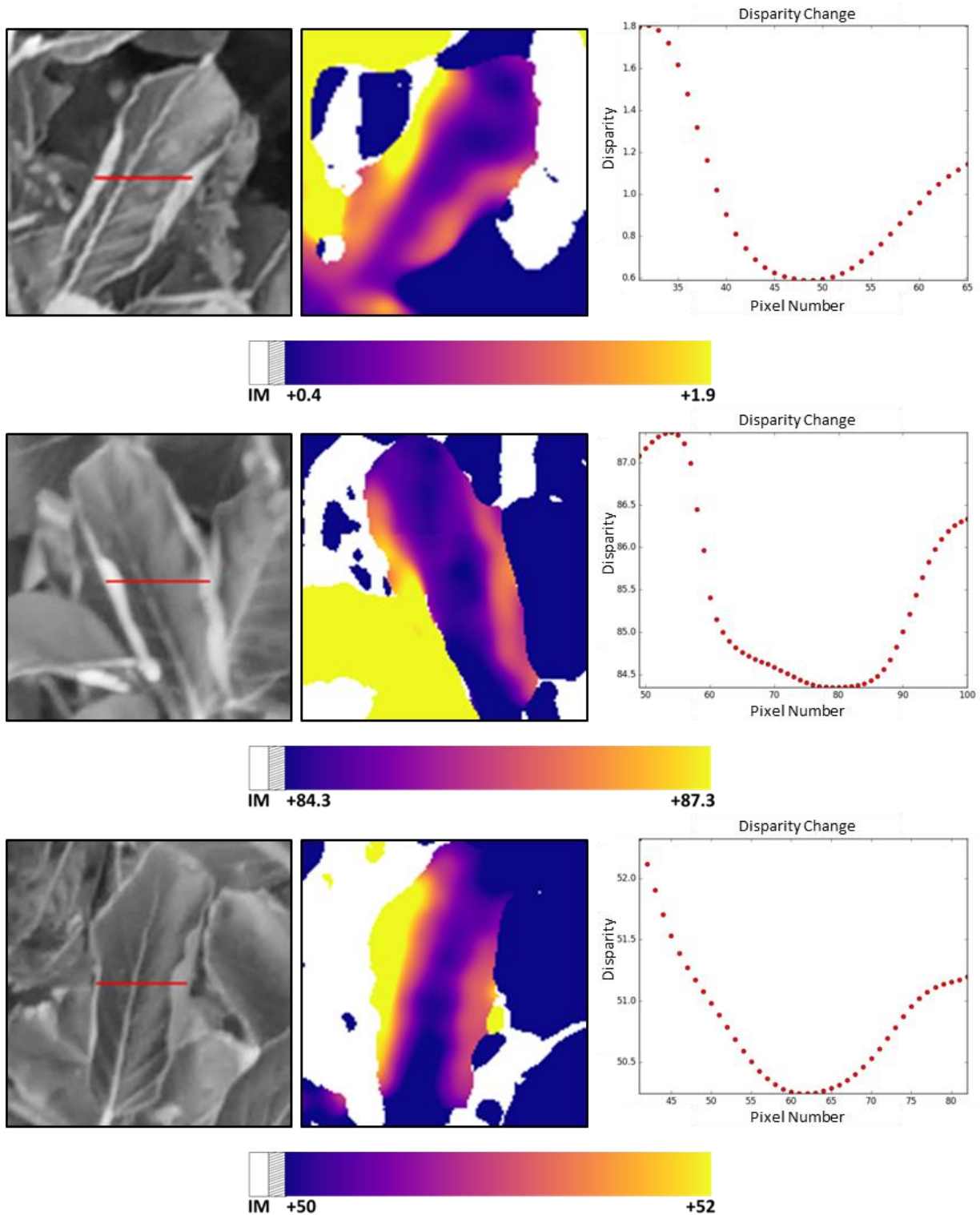


Fig. 4.9. Leaves under drought stress. From left to right in each subset: the master epipolar image, the disparity map of the self-trained MC-CNN matching scheme, and the disparity profile along the red line. The color represents the disparity. From blue to yellow, the targets get closer to the camera. Pixels with inconsistent matching (IM) are shown in white color.

#### 4.1.6 Discussion and Outlook

Dense stereo matching remains to be an open question through decades, considering the practical difficulties encountered for data collection, annotation, etc. SGM combined with MC-CNN has proven to outperform most previous algorithms, however, in practice it is extremely difficult to capture a large amount of high-quality training data. When the object

is complex, e.g. the plant reconstruction in this study for which the leaves exhibit similar shape and intensity information, MC-CNN should provide accurate measure to adequately represent the similarity between patches as the basis for the subsequent SGM processing. In this research, a self-trained MC-CNN without the use of ground truth is tested to reconstruct the plant. Based on the dense matching results of MC-CNN pre-trained on the Middlebury datasets, a rigid left-right consistency check is applied to limit the outliers and the filtered results are utilized to further train the net. The reconstructed plant shows superior performance for the self-trained version than for the pre-trained one and the classic Census algorithm. Compared with MC-CNN further trained using the ground truth from LiDAR, the self-trained net behaves slightly worse in accuracy but better in reconstruction completeness. The self-training strategy of MC-CNN is also applied to the stereo imagery of a natural forest tree under drought condition. The resultant disparity map is capable of showing the deformation of leaves, which highlights the possibility of the self-trained MC-CNN to monitor the tree health situation.

In future research, more approaches will be tested to capture the ground truth for outdoor experiments, for instance the structured light technique (Scharstein and Szeliski, 2003). Also the reconstruction of other more stable objects like buildings could be attempted. Furthermore, multi-viewed dense matching can be used to improve the self-training. Multiple images can in fact provide denser reconstruction results, meanwhile a consistency check among more than two images is able to further remove outliers which guarantees more reasonable training data. The self-training strategy of MC-CNN provides the possibility for complex object reconstruction and avoids the complexity of collecting ground truth especially in extreme situations.

## 4.2 Adaptive Scanlines Selection in Semi-Global Matching

A good pixel similarity measure could guarantee an accurate matching cost computation, in order to determine correspondences between the stereo pair for the scene depth prediction. However, the performance of a stereo matching algorithm is also largely dependent on the disparity distribution among neighboring pixels, considering the smoothness of the resultant disparity map. Traditional Semi-Global Matching (SGM) algorithm realizes the spatial smoothness, via approximating a 2D Markov Random Field (MRF) from multiple 1D scanline optimizations (SO) (Scharstein and Szeliski, 2002). Thus, it acquires a good compromise between accuracy and efficiency. Nevertheless, the empirical scanline summation, applied in SGM to approach 2D smoothness, is essentially a sub-optimal solution due to the difference of the depth prediction by each single scanline. Therefore, SGM's performance varies a lot within the scene for different objects. SGM-Forest (Schönberger et al., 2018) improves SGM by training a random forest to predict the best scanline for further disparity estimation, according to each scanline's disparity proposal. The best scanline then acts as the reference to adaptively adopt other scanline with close disparity proposals for the result refinement. In our research, it is frequently found that more than one scanline is capable of well predicting the disparity. Aiming at selecting a single scanline for training the random forest may hamper or even confuse the model when other good scanlines exist. Hence, we propose a multi-label classification strategy to modify the SGM-Forest method (termed SGM-ForestS for the follow-up). Each training sample is allowed to be described by multiple labels (or zero label) if more than one (or none) scanline provides a good prediction. The proposed method (termed SGM-ForestM) is tested on multiple stereo datasets cross domains, from Middlebury (Hirschmuller and Scharstein, 2007; Scharstein and Pal, 2007; Scharstein et al., 2014), ETH3D (Schöps et al., 2017), EuroSDR image matching benchmark (Haala, 2014), and the 2019 IEEE GRSS data fusion contest (Bosch et al., 2019; Le Saux et al., 2019), indicating that the multi-label strategy enhances the performance consistently.

### 4.2.1 Background

Stereo matching recovers the depth information according to the dense correspondence between stereo images. Local stereo methods intuitively locate the corresponding pixels by searching for the most similar points (and the surrounding patches), while global stereo methods additionally consider the spatial smoothness (Bleyer and Breiteneder, 2013). Thus, the former is normally more efficient but less accurate than the latter. SGM well balances the two categories of methods, via a data term for matching cost measure and a smoothness term based on dynamic programming (DP) to accomplish the spatial harmony among neighboring points. Hence, the method is regarded as the baseline, and keeps promoting the stereo methods development. As for the data term, Ni et al. (2018) combined three measures to calculate the matching cost for SGM, to keep robust in non-ideal radiometric conditions. Zbontar and LeCun (2016) initiated a CNN based method to calculate a similarity score between image patches, for matching cost computation in SGM. Luo et al. (2016) accelerated the mode based on multi-class classification.

Regarding the smoothness term, Seki and Pollefeys (2017) designed a CNN to adaptively penalize conflicting disparity prediction between neighboring pixels, to better guarantee the smoothness of the disparity map. The approach performed well in various situations, e.g. slanted plane, and depth boundaries, etc. Scharstein et al. (2017) enhanced SGM's robustness to handle untextured or weakly-textured slanted regions. The penalty term could be adjusted according to the prior knowledge of the scene's depth via the pre-computed surface orientation. Michael et al. (2013) demonstrated that the disparity map generated using each single scanline might exhibit varying qualities depending upon the global scene structure. Therefore, they assigned a specific weight to each scanline for deriving a weighted summation before WTA in SGM. Poggi and Mattocchia (2016) extracted a feature vector for each pixel using the disparity map estimated by a single scanline. The statistical dispersion of disparity within the surrounding patch was included in the feature, which was then fed to a random forest to predict a confidence measure for the corresponding scanline. Zhang et al. (2019) proposed a semi-global aggregation layer as a differentiable approximation of SGM to accomplish an end-to-end network. The network could adaptively learn a penalty term for each single pixel along a certain directed scanline, allowing for a more reasonable penalty for neighboring disparity inconsistency. Beside, a local guided aggregation layer was proposed for thin structures refinement. The network improved the performance of SGM significantly, and achieved good predictions for challenging situations, e.g. occlusions, textureless areas, etc.

### 4.2.2 Limitations of SGM

Global stereo methods explicitly consider the smoothness demand in addition to photo consistency. Accordingly, an energy function is defined for which a disparity map is optimized to properly balance the two terms (data term and smoothness term) and approach the energy minimization. This optimization, however, cannot be achieved in 2D since that the disparity determination for each pixel will affect every other pixel under the smoothness assumption, resulting in an np-complete problem (Bleyer and Breiteneder, 2013). SGM regularizes the disparity estimation by performing 1D SO in multiple canonical directions, typically 8 or 16, and then summing up the corresponding energy functions. Thus, 2D SO is approximated and the disparity value corresponding to the minimum energy is selected based on the WTA strategy.

Starting from the image boundaries, SGM aggregates the energy towards the target pixel along a 1D path (scanline). Thus, for each pixel, the previous points are already considered during the energy aggregation, which contributes to 1D smoothness. By summing up the aggregated energy from multiple 1D paths, the disparity corresponding to the minimum

energy is found based on the WTA strategy and 2D smoothness is approximated. For a pixel located at image position  $p$  with a sampled disparity  $d$  from the disparity space, the energy along the path traversing in direction  $r$  is defined as:

$$L_r(p, d) = C(p, d) + \min ( L_r(p - r, d), L_r(p - r, d - 1) + P_1, L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2 ). \quad (4.13)$$

$L_r(p, d)$  represents the energy.  $C(p, d)$  is the photo inconsistency under the current parallax and the rest of Equation 4.13 controls the smoothness by imposing a penalty term for a conflicting disparity setting between  $p$  and its previous neighbor  $p - r$ . A small penalty  $P_1$  is applied for only 1 pixel difference, otherwise a larger penalty term  $P_2$  is used. With multiple canonical directions  $r$  considered, the energy is summed up.

$$S(p, d) = \sum_r L_r(p, d). \quad (4.14)$$

The disparity is computed according to the WTA strategy as:

$$d_p = \operatorname{argmin}_d S(p, d). \quad (4.15)$$

SGM is able to derive a suitable disparity for each pixel with spatial smoothness considered, meanwhile spending reasonable runtime proportional to the reconstructed volume (d'Angelo and Reinartz, 2012; d'Angelo, 2016). Thus the algorithm has been applied in numerous fields, including building reconstruction, digital surface model generation, robot navigation, driver assistance, etc. (Hirschmüller, 2011; Kusch et al., 2014; Qin et al., 2015). However, the energy summation from all scanlines and the corresponding WTA strategy are empirical steps without a theoretical background, which is essentially inadequate when different scanlines propose inconsistent solutions.

### 4.2.3 Limitations of SGM-Forest

Schönberger et al. (2018) inferred that the upper bound of the matching accuracy can be approached by always selecting the best disparity proposal from all the scanlines. Therefore, they trained a random forest for the best scanline selection, via simply using the disparity proposed by each scanline and the corresponding costs as input, instead of handcrafting feature to feed random forest. A confidence value is obtained for each scanline. Based on the disparity predicted by the best scanline, other close disparity proposals were also adopted for a weighted average according to the corresponding confidence measures.

Specifically, the input feature for the random forest is constructed in this way. Assuming a pixel at location  $p$  has a WTA winner  $d_p^{r'}$  along a certain path  $r'$  as:

$$d_p^{r'} = \operatorname{argmin}_d L_{r'}(p, d), \quad (4.16)$$

the corresponding costs  $K_p^r(r')$  on  $d_p^{r'}$  along all  $N$  scanlines are calculated, where  $N$  is the number of directions considered.

$$K_p^r(r') = L_r(p, d_p^{r'}), \quad r = 1 \dots N. \quad (4.17)$$

$N+1$  elements  $\{d_p^{r'}, K_p^{r=1}(r'), \dots, K_p^{r=N}(r')\}$  are obtained for the current scanline of  $r'$ . Thus for all the scanlines, a feature vector with a length of  $(N+1) * N$  is acquired for  $p$  which is then

fed into a random forest for the best scanline prediction  $r^*$  and a posterior probability  $\rho^*$ . In order to achieve a more robust estimation, the corresponding disparity  $d_p^{r^*}$  acts as a baseline to select other scanlines with a close prediction for a weighted averaging computation:

$$\hat{d}_p = \frac{\sum_r d_p^r \rho_p^r}{\sum_r \rho_p^r}, \quad (4.18)$$

where  $d_p^r$  is selected from a set of WTA winners differing  $d_p^{r^*}$  by less than  $\epsilon_d$ , and  $\rho_p^r$  is the corresponding posterior probability predicted by the random forest as:

$$D_p = \left\{ (d_p^r, \rho_p^r) \mid |d_p^r - d_p^{r^*}| < \epsilon_d \right\}, \quad r = 1 \dots N. \quad (4.19)$$

The sum of selected posterior probabilities  $\hat{\rho}_p = \sum_r \rho_p^r$  is the confidence measure of  $\hat{d}_p$ .  $\hat{\rho}_p$  is then used for a confidence-based median filtering within an adaptive local neighborhood  $\mathcal{N}_p$  centered around  $p$  as follows:

$$\bar{d}_p = \text{median}(\hat{d}_q) \quad \text{and} \quad \bar{\rho}_p = \text{median}(\hat{\rho}_q), \quad q \in \mathcal{N}_p \quad (4.20)$$

$$\mathcal{N}_p = \left\{ q \mid \|q - p\| < \epsilon_p \wedge |I_q - I_p| < \epsilon_I \wedge \hat{\rho}_q > \epsilon_\rho \right\}, \quad (4.21)$$

where  $\|q - p\|$  measures the Euclidean distance between  $q$  and  $p$ .  $I$  is the image intensity.  $\epsilon_p$ ,  $\epsilon_I$  and  $\epsilon_\rho$  are the corresponding pre-defined thresholds.

As for the training, assuming the pixel at location  $p$  has the ground truth disparity available as  $d_p^{GT}$ , the label for this training sample is set as:

$$\tilde{r} = \text{argmin}_r |d_p^r - d_p^{GT}|, \quad r = 1 \dots N. \quad (4.22)$$

The algorithm is robust and performs steadily better than standard SGM in multiple stereo matching benchmark datasets (Scharstein et al., 2014; Menze and Geiger, 2015; Schöps et al., 2017). However, in practice, there can be more than one scanline with good disparity prediction. It appears when multiple scanlines properly perceive the scene structure, therefore, are capable of predicting accurate disparity values simultaneously. For example, on a slanted plane extending horizontally, the two vertical scanlines (from bottom to top, and inversely), along which the slope is not explicitly expressed, should have better disparity estimation than the horizontal ones but achieve similar performance. Thus, the random forest gets confused when only a single best has to be selected. Figure 4.10 provides such an example. SO1-SO8 represent the disparity estimation through a single scanline in each of the 8 canonical directions. Along the green line in (a), the disparities predicted by each scanline (defined in (b)) are shown in (c) (blue dots), compared with the ground truth (red line). It is found that SO3 and SO7 accomplish better solution than the other scanlines, however, barely differ from each other. In this case, both scanlines should be selected.

To further analyze the problem, we investigate Middlebury (2005 and 2006) (Hirschmuller and Scharstein, 2007; Scharstein and Pal, 2007) and ETH3D (Schöps et al., 2017) benchmark datasets, recording the percentage of pixels with multiple ( $\geq 2$ ) scanlines predicting disparities close to the ground truth (differing by less than 1 pixel) in Table 4.8. The percentage of pixels with at least one well-predicting scanline is appended below, which indicates the theoretical upper bound of the performance, for SGM based on the random forest to select scanlines. Census (Zabih and Woodfill, 1994) is used here as the matching cost. It is found that, for most pixels (75.52% in Middlebury, 81.69% in ETH3D), more than one scanline potentially achieves a good disparity estimation.

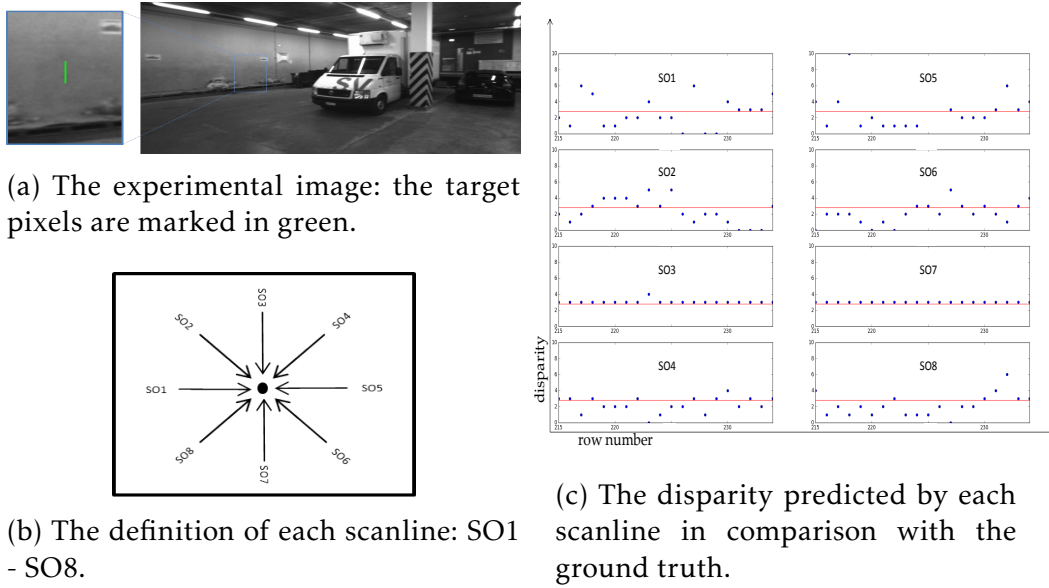


Fig. 4.10. The comparison between each single scanline’s disparity prediction and the ground truth, for pixels marked green in (a). It is found that both SO3 and SO7 accomplish good prediction and should be selected for further processing.

Table 4.8. The percentage of pixels with more than one scanline achieving good prediction for Middlebury and ETH3D benchmarks.

	Middlebury	ETH3D
Good scanline $\geq 2$	75.52%	81.69%
Good scanline $\geq 1$	83.83%	90.65%

It should be noted that SGM-ForestS further refines the disparity prediction by considering other scanlines with close proposals. However, it’s supposed to be more reasonable if the random forest learns to select all the proper scanlines directly in training. Therefore, we adjust the scanline selection based on a multi-label classification strategy and propose SGM-ForestM.

#### 4.2.4 Multi-Label Classification based Scanlines Determination

In this research, we define a standard to determine good or bad scanlines, aiming at guiding the random forest to select as many good scanlines as possible for disparity prediction. The samples with zero scanline selection (all regarded as bad) are included for training, so that a more comprehensive prediction is obtained.

Regarding the classification, traditional pattern recognition focuses on tasks with each class defined mutually exclusive (Duda et al., 2001). For some scenarios, however, there are samples with multiple attributes among different classes, e.g. a movie categorized into comedy and action film, which may confuse the classifier during training. In order to handle these samples properly, the first issue is label assignment. The most intuitive solution is to label a sample by the class it most likely belongs to. This strategy, nevertheless, is ambiguous and may result in a subjective judgment. An alternative is to neglect the samples related to multiple classes and concentrate only on the rest with a distinct definition. Yet, the classifier trained in this way is not able to deal with multi-label samples in the test period.

The two schemes above simply ignore the multi-label attribute of the samples and still treat the problem based on a single label classification strategy, therefore, the performance is limited. To cover all the corresponding labels of each sample, a new option is to define some “composite” classes, of which each class includes a certain combination of base classes, e.g.



"building + plant" from "building" and "plant". Then each composite class is allocated with a new label number above the original range for training. The samples categorized as composite classes, however, are normally too sparse to train a well-behaved classifier (Boutell et al., 2004). Hence, Boutell et al. (2004) propose a "cross-training" strategy which simultaneously trains multiple binary classifiers. Each binary classifier aims at determining the existence of a certain base class, and regards the corresponding multi-label samples as positive examples for training. For example, the samples of "building + plant" are regarded as "building" and "plant", respectively, when training the "building classifier" and "plant classifier". Thus, all the labels of each training sample are considered, meanwhile the training data are explored more effectively. In this research, the "cross-training" scheme is applied for training the random forest based on a multi-label classification strategy, in order to process pixels with more than one scanline predicting appropriate disparities. With the cost aggregation applied along a certain path as Equation 4.16, if the estimated disparity is close to the ground truth, the corresponding pixel should be regarded as a positive sample for training the binary classifier of the path. Regarding the pixels marked green in Figure 4.10 as an example, the label should be set as positive for the classifier of SO3 and SO7, and as negative for the others. The multi-label strategy is appropriate for classification when overlap exists among different categories. The label assignment is more reasonable for non-mutually exclusive classes, in which one sample can be essentially related to multiple labels. It applies not only to computer vision, e.g. semantic scene classification, but also in many other fields including document analysis (e.g. text categorization), medicine (e.g. disease diagnosis), etc. (McCallum, 1999; Schapire and Singer, 2000; Clare and King, 2001; Boutell et al., 2004; Tsoumakas and Katakis, 2007).

The feature for our SGM-ForestM is extracted in the same way as SGM-ForestS, however, the label setting is adjusted to satisfy our multi-label concept. Instead of selecting the best scanline with the closest prediction to the ground truth as Equation 4.22, we define a threshold  $\epsilon_{dso}$  to extract all the promising scanlines as:

$$\mathcal{R}_p = \{r \mid |d_p^r - d_p^{GT}| < \epsilon_{dso}\}, \quad r = 1 \dots N. \quad (4.23)$$

Thus, the pixel  $p$  is a positive example when training the binary classifiers of all the corresponding scanlines contained by  $\mathcal{R}_p$ . Otherwise,  $p$  is regarded as negative.

Afterwards in the test period, the trained random forest gives  $N$  predictions and  $N$  probabilities for each pixel, indicating which scanlines should be regarded as good disparity proposals (with the corresponding probability,  $\rho_p^r$ , larger than 0.5). It should be noted that a probability value is calculated exclusively for a certain scanline with no dependency on the others. Unlike the single label classifier that the probabilities for all classes should be sum-to-one, the multi-label classifier is not restricted to follow the rule.

With multiple (or zero) scanlines proposed by the random forest, the one with the highest probability,  $r^*$ , is considered as a baseline to refine the disparity estimation as given in Equation 4.24 and 4.25 below:

$$\hat{d}_p = \frac{\sum d_p^r * \rho_p^r}{\sum \rho_p^r}, \quad d_p^r, \rho_p^r \in D_p \quad (4.24)$$

$$D_p = \{(d_p^r, \rho_p^r, r) \mid |d_p^r - d_p^{r^*}| < \epsilon_d\}, \quad r = 1 \dots N. \quad (4.25)$$

$D_p$  is constructed via selecting disparity estimation close to  $d_p^{r^*}$  from the WTA winners as SGM-ForestS. Thus, we limit the influence from the outliers, and ensure that one disparity value is available for further processing. As Equation 4.18 and 4.19, we refer to SGM-ForestS's strategy to consider scanlines with close disparity proposals, however, it should

be pointed out that the disparity refinement of our SGM-ForestM is based on more reasonable prediction,  $r^*$ , owing to multi-label classification. In addition, the confidence measure should be adjusted accordingly as:

$$\hat{\rho}_p = \frac{\sum_{r \in D_p} \rho_p^r}{\sum_{r=1}^N \rho_p^r}, \quad (4.26)$$

in which the nominator is still the sum of probabilities for selected scanlines as SGM-ForestS. The denominator, on the other hand, is the sum of all scanlines' probabilities in order to confine the confidence in the range of  $[0, 1]$ . Following SGM-ForestS, a confidence-based median filter is exploited as well.

Before testing the proposed algorithm, the efficiency and memory usage are evaluated. SGM approximates global energy function by summing up the aggregated costs along multiple 1D paths. The number of paths is determined according to application demands, hardware constraints or quality requirements (Schumacher and Greiner, 2014). With more paths considered, e.g. 8 or 16, better results are obtained incurring reduced streaking artifacts, however, at the expense of high computational complexity (Schumacher and Greiner, 2014; d'Angelo, 2016). As shown in Figure 4.11, SGM-Forest requires storing the full aggregated cost volumes for all aggregation directions, leading to increased memory usage over standard SGM. Thus, resource efficient solutions and high resolution data processing are hampered as the number of paths increases.

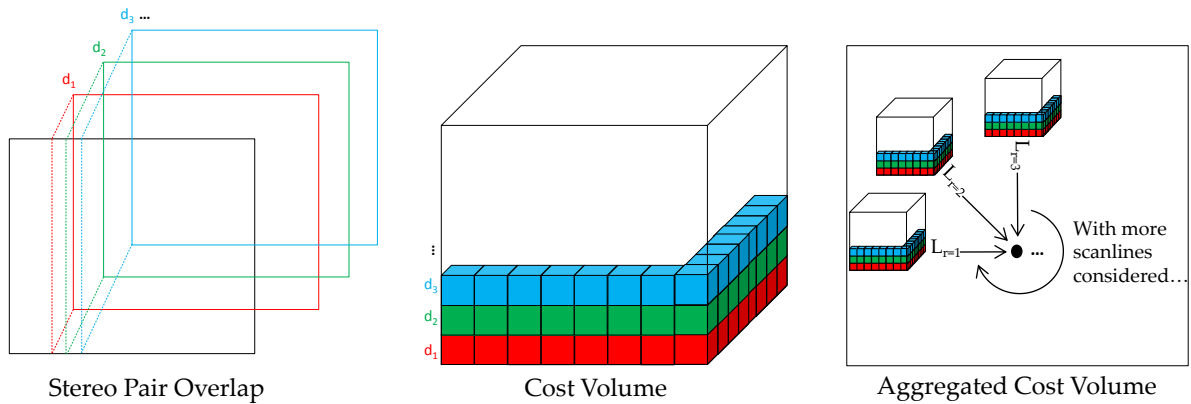


Fig. 4.11. Stereo pair, cost cube and the corresponding aggregated cost cube in SGM. As more scanlines are considered, the memory usage is highly increased.

Hence, we test different implementations of SGM, SGM-ForestS, and SGM-ForestM, by varying the number of scanlines considered for further processing. We aim at observing how the SGM-Forest algorithms are influenced, when fewer scanline proposals are applied. A particularly interesting case is the configuration with 5 scanlines starting from left, top-left, top, top-right and right, as this allows a memory efficient top down sweep implementation which only requires storing two lines of the  $C$  and  $L_r$  volumes, greatly reducing the amount of required memory. This enables the processing of very large stereo pairs with sizes of 200 to 2000 Megapixels, as typically occurring in aerial and satellite data. Thus, the potential of SGM-Forest for efficient systems can be explored, such as real-time designs in CPU and GPU systems, or embedded modules on e.g. embedded multi-core architectures and Field-Programmable Gate Arrays (FPGAs) (Gehrig et al., 2009; Banz et al., 2010; Gehrig and Rabe, 2010; Arndt et al., 2013; Schumacher and Greiner, 2014).

### 4.2.5 Performance Evaluation on Multiple Data Sources

In the experiments, SGM-ForestS and SGM-ForestM are compared with SGM as the baseline method. The implementation details are controlled among each method for an unbiased comparison, referring to Schönberger et al. (2018). As for the matching cost, Census (Zabih and Woodfill, 1994) and MC-CNN-acrt (Zbontar and LeCun, 2016) are tested, covering both classic and learning based algorithms. With regard to Census, a  $7 \times 7$  window size is set. For MC-CNN-acrt, the proposed network architecture is applied. The number of convolutional layers is 5, with 112 feature maps and  $3 \times 3$  kernel size for each. The number of fully-connected layers is 3, with the corresponding number of units as 384.

Regarding the SGM setting, the calculated matching cost is scaled to be in the range of  $[0, 1023]$ , and  $P_1$  and  $P_2$  are set as 400 and 700, respectively. We perform SO along 8 canonical directions ( $N = 8$  with 2 horizontal, 2 vertical, and 4 diagonal scanlines, as Figure 4.10) in order to implement SGM and generate input proposals to train the random forest for SGM-ForestS and SGM-ForestM. As described above, we also have an implementation for SGM, SGM-ForestS, SGM-ForestM by applying 5 SOs, in order to check the influence when using fewer scanlines. 2 horizontal, 1 vertical (pointing downwards), and 2 diagonal (pointing downwards) scanlines are included, which accomplish a top-down sweep of the scene to enable single-pass algorithms and consume less aggregation buffer (Schumacher and Greiner, 2014). As for the 8-scanlines version, both Census and MC-CNN-acrt are employed as matching cost, for a general comparison among the three SGM based algorithms. As the 5-scanlines version targets fast implementation, it is only tested using the faster Census data term.

Considering SGM-Forest, we exploit the same parameter setting as proposed in Schönberger et al. (2018). For both SGM-Forest versions, the same forest structure is adopted comprising 128 trees with the maximum depth of each as 25, based on *Gini impurity* to measure the split quality. Before feeding to the random forest, we normalize the disparity proposals  $d_p^r$  to relative values for feature vectors construction, in order to generalize across datasets. The disparity estimates are then denormalized to absolute values for further confidence based filtering.  $\epsilon_d$ ,  $\epsilon_p$ ,  $\epsilon_I$ , and  $\epsilon_\rho$  are respectively set as 2, 5, 10, and 0.1, which are determined according to parameter grid search and 3-fold cross validation based on Middlebury 2014 training datasets (Schönberger et al., 2018).  $\epsilon_{dso}$  is set as 1 pixel in SGM-ForestM. All our implementations are based on Python and C.

Our first experiment is conducted on close-range datastes, including two benchmarks, Middlebury and ETH3D, which supply a certain number of stereo pairs with ground truth disparity maps available. We split the datasets into non-overlapping training and validation sets (as shown below), in order to train our proposed algorithm and test the performance according to the validation accuracy. From the manually split training set, 500K pixels are randomly selected for training the random forest, while all the pixels are used to train MC-CNN-acrt. As for the Middlebury benchmark, the training set is acquired from 2005 and 2006 scenes, while 2014 scenes provide the validation set, as shown in Table 4.9. Each dataset from Middlebury 2005 and 2006 consists of 7 views under 3 illumination and 3 exposure conditions (63 images in total). Ground truth disparity maps are provided for view-2 and view-6. We regard the former as the master epipolar frame, and randomly select illumination and exposure condition for two images to construct stereo pairs for further processing.

ETH3D stereo benchmark contains various indoor and outdoor views with ground truth collected using a high-precision laser scanner. The images are acquired using a Digital Single-Lens Reflex (DSLR) camera synchronized with a multi-camera rig capturing varying field-of-views. The benchmark provides high-resolution multi-view stereo imagery, low-resolution many-view stereo on video data, and low-resolution two-view stereo images

Table 4.9. Train/validation splits for Middlebury benchmark.

	<b>Train</b>	<b>Validation</b>
<b>Middlebury 2005</b>	Books	Adirondack
	Dolls	ArtL
	Laundry	Jadeplant
	Moebius	Motorcycle
	Reindeer	MotorcycleE
<b>Middlebury 2006</b>	Aloe	Piano
	Baby1	PianoL
	Baby2	<b>Middlebury 2014</b> Pipes
	Baby3	Playroom
	Bowling1	Playtable
	Bowling2	PlaytableP
	Cloth1	Recycle
	Cloth2	Shelves
	Cloth3	Teddy
	Cloth4	Vintage
	Flowerpots	
	Lampshade1	
	Lampshade2	
	Midd1	
	Midd2	
	Monopoly	
	Plastic	
Rocks1		
Rocks2		

which are used in this experiment. There are 27 frames with ground truth for training and 20 for test. We exploit the former for train/validation splits, as shown in Table 4.10. For some scenes, the data include two different sizes. Both focus on the same target, however, with one contained in the field of view from the other (e.g. delivery\_area\_1s and delivery\_area\_1l). Therefore, we manually divide the datasets for training and validation, in order to avoid images taken for the same scene appearing in both splits.

Table 4.10. Train/validation splits for ETH3D benchmark.

<b>Train</b>	<b>Validation</b>
delivery_area_1s	delivery_area_2s
delivery_area_1l	delivery_area_2l
delivery_area_3s	electro_1s
delivery_area_3l	electro_1l
electro_2s	facade_1s
electro_2l	forest_2s
electro_3s	playground_2s
electro_3l	playground_2l
forest_1s	playground_3s
playground_1s	playground_3l
playground_1l	terrace_1s
terrains_2s	terrace_2s
terrains_2l	terrains_1s
	terrains_1l

Regarding the accuracy evaluation, the disparity results of SGM, SGM-ForestS, and our SGM-ForestM are compared with ground truth, with only the non-occluded pixels considered. The percentage of pixels with an estimation error less than 0.5, 1, 2, and 4 pixels, respectively, are calculated as indicated by Table 4.11 and 4.12. In Table 4.11, a suffix of '-5dirs' or '-8dirs' is appended at the end of each algorithm to differentiate SGM, SGM-ForestS, and SGM-ForestM implemented using 5 or 8 scanlines, respectively. For the follow-up, unless mentioned explicitly, all the SGM related terms without a suffix represent the 8-scanlines implementation.

Table 4.11. The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: Census. '-5dirs' for 5 scanlines version and '-8dirs' for 8 scanlines version).

	Middlebury				ETH3D			
	0.5pix	1pix	2pix	4pix	0.5pix	1pix	2pix	4pix
SGM-5dirs	55.89%	67.60%	73.34%	77.48%	67.60%	79.18%	85.80%	90.33%
SGM-ForestS-5dirs	55.97%	68.71%	74.44%	78.37%	70.87%	82.97%	89.93%	95.03%
SGM-ForestM-5dirs	56.88%	70.30%	76.44%	80.37%	71.83%	85.00%	91.69%	95.96%
SGM-8dirs	58.92%	69.47%	74.87%	78.84%	70.14%	80.88%	87.02%	91.27%
SGM-ForestS-8dirs	59.38%	70.71%	76.33%	80.41%	72.87%	83.91%	90.55%	95.44%
SGM-ForestM-8dirs	<b>60.38%</b>	<b>72.16%</b>	<b>78.00%</b>	<b>82.19%</b>	<b>74.04%</b>	<b>86.20%</b>	<b>92.48%</b>	<b>96.37%</b>

Table 4.12. The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: MC-CNN-acrt).

	Middlebury				ETH3D			
	0.5pix	1pix	2pix	4pix	0.5pix	1pix	2pix	4pix
SGM	69.35%	79.35%	83.37%	86.07%	72.39%	83.29%	89.48%	94.18%
SGM-ForestS	<b>70.01%</b>	<b>81.34%</b>	<b>85.71%</b>	<b>88.64%</b>	74.25%	86.03%	92.04%	96.30%
SGM-ForestM	69.92%	81.32%	85.56%	88.28%	<b>74.61%</b>	<b>86.47%</b>	<b>92.36%</b>	<b>96.44%</b>

As for 8-scanlines implementation, it is found that the two SGM-Forest implementations perform steadily better than the standard SGM, in both benchmarks considering different estimation errors as the upper limit. With MC-CNN-acrt as matching cost, the results on Middlebury datasets report slightly worse performance of SGM-ForestM (about 0.1% difference) than SGM-ForestS. However, a stable improvement is achieved by SGM-ForestM in all the other cases (the results on Middlebury and ETH3D using Census as matching cost, on ETH3D using MC-CNN-acrt as matching cost), which indicates the significance of applying the multi-label classification strategy to train the random forest.

For 5-scanlines version, the performance of all the algorithms decreases as expected due to the information loss using fewer scanlines. Nevertheless, SGM-ForestM is still better than SGM-ForestS, and both of them are superior to the standard SGM. It is worth to mention that, SGM-ForestS-5dirs and SGM-ForestM-5dirs achieve even better results than SGM-8dirs on ETH3D datasets, which indicates the potential to embed SGM-Forest into efficient stereo systems. On Middlebury datasets, SGM-ForestS-5dirs is not able to keep its superiority to SGM-8dirs. However, it's good to find that SGM-ForestM-5dirs remains to be better than the standard SGM using 8 scanlines (except for 0.5 pixel error) and proves its robustness.

On the other hand, MC-CNN is a "data-hungry" method, requiring a large amount of training samples before achieving high performance (Zbontar and LeCun, 2016). The training of the random forest in SGM-Forest, nevertheless, relies on much less data (500K pixels

used here and in Schönberger et al. (2018)). With Census as matching cost, SGM-ForestM consistently outperforms SGM and SGM-ForestS in all settings, which further indicates the potential of the algorithm, especially when the amount of data is too limited for training a well-performing MC-CNN.

In order to complete the demonstration for our multi-label classification strategy, below in Table 4.13, we provide the official results of the ETH3D benchmark by evaluating our SGM-ForestM on the test datasets. As the proposed method focuses on the refinement of SGM itself, we simply use Census for a quick test. The random forest is also trained on 500K pixels, with 8 scanlines for disparity proposals.

Table 4.13. The benchmark results of SGM-ForestM on ETH3D datasets (Matching cost: Census).

	SGM-ForestM			
	0.5pix	1pix	2pix	4pix
non-occluded	76.28%	83.01%	87.44%	91.11%
all	74.79%	81.39%	85.75%	89.42%

The accuracy for ‘non-occluded pixels’ is consistent with the numbers obtained in Table 4.11 (SGM-ForestM-8dirs), however, compared with other algorithms, our result is not competitive. The reason includes that, we execute no post-processing, e.g. left-right consistency check, interpolation, etc., and Census is used for calculating matching cost instead of a well-trained MC-CNN. It should be noted that the main goal of this research is to improve SGM and SGM-ForestS, therefore, the whole processing pipeline is not fully considered.

To explore deeper the random forest prediction, we also analyze the quality of  $r^*$ , which is the reference for further confidence based processing. Adaptive scanline selection based on a classification strategy is the core concept of SGM-Forest that is superior to the scanline average of the standard SGM. Hence,  $r^*$  and the corresponding  $d_p^{r^*}$  are necessary for comparison between SGM-ForestS and SGM-ForestM. In Figure 4.12 and 4.13, the error plots are displayed for SGM-ForestS, SGM-ForestM, and the upper bound of SO if the best scanline can always be selected from the 8 alternatives. At here, it should be noted that the disparity prediction of the random forest ( $d_p^{r^*}$ ) is directly compared to the ground truth for calculating the ratio of correct disparity estimation (y-axis), considering different estimation errors allowed (x-axis). We still test two matching cost algorithms (Census and MC-CNN-acrt) on two benchmark datasets (Middlebury and ETH3D).

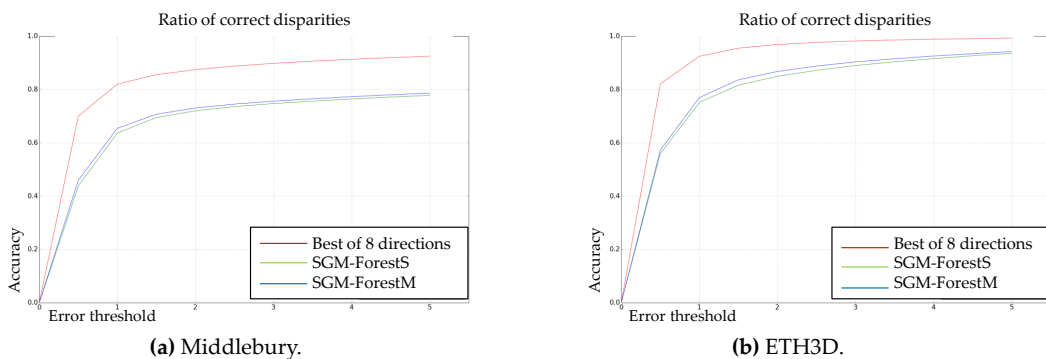


Fig. 4.12. Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: Census).

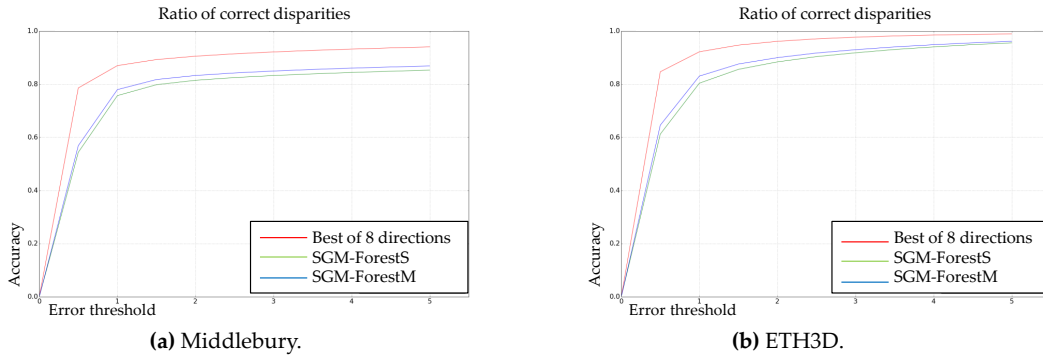


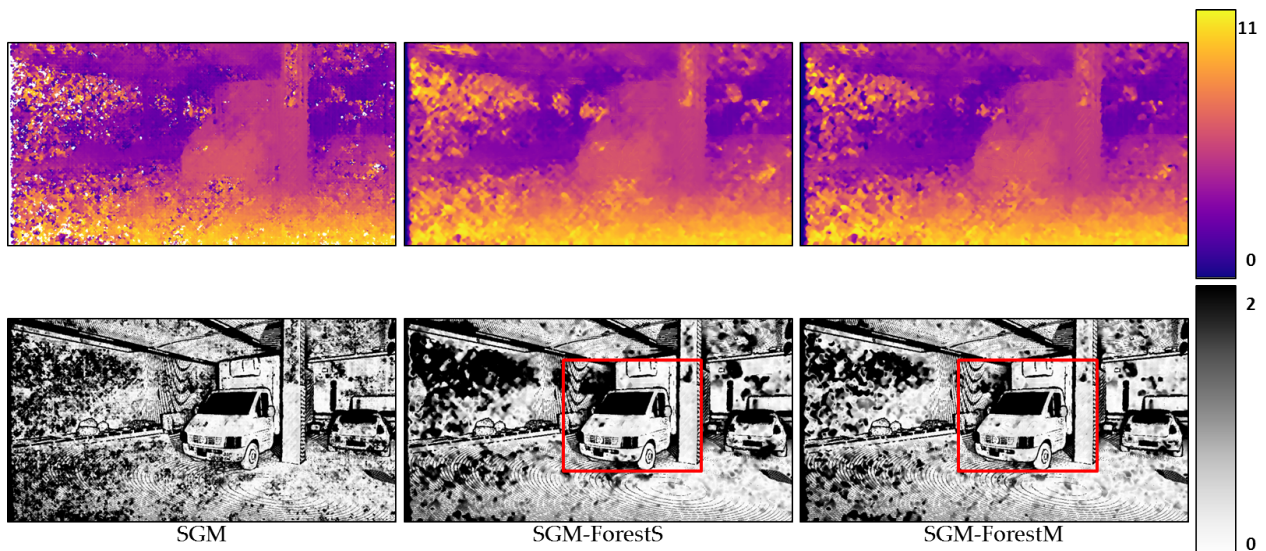
Fig. 4.13. Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: MC-CNN-acrt).

It is found that both SGM-Forest implementations achieve good performance to approach the best SO, which demonstrates the feasibility of scanline selection based on a classification framework. In addition, SGM-ForestM is superior to SGM-ForestS in all cases. The results indicate that SGM-ForestM is essentially better at scanline prediction and capable of deriving preferable initial disparity estimation for further processing.

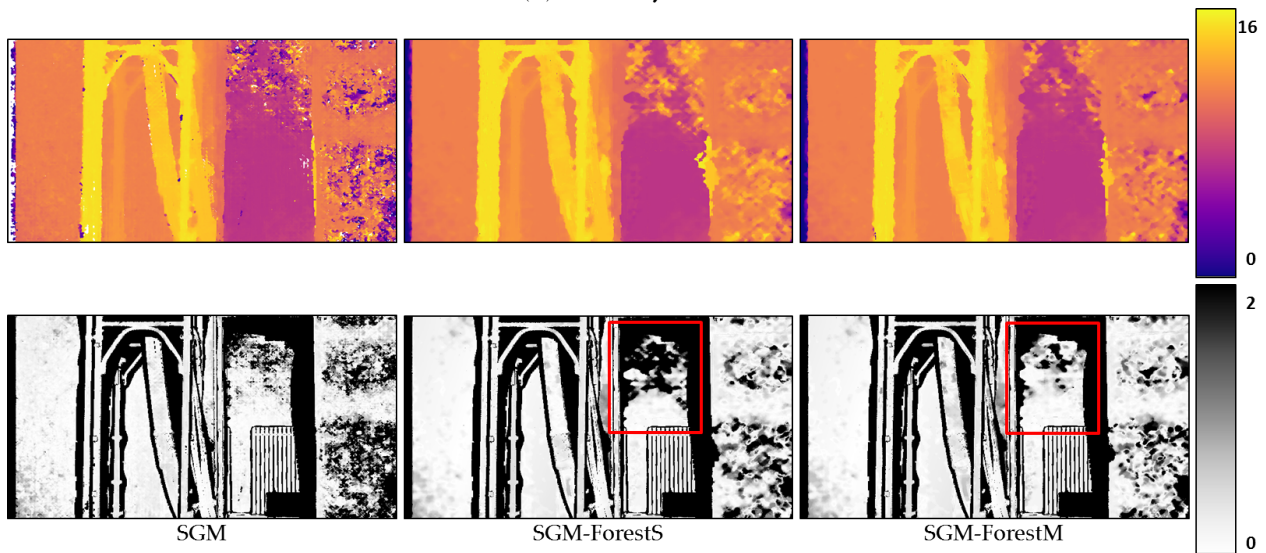
Figure 4.14 and 4.15 visualize the disparity results created by SGM, SGM-ForestS, and SGM-ForestM, respectively, on data selected from ETH3D. The error map is appended below each disparity map. Regarding "2 pixels" as the upper bound, all the pixels with an error above the bound are colored black, while the rest are colored uniformly according to the color bar. We apply Census and MC-CNN-acrt to calculate the matching cost, respectively.

In each subfigure, the disparity map and the error map for SGM, SGM-ForestS, and SGM-ForestM, respectively, are displayed from left to right, with a color bar at the end. The red rectangles marked in the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM. It is found that the disparity maps generated by the two SGM-Forest implementations are smoother than SGM. Moreover, according to the error map, SGM-ForestM suffers fewer errors compared with SGM-ForestS. Especially for the ill-posed regions (e.g. textureless areas, reflective surfaces, etc.), SGM-ForestM performs better as highlighted by the red rectangles.

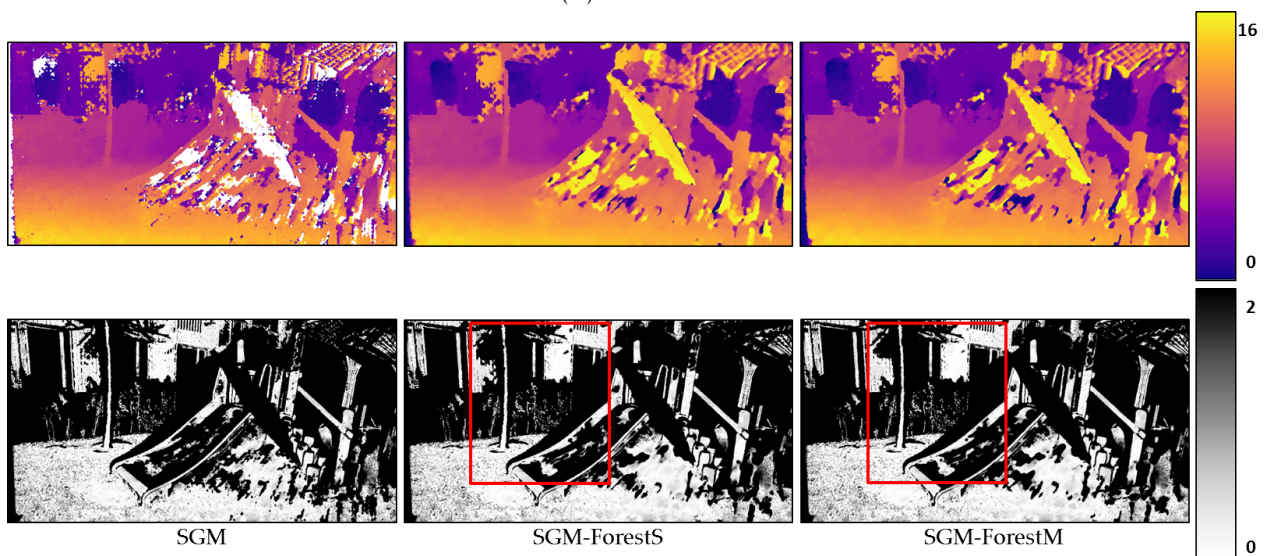
The second experiment is conducted on airborne data, using the aerial image matching benchmark from EuroSDR. The benchmark project is motivated by the development of matching algorithms and the improved quality of the elevation data obtained by advanced airborne cameras. Based on the benchmark datasets and the corresponding evaluation platform, the potential of the ongoing photogrammetric software is assessed by comparing their generated 3D products, including point clouds, DSMs, etc. We use the nadir airborne datasets, Vaihingen/Enz with moderate ground sampling distance (20 cm) and overlap (63% in flight and 62% cross flight) in the experiment. A stereo pair is randomly selected on which SGM, SGM-ForestS, and SGM-ForestM are applied to generate the disparity map, respectively. The master epipolar image and the corresponding result of each algorithm are displayed in Figure 4.16, with an area highlighted by a green rectangle for detailed comparison. According to the results, it is also found that the two SGM-Forest implementations generate a smoother disparity map than the standard SGM. Within the highlighted region, SGM-ForestM suffers less noise than SGM-ForestS, which further demonstrates the superiority of the former.



(a) delivery\_area\_21



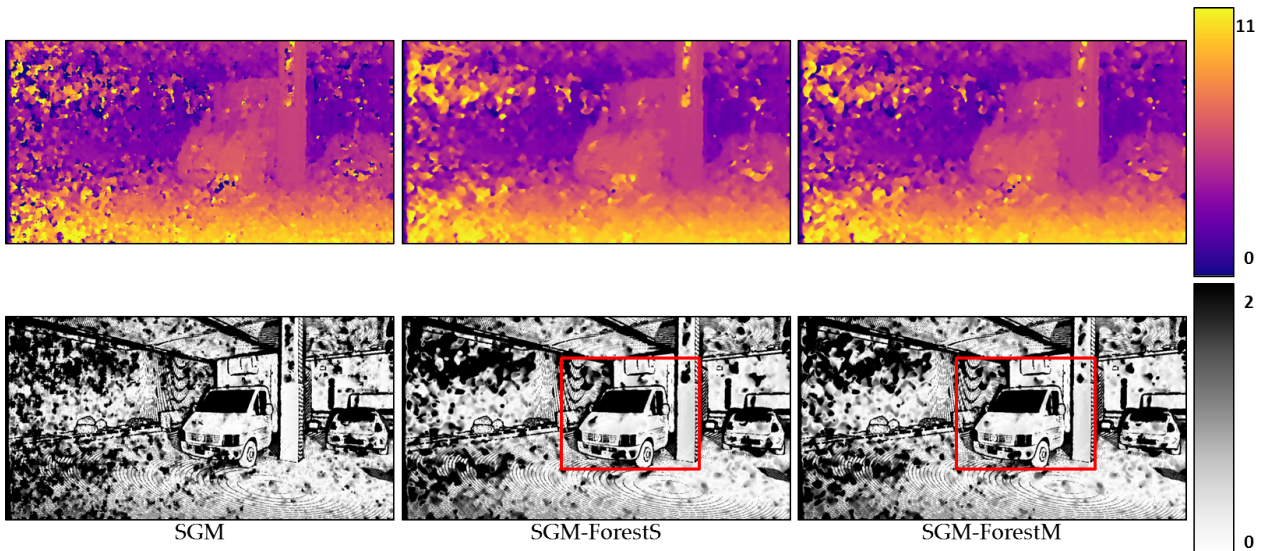
(b) electro\_1s



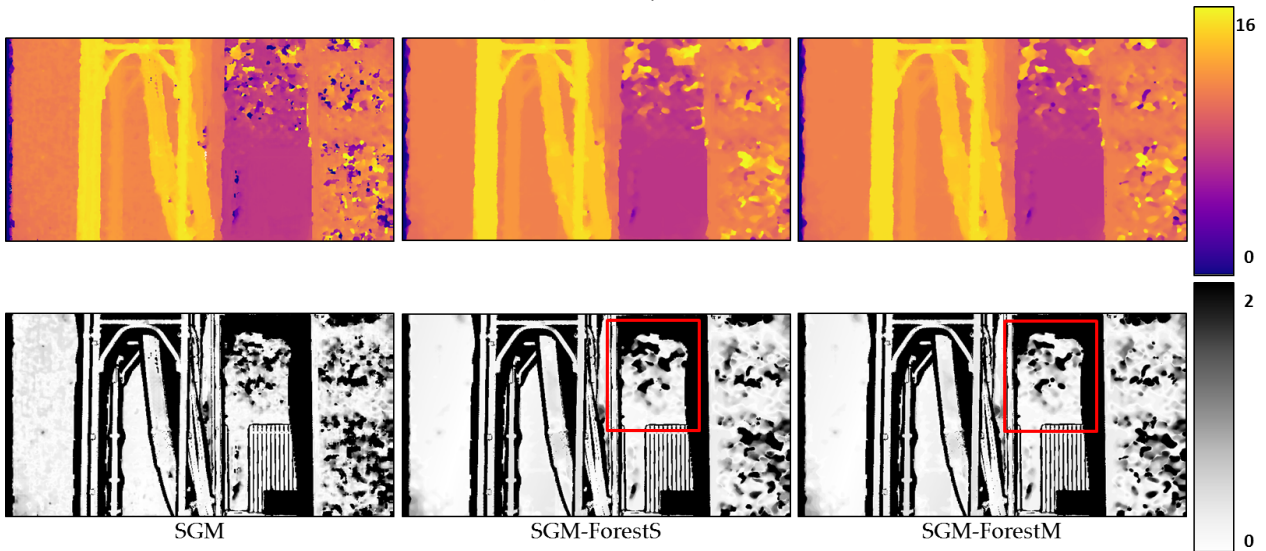
(c) playground\_21

Fig. 4.14. The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: Census). The red rectangles marked in the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM.

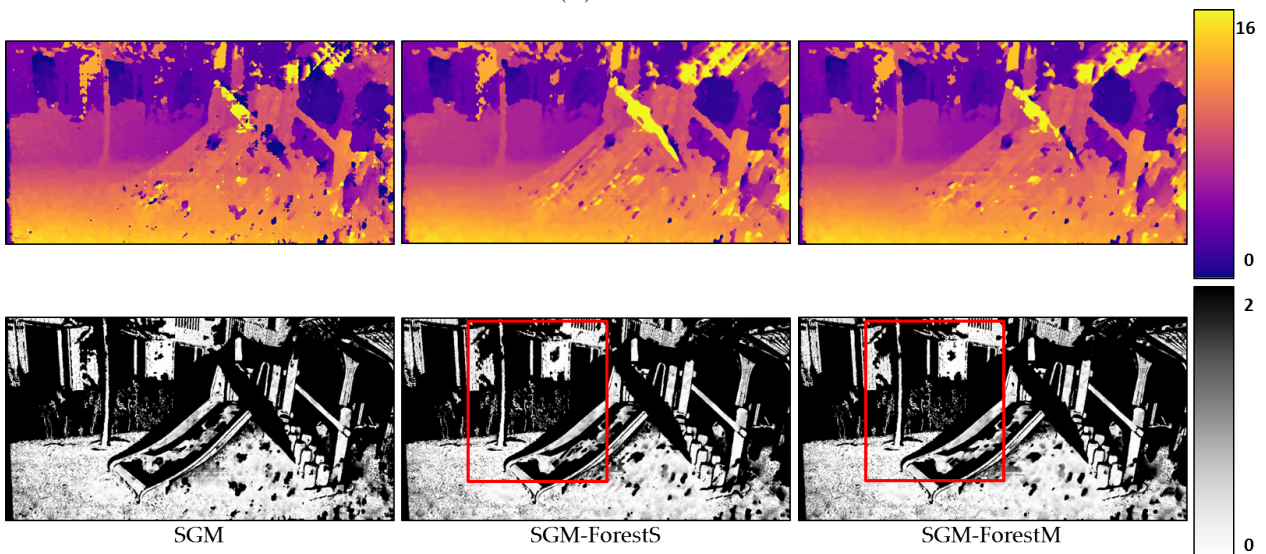




(a) delivery\_area\_21



(b) electro\_1s



(c) playground\_21

Fig. 4.15. The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: MC-CNN-act). The red rectangles marked in the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM.

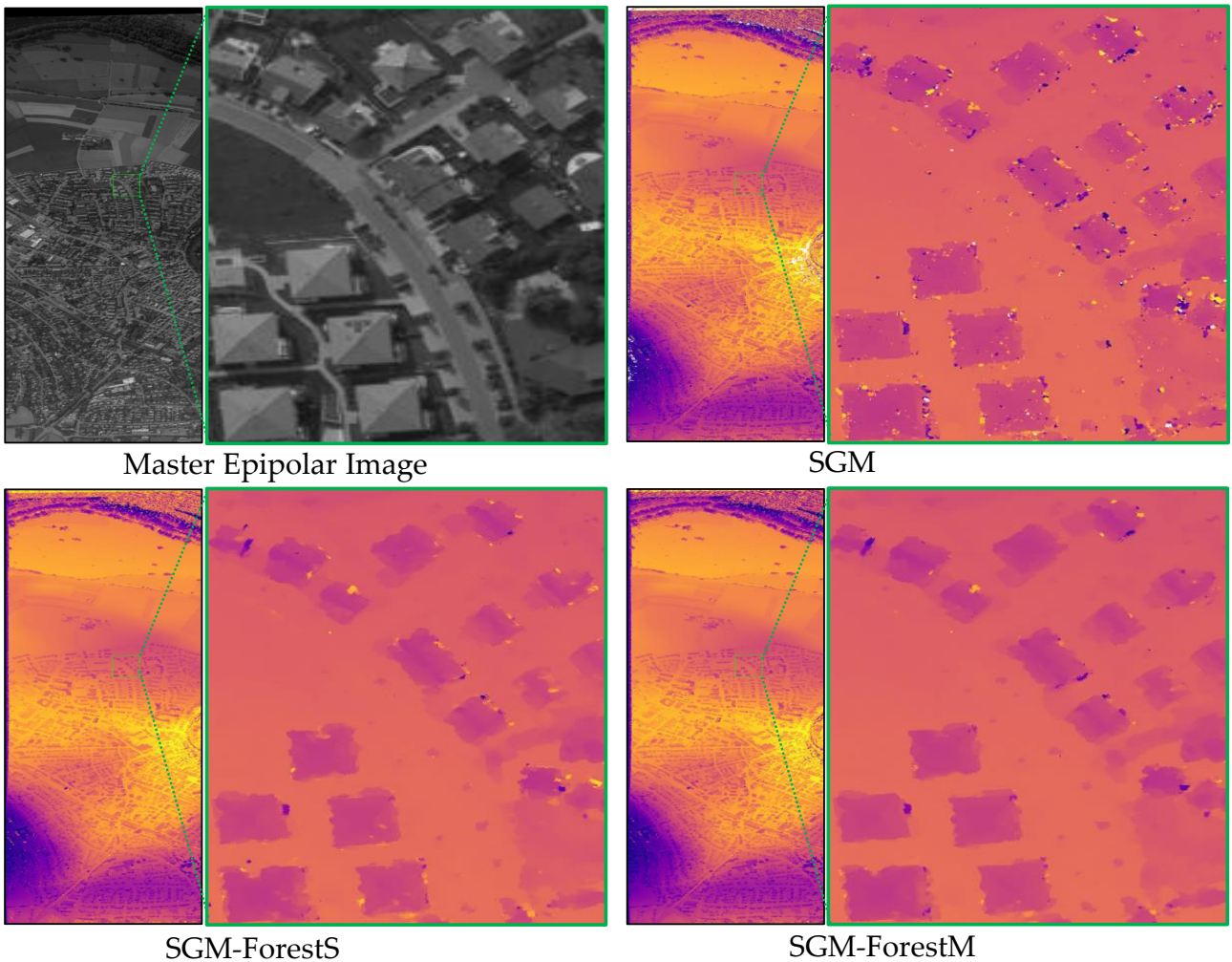


Fig. 4.16. Stereo matching results on EuroSDR benchmark datasets (Vaihingen/Enz). The master epipolar image and the corresponding disparity results are displayed. The green rectangle marks the region for detailed comparison.

The last experiment explores the algorithms' performance on satellite data, which is obtained from the pairwise semantic stereo challenge (Track 2) in the 2019 IEEE GRSS data fusion contest (Le Saux et al., 2019). The organizer provides the `grss_dfc_2019` dataset, a subset of the Urban Semantic 3D (US3D) data (Bosch et al., 2019), including multi-view, multi-band satellite images and ground truth geometric and semantic labels. Several tasks are designed to reconstruct both a 3D geometric model and a segmentation of semantic classes for urban scenes, aiming at further supporting the research in stereo and semantic 3D reconstruction using machine intelligence and deep learning.

The data are captured by WorldView-3 satellite including RGB and 8-band visible and near infrared (VNIR) multi-spectral images, with ground sampling distance as approximately 35 cm. 26 images are collected between 2014 and 2016 over Jacksonville, Florida, and 43 images are collected between 2014 and 2015 over Omaha, Nebraska, United States. In our experiment, epipolar rectified stereo pairs from challenge track 2 are used, with pairwise ground truth disparity images generated using airborne LiDAR data. For evaluation, we only consider the reconstructed stereo geometry, ignoring the semantics information. SGM, SGM-ForestS and SGM-ForestM are applied on 150 stereo pairs randomly selected from Jacksonville data. Due to the data inconsistency between the stereo images and LiDAR point clouds, the random forest is trained on ETH3D datasets for SGM-ForestS and SGM-ForestM. Thus, the robustness of the proposed algorithm is also tested when different data sources are used for training and validation.

The validation accuracy using 3 pixels as the upper limit of the allowed error for SGM,

SGM-ForestS, and SGM-ForestM are 66.06%, 61.36%, and 67.18%, respectively. With different datasets to train the random forest, the performance of SGM-ForestS is limited and even surpassed by SGM. The reason is the poor inference of the random forest when data different from the training sets are fed as input. However, SGM-ForestM is capable of providing more reliable scanline prediction, which is consistent with our demonstration in Figure 4.12 and 4.13. Therefore, it performs the best. Some visualization results are displayed in Figure 4.17. The reference LiDAR data were collected several years before the satellite images. Therefore, the images containing stable objects, e.g. buildings, are selected for visualization and evaluation. It is found that SGM-ForestM is capable of better recovering the roads and buildings (as highlighted by the red rectangles).

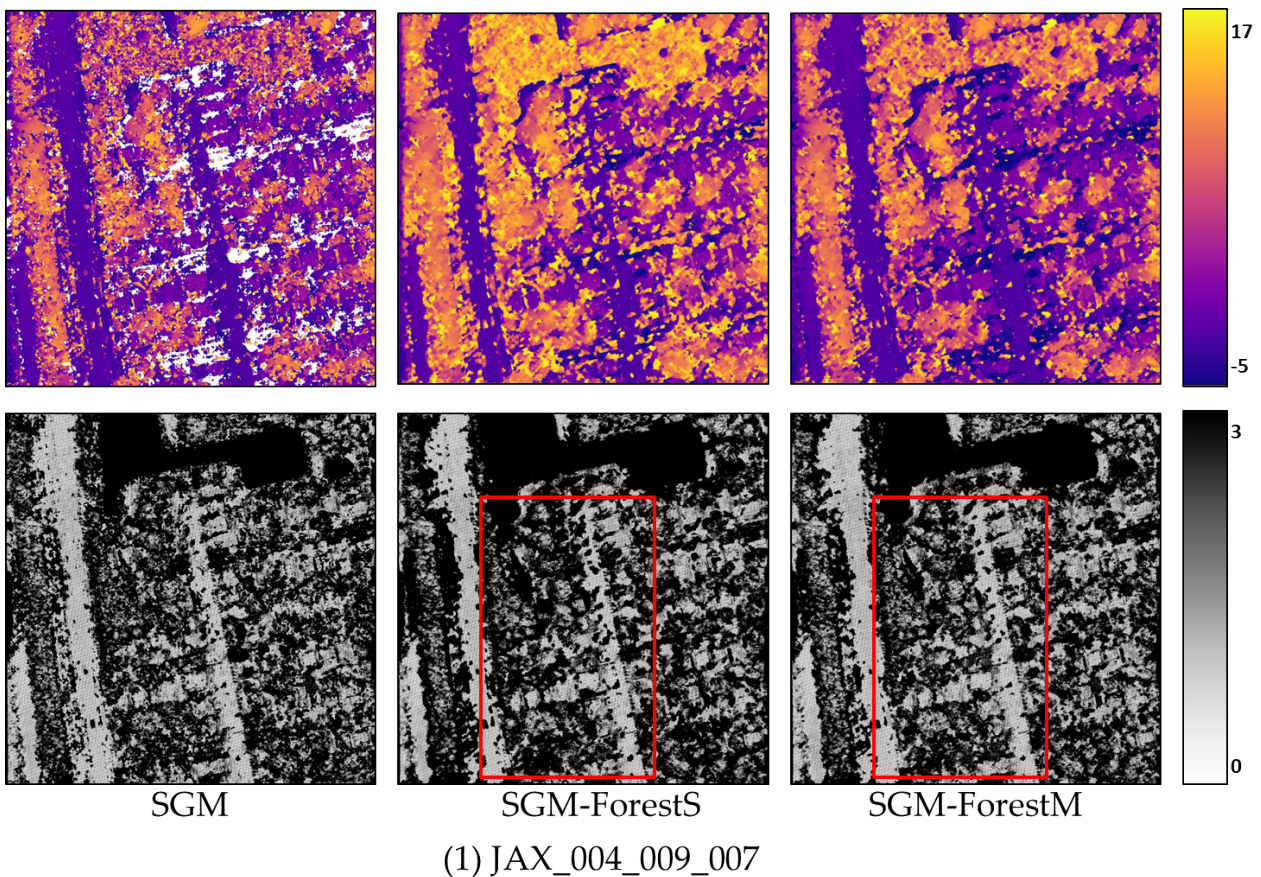


Fig. 4.17. Results on stereo datasets from the 2019 IEEE GRSS data fusion contest (Track 2, pairwise semantic stereo challenge). The disparity and error maps are displayed, with the red rectangles highlighting the performance difference between SGM-ForestS and SGM-ForestM.

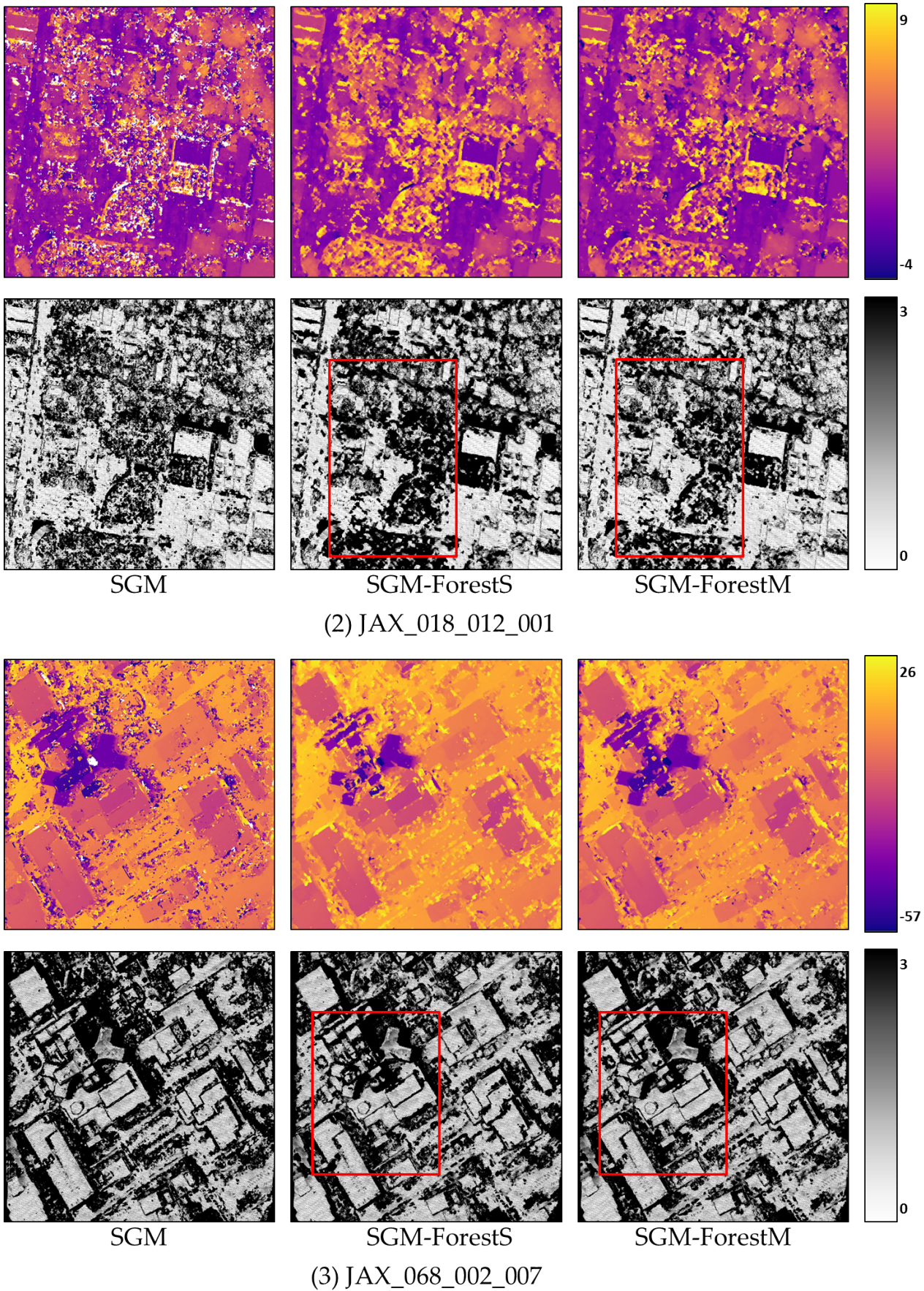


Fig. 4.17. Results on stereo datasets from the 2019 IEEE GRSS data fusion contest (Track 2, pairwise semantic stereo challenge). The disparity and error maps are displayed, with the red rectangles highlighting the performance difference between SGM-ForestS and SGM-ForestM. (cont.)

### 4.2.6 Conclusion and Outlook

SGM combines local and global stereo methods to locate correspondences and approximate 2D smoothness via multiple 1D scanline optimizations. Thus, a good compromise between accuracy and efficiency is obtained. However, adaptive scanline selection for disparity estimation is essentially more reasonable than the empirical scanline summation exploited in SGM, considering the varying performance among each scanline. In this research, we propose SGM-ForestM for scanline selection, as an extension of SGM-ForestS based on a multi-label classification strategy. SGM-ForestS trains a random forest aiming at finding the best scanline, the prediction of which is used as a baseline for further disparity estimation. On the other hand, we collect all the promising scanlines, given that normally more than one scanline is capable of predicting the correct disparity. We test the method on several datasets from close-range imagery, to airborne and satellite data. The results indicate that SGM-ForestM performs better almost in all cases, since it reconstructs the ill-posed regions more reasonably, for example, textureless areas, reflective surfaces, and so forth. It is found that the inference of the random forest is improved when using the proposed multi-label scheme, leading to improvements between 0.5% to 2.3%, depending on the benchmark used.

In future work, the idea of adaptive scanline selection can be embedded to other stereo matching systems as a further optimization step, such as the Sgm-nets (Seki and Pollefeys, 2017). Furthermore, self-supervision is promising as the random forest has low demand on the number of training samples. A rigid standard can be set to exclude outliers for a reliable supervision.

## 4.3 End-to-End Hierarchical Disparity Estimation and Refinement

Convolutional neural networks supervise a model to learn deep feature to better express the data using reference training samples, thus achieving a great success when applied in specific tasks. In the field of dense matching, the top ranking methods on vision benchmarks such as KITTI (Geiger et al., 2012; Menze and Geiger, 2015), ETH3D (Mayer et al., 2016), etc., are mostly end-to-end neural networks which are trained to predict dense disparity maps directly from stereo imagery. The model could simulate traditional matching schemes via differentiable layers and adaptively integrate each module to construct the processing pipeline. It is, therefore, promising to exploit robust algorithms from computer vision to process remote sensing stereo data to deliver better geographic products. However in practice, a well-performed model in close-range domain may struggle to handle aerial and satellite stereo images, considering the large data amount and baselines. Days of runtime and gigabytes of GPU memory could be consumed. In this research, we aim to adjust a state-of-the-art stereo matching network, Guided Aggregation Network (GA-Net), via a pyramid architecture for efficient disparity prediction from coarse to fine. Starting from a downsampled stereo input, the disparity is estimated and continuously refined through the pyramid levels until the original resolution is recovered. Thus, the depth search is only applied for a small size of stereo pair and then confined within a short residual range for minor correction, leading to highly reduced memory usage and runtime. We successfully process remote sensing datasets with very large disparity ranges, which could not be processed with the GA-Net due to GPU memory limitations. Tests on close-range, aerial and satellite data demonstrate that the proposed algorithm achieves significantly higher efficiency and comparable results with GA-Net on remote sensing data.

### 4.3.1 Background

Semi-Global Matching (SGM) (Hirschmüller, 2005) acquires dense correspondences via a simple pixel-wise cost comparison under a disparity searching range. The (piece-wise) smoothness of the reconstructed surface is guaranteed by enforcing each neighboring pixel to have similar disparity estimation. By repeatedly considering neighbors along multiple 1D scanlines (normally 8 or 16 symmetric paths), 2D regularization is realized. As more high-quality, high-resolution data becomes available, the computational cost of dense matching rises exponentially, especially in the field of remote sensing. To limit the memory usage and runtime, Rothermel (2017) proposed tSGM. Images are firstly downsampled to several scales constituting a pyramid structure, in which SGM is applied from the lowest resolution to the highest, level by level. On the pyramid top, the disparity range is downscaled accordingly together with the image size, leading to reduced workload. The matching result is then passed to the next higher resolution level as an initial prediction, from which a small disparity buffer is set as a new search range to locally refine the estimation. The coarse-to-fine scheme, thus greatly reduces the demand for memory and runtime. Besides, the influence of ambiguous disparity candidates is limited. Additionally, this strategy enables the use of deep learning based algorithms, which typically only support small search ranges due to memory limits, on datasets with large disparity ranges of sometime several thousand pixels, as typically occurring in extreme mountainous regions, such as the Himalayas.

GA-Net approximates SGM as a differentiable Semi-Global Guided Aggregation (SGA) layer, to construct an end-to-end neural network for stereo matching (Zhang et al., 2019). All the user-defined parameters in SGM can be learned, thus the smoothness requirement is satisfied in a smarter way depending on the specific scene situation. With SGA and only a few 3D convolutional layers to regularize the cost volume, the algorithm is more efficient than other networks, e.g. GC-Net (Kendall et al., 2017), PSMNet (Chang and Chen, 2018), etc, and achieves state-of-the-art performance. For processing high resolution remote sensing data, however, the training and prediction are still memory- and time-consuming (days are needed for training on patches of  $384 \times 576$ , with  $[0, 192]$  as the disparity search range, consuming around 15 GB GPU memory for each batch). Hence, we naturally refer to the pyramidal strategy of tSGM, for modifying GA-Net (termed GA-Net\_Ori for the follow-up) towards a pyramid architecture, and propose our GA-Net\_Pyramid. The efficiency is significantly enhanced with moderately decreased accuracy especially for remote sensing data. As mentioned above, the proposed method is tested on large scale aerial/satellite stereo data. The experiments prove the advantage of our strategy consistently. In addition, we also test our methods on close-range benchmarks, Scene Flow (Mayer et al., 2016) and KITTI-2012 (Geiger et al., 2012), to fill the domain gap. When the complexity of the target scene increases, our pyramid architecture is still more efficient, however, a reasonable decrease of the accuracy happens due to the lose of details and edges through the downsampling-upsampling processing.

### 4.3.2 Differentiable Approximation of SGM

In SGM, the scanline optimization technique (Scharstein and Szeliski, 2002) is applied through multiple scanlines simultaneously along several canonical directions, to satisfy the spatial 2D smoothness and avoid streaking problem. Along a certain scanline traversing in direction  $r$ , the cost for a pixel located at the image position  $p$  assuming  $d$  as the disparity, is calculated as:

$$L_r(p, d) = C(p, d) + \min ( L_r(p - r, d), L_r(p - r, d - 1) + P_1, \\ L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2 ). \quad (4.27)$$

$P_1$  and  $P_2$  are defined for penalizing the prediction when the previous neighboring point  $p - r$  prefers a different disparity value. In practice, however, two problems exist. Firstly, the users need expertise to determine appropriate  $P_1$  and  $P_2$  to punish neighboring disparity inconsistency. Tuning of  $P_1$  and  $P_2$  additionally depends on scene structure and the used similarity measure. Moreover, the values of  $P_1$  and  $P_2$  are fixed throughout the stereo processing or simply adapted according to, e.g. pixel gradients, which are not optimal for all the pixels within the image, especially under a varied scene structure, e.g. from plains to mountains. Therefore, GA-Net\_Ori introduces the SGA layer to address the issues, which is a differentiable approximation of Equation 4.27. Specifically, the master epipolar image provides guiding information through a sub-network to better penalize depth discontinuity, and enable a self-adaptive parameter setting. Thus, the penalty terms for conflicting neighboring disparities are determined according to the pixel location and scanline direction, which is more reasonable for smoothness regularization. Via the guidance sub-network, a weight is supplied for each term in Equation 4.27 to constitute the following equation:

$$\begin{aligned} L_r(p, d) = & C(p, d) + \text{sum} ( w_1(p, r) \cdot L_r(p - r, d), \\ & w_2(p, r) \cdot L_r(p - r, d - 1), w_3(p, r) \cdot L_r(p - r, d + 1), \\ & w_4(p, r) \cdot \max_i L_r(p - r, i) ). \end{aligned} \quad (4.28)$$

Compared with Equation 4.27, the punishment from  $P_1$  and  $P_2$  is replaced by the relative importance (weight)  $w_i$  of each term, which is predicted separately for each pixel along a directed scanline. Besides, there are two differences with SGM, one of which is that the first/external minimum operation is substituted by a weighted sum. This can be regarded as a replacement from a max-pooling layer to a convolution with strides, which is proven effective without accuracy loss (Springenberg et al., 2015). In addition, the second/internal minimum search is changed to a maximum, which embodies the learning target to maximize the probability at the ground truth disparity rather than minimizing the cost. To avoid the exploding accumulation of  $L_r(p, d)$  along the scanline,  $C(p, d)$  is also included within the weighted summation, with the sum of all the weights equal to 1. Thus, SGA is finally formulated as:

$$\begin{aligned} L_r(p, d) = & \text{sum} ( w_0(p, r) \cdot C(p, d), w_1(p, r) \cdot L_r(p - r, d), \\ & w_2(p, r) \cdot L_r(p - r, d - 1), w_3(p, r) \cdot L_r(p - r, d + 1), \\ & w_4(p, r) \cdot \max_i L_r(p - r, i) ), \end{aligned} \quad (4.29)$$

$$\sum_{i=0,1,2,3,4} w_i(p, r) = 1.$$

In SGM, the cost  $L_r(p, d)$  from each scanline is simply summed up to approximate 2D smoothness, which is demonstrated not reasonable for incurring inferior scanlines (Schönberger et al., 2018; Xia et al., 2020). Accordingly, GA-Net\_Ori takes the maximum as  $L(p, d) = \max_r L_r(p, d)$  to keep the best information.

The guidance sub-network also provides weights for another layer, LGA, to further filter the cost volume as below:

$$\begin{aligned} L_*(p, d) = & \text{sum} \left( \sum_{q \in N_p} w_0(p, q) \cdot L(q, d), \right. \\ & \sum_{q \in N_p} w_1(p, q) \cdot L(q, d - 1), \\ & \left. \sum_{q \in N_p} w_2(p, q) \cdot L(q, d + 1) \right), \end{aligned} \quad (4.30)$$

$$\sum_{q \in N_p} w_0(p, q) + w_1(p, q) + w_2(p, q) = 1,$$

from which a 3D neighborhood (in both spatial and disparity dimensions) centered around each pixel within the cost volume is utilized for a weighted average to protect thin structures. Afterwards as suggested by Kendall et al. (2017), a softmax operation  $\sigma(\cdot)$  is applied to

the filtered cost volume in order to acquire a probability for each disparity candidate (from  $[0, D_{max}]$ ) and regress the final result as:

$$\hat{d} = \sum_{d=0}^{D_{max}} d \times \sigma(-L_{*d}). \quad (4.31)$$

### 4.3.3 Efficiency Enhancement via Coarse-to-Fine Strategy

SGM’s scanline optimization scheme is adapted by GA-Net.Ori via an end-to-end neural network, from which the disparity of each pixel can be estimated with the support from neighboring pixels along multiple paths. The use of SGA and LGA layers is computationally more efficient than convolution based encoder-decoder, leading to superior efficiency of GA-Net.Ori than most state-of-the-art methods (Kendall et al., 2017; Chang and Chen, 2018). However it can still take days to train a well performed model, when the computational power is limited. In our case for example, the training on the Scene Flow dataset (patch size  $384 \times 576$ ), which is normally used for the initial learning phase, takes around 12 days to finish 8 epochs on two Quadro P6000 GPU cards. Hence, the employment of the network is hampered. In the field of remote sensing, it can be imagined that GA-Net.Ori would struggle to process high resolution aerial or satellite stereo data, especially for wide baseline stereo pairs requiring larger disparity search ranges.

Therefore, we refer to tSGM (Rothermel, 2017), and restructure GA-Net.Ori with a pyramid architecture to regress the depth from coarse to fine. Figure 4.18 presents the schematic overview of our GA-Net.Pyramid. Three pyramid levels are depicted which could be extended. We use the same stacked hourglass module (a double U-Net structure) as GA-Net.Ori, which is essentially a Siamese network (Bromley et al., 1993) for symmetric feature extraction from the left and right image, respectively. The input of the feature extraction module, however, is a stereo pair downsampled in accordance with the pyramid level. Afterwards, the cost volume is generated and then processed by SGA and LGA for disparity regression, in order to guide the subsequent level for the disparity refinement until the original resolution is recovered.

In Figure 4.18, the stereo images are processed from the pyramid top after downsampled by a factor of 4 along both row and column directions in our implementation (termed as ‘Scale 1/4’). Then the feature is extracted to construct a 4D cost volume by concatenating the left and right feature maps along the channel dimension, with a horizontal shift indicated by a disparity candidate within the search range. Assuming the cost volume on the original full resolution image is in size of  $H \times W \times D_{max} \times 2C$ , for the image height, width, the maximum disparity, and twice the channel number of the generated feature maps, respectively, our cost volume on the pyramid top reaches a highly reduced dimension as  $H/4 \times W/4 \times D_{max}/4 \times 2C$ . Thus, the memory consumption and computational complexity are decreased by a factor of 1/64.

Afterwards, the cost volume enters the cost aggregation block containing SGA and LGA layers, for which the guiding information is obtained from the downsampled master epipolar image. At last, the filtered cost is used for the following disparity regression as GA-Net.Ori. Thus, a disparity map of the downsampled image ‘Scale 1/4’ is obtained for the pyramid top. From here, the depth of the scene is already roughly estimated and the large scale context is perceived, which provide a good guidance for the following processing. Based on the prediction of the pyramid top, the other levels thus only need to locally refine the disparity values. Therefore, the disparity map from ‘Scale 1/4’ level is upsampled by a factor of 2 via bilinear interpolation, to match the resolution of ‘Scale 1/2’ level as an initial estimation. Afterwards, we warp the right feature maps, pixel by pixel, according to the dense correspondences indicated by the disparity map, and acquire a synthetic left feature. Assuming



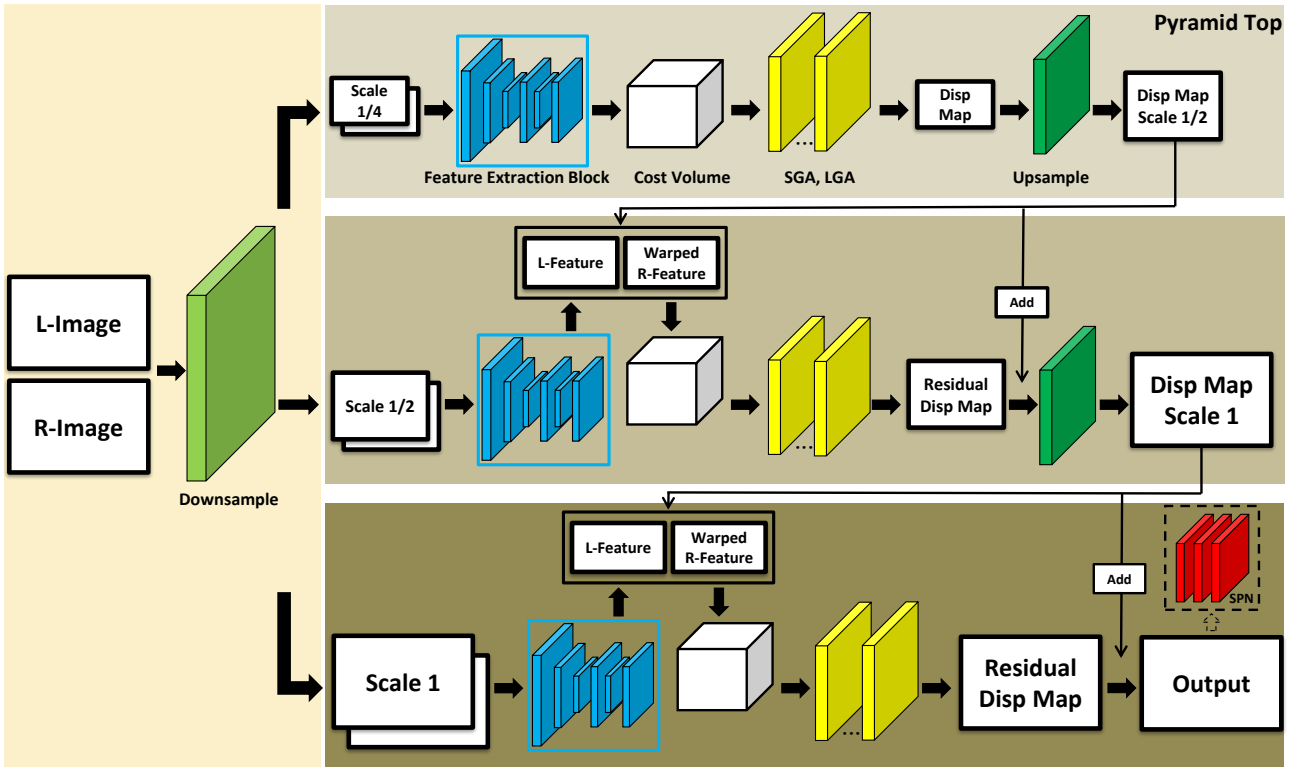


Fig. 4.18. GA-Net\_Pyramid with explicit downsampling. The input stereo pair is downsampled explicitly according to the resolution required by each pyramid level. At the pyramid top, the stereo correspondences are located within an absolute disparity range in low resolution. The following pyramid levels perform disparity refinement within a pre-defined residual disparity range until the original resolution is recovered at the pyramid bottom. (SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement.)

the disparity map from 'Scale 1/4' level is accurate enough, the synthetic and the original left feature maps would perfectly match each other. However, considering the details lost through downsampling and the corresponding matching error in practice, a small shift exists between the left and the warped right feature, which we call the disparity residual. Accordingly, a cost volume is built in size of  $H/2 \times W/2 \times 2disp\_resi \times 2C$  for 'Scale 1/2' level. In contrast to the cost cube on the pyramid top, the height and width are increased to match the current resolution, however, the disparity range is confined to a small buffer.  $disp\_resi$  is a pre-defined threshold, leading to a range  $[-disp\_resi, +disp\_resi]$  around the initial disparity estimation to locate the residual for refinement. For each pixel in the reference/left frame, a positive residual suggests a further shift towards the left in the target/right frame, from the current predicted matching position. On the contrary, a negative residual expects a decreased displacement along the row for a better correspondence. Then the cost volume is regularized by SGA and LGA, and the residual disparity map is calculated. The disparity estimation for the current level is obtained by adding the residual and the previously upscaled disparity map.

The stereo pair on 'Scale 1/2' level is twice larger in height and width, however, the search for correspondences is restricted within a narrow range. Hence only a small overhead is accumulated. We apply the same procedure for the remaining pyramid level, to continuously improve the disparity estimation until the original resolution is reached. Each pyramid level only requires the input epipolar imagery at its level and the disparity image of the previous level. For an efficient and memory saving implementation during disparity estimation, computation of the levels could be decoupled to significantly lower the memory footprint while allowing large input image sizes. Compared to GA-Net\_Ori, it is thus feasible to significantly increase both image size and disparity range, as only the pyramid top needs to process the full disparity search range, for example processing of images with a four times larger width, height and disparity range are possible without additional GPU memory requirements in

this case. Note that the efficiency evaluation in this research is performed without adding these optimizations.

Regarding the loss calculation during the training, we use the same smooth  $L_1$  loss function as GA-Net\_Ori. However, our pyramid architecture predicts more than one disparity map, which should all be considered to allow for intermediate supervision. Hence, a weight is assigned to each pyramid level for a weighted loss summation as:

$$L = \sum_{i=1}^N l(|\hat{d}_i - \bar{d}|) \cdot \omega_i, \quad (4.32)$$

in which  $\hat{d}_i$  denotes the disparity predicted by the pyramid level  $i$  (starting from 1 as the pyramid top), and  $\bar{d}$  is the corresponding ground truth.  $l$  computes the smooth  $L_1$  loss from the disparity difference. A weight  $\omega_i$  is assigned to the level  $i$  for a weighted summation through all  $N$  pyramid levels. The disparity map from each level is upsampled to the original full resolution before computing the loss. As the estimation is improved from the pyramid top to the bottom, the corresponding weight is also increased.

In order to achieve efficient and robust estimation on cross-domain datasets, we design different feature extractors and observe the corresponding performance, so that an appropriate model could be used to handle specific data types. The architecture in Figure 4.18 simply applies GA-Net\_Ori in a pyramidal manner, which takes the linearly downsampled stereo pair as input to extract feature for further processing. Therefore, we propose another architecture to implicitly learn the downsampled feature, as displayed Figure 4.19, such that both explicit and implicit image downsampling strategies are tested.

Instead of downsampling the input stereo pair level by level, we only use the stacked hour-glass module once to extract feature from the original (full resolution) images for feeding all the pyramid levels. In order to keep both geometric context and local details within the feature maps, the input images are firstly downsampled via convolutions with stride two, and then deconvolved to gradually recover the resolution, in which a skip connection is exerted between corresponding feature maps of the encoder and decoder at the same resolution. Before reaching the original size, we directly extract the intermediate feature maps from the decoder to feed each level, as long as the expected resolution is acquired. Then regarding the guidance sub-network, the weights of SGA and LGA are first computed for the pyramid bottom according to the original master frame. Afterwards, we apply convolutions with strides to obtain the guiding weights of the subsequent lower resolution level until reaching the pyramid top. To differentiate the GA-Net\_Pyramid with explicit and implicit downsampling, we name the two variants as GA-Net\_PyramidED and GA-Net\_PyramidID, respectively.

As the disparity is estimated and refined through the pyramid, we add a Spatial Propagation Network (SPN) (Liu et al., 2017) as a post-processing step to explore its influence on the matching results. SPN is capable of sharpening the object boundaries, by learning from the source image (in our case, the master epipolar image) in a data-driven mode, which is appropriate as a further refinement in our pyramid architecture especially for close-range data with rich details. Hence, four models are finally proposed including GA-Net\_PyramidED and GA-Net\_PyramidID, respectively, with or without SPN added at the end of the pyramid bottom.

#### 4.3.4 Performance Evaluation on Multiple Data Sources

In the experiments, we compare our GA-Net\_Pyramid with GA-Net\_Ori using cross-domain datasets including close-range, Scene Flow and KITTI-2012, aerial, and satellite stereo data.

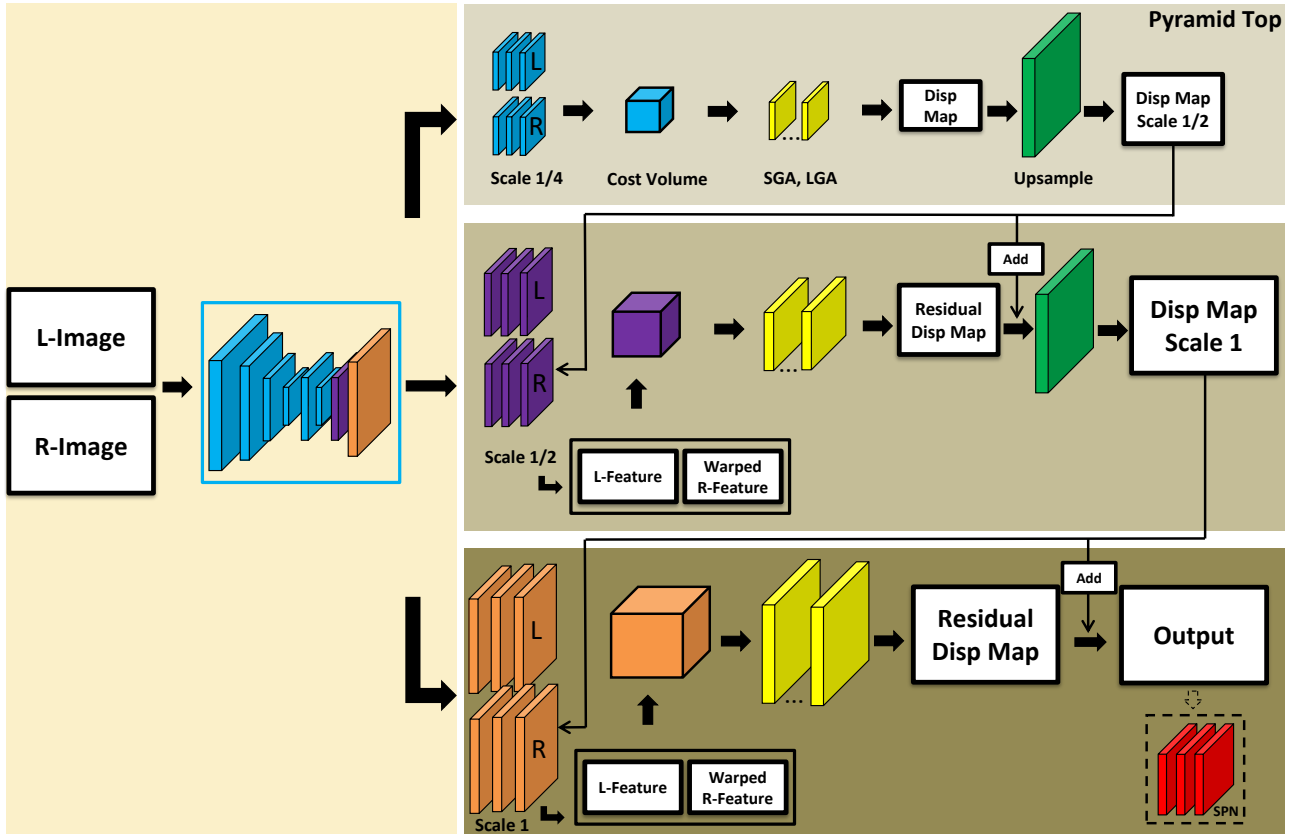


Fig. 4.19. GA-Net\_Pyramid with implicit downsampling. The feature extractor is applied on the stereo pair in original resolution, with the intermediate feature maps from its decoder to feed each pyramid level according to the expected resolution. Thus, an implicit downsampling is achieved. (SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement.)

For a fair comparison, the implementation details are rigidly controlled between the two algorithms. Regarding the training, we use the same patch size with a pre-defined disparity search range, to train the networks for certain epochs, based on Adam optimization strategy (Kingma and Ba, 2017). Each stereo pair is normalized, according to the mean and standard deviation of the pixel values from each channel, before feeding to the network. As suggested in Zhang et al. (2019), the SGA is applied along four directions (horizontally and vertically) for both GA-Net\_Pyramid and GA-Net\_Ori.

For GA-Net\_Pyramid specifically, the number of pyramid levels is 3 and the search range for the disparity residual after the pyramid top is set as  $[-6, +6]$  to refine the matching results. Details about the pyramid setting are discussed. We apply 3 SGA and 2 LGA layers to regularize the cost volume on our pyramid top, which is the same as GA-Net\_Ori. With regard to the other pyramid levels, only 1 SGA layer (and 2 LGA layers) is utilized due to the small disparity search range. The weight is set as 0.25, 0.5 and 1, to the pyramid level 1 (top), 2 and 3 (bottom), respectively, to calculate the final loss in Equation 4.32. The implementation of the methods is based on Python and Pytorch.

As for the close-range datasets, the scene structure is relatively complicated with rich details. Referring to most learning based dense matching algorithms, we train the models on Scene Flow data from scratch, and utilize real data, KITTI-2012 in our case, for finetuning. Scene Flow is a synthetic dataset via randomly combining human-made objects with backgrounds from real images, which is used by most stereo networks for initial training. Afterwards, only a small dataset from a specific field is sufficient to adjust the model into practical scenarios. The dataset contains three subsets, namely FlyingThings3D, Monkaa and Driving, including around 35000 images for training and 4370 images for validation. KITTI-2012 is a stereo dataset with a focus on outdoor street views, which is normally ap-

plied in the field of autonomous driving. The dataset includes 194 training and 195 test stereo pairs, with ground truth disparity maps based on LiDAR measurements provided or withheld.

Both the pre-trained and finetuned models are tested on the corresponding dataset. Regarding the former, the whole Scene Flow training dataset is used for training (8 epochs), while only 1000 stereo pairs from its validation set are selected for test to save time. On the other hand, 170 images from KITTI-2012’s training data are exploited to finetune the models for 800 epochs, with the remaining 24 images for validation. All the data selection is random, so that a fair evaluation is achieved. Through the entire training period, we use the same patch size (384×576) with the maximum disparity set to 192. The networks are trained with a batch size of two, on two Quadro P6000 GPU cards.

The quantitative and visual comparison between our pre-trained pyramid models and GA-Net\_Ori is shown in Table 4.14 and Figure 4.20. As indicated by the table, we calculate the percentage of pixels, for which the estimation error is smaller than 1, 2, and 3 pixels, respectively, and the end point error (EPE) for accuracy evaluation. Regarding the efficiency, the runtime and GPU memory consumption are reported. For all the experiments in this research, the runtime in test period is counted for processing the whole test dataset. Specifically, we generate a binary file to save the disparity value of each correspondence, and a png (Portable Network Graphics) file to visualize the result. In the tables, M denotes megabytes for the GPU memory consumed by each network, while the time spent in training and test is expressed in hours (h) or seconds (s). Better performance is highlighted in bold.

Table 4.14. Accuracy and efficiency comparison between GA-Net\_Pyramid, including GA-Net\_PyramidED and GA-Net\_PyramidID, and GA-Net\_Ori on Scene Flow data.

	Accuracy				Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	EPE	Memory	Runtime	Memory	Runtime
GA-Net_PyramidED	81.77%	88.59%	91.42%	1.61	<b>7052M</b>	38.25h	<b>2761M</b>	<b>0.39h</b>
GA-Net_PyramidED+SPN	83.04%	89.97%	92.67%	1.44	7140M	40.62h	<b>2761M</b>	<b>0.39h</b>
GA-Net_PyramidID	81.26%	89.10%	92.05%	1.49	7264M	<b>30.07h</b>	3501M	0.40h
GA-Net_PyramidID+SPN	84.27%	91.09%	93.64%	1.23	7422M	31.69h	3501M	<b>0.39h</b>
GA-Net_Ori	<b>91.41%</b>	<b>95.35%</b>	<b>96.60%</b>	<b>0.86</b>	30464M	280.53h	6983M	2.10h

From the results, it is found that GA-Net\_Ori outperforms the two pyramid models in accuracy, however, the latter consume much less memory and runtime usage in both training and test periods. In case of the close-range data, the objects are captured under an ideal viewing condition, thus very high resolution is achieved with plenty of details and texture information contained. Besides, as Scene Flow is a synthetic dataset, the random arrangement of man-made objects makes the scene non-natural, non-logical and highly complicated with many occlusions. Hence, our GA-Net\_Pyramid is surpassed by GA-Net\_Ori, considering the information loss due to a sequence of downsampling-upsampling through the pyramid levels. On the other hand, our hierarchical strategy highly simplifies the problem complexity, via refining the disparity estimation from coarse to fine. Therefore, the stereo matching procedure can be finished using far less computational source but at a much higher speed. Between the two pyramid models, GA-Net\_PyramidED and GA-Net\_PyramidID, similar accuracy is obtained. Regarding the SPN processing, a positive effect is achieved for both pyramid structures, while GA-Net\_PyramidID could be improved by a larger extent. Since that the experiments are implemented on a server open to multiple users, the runtime of each model could be slightly influenced by unknown processes. We recommend referring to the training time to evaluate the speed of the algorithms, especially for each pyramid model with similar efficiency, considering the relatively long training process compared with the test period. GA-Net\_PyramidID is faster than GA-Net\_PyramidED, since the feature extrac-

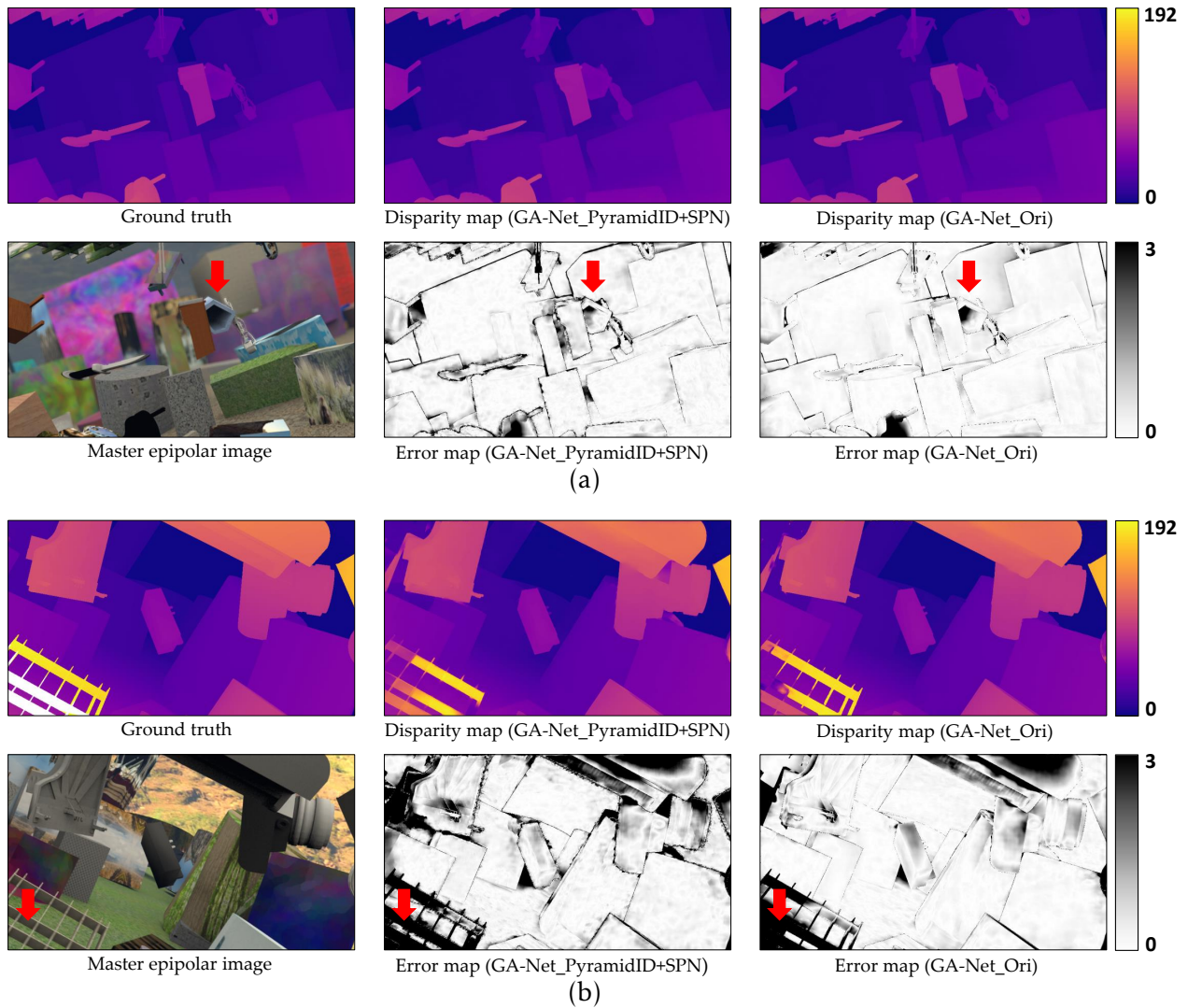


Fig. 4.20. Visual comparison between GA-Net\_PyramidID+SPN and GA-Net\_Ori on Scene Flow data. In each subfigure, the disparity maps from the ground truth and each network are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. Regions where the proposed algorithm outperforms GA-Net\_Ori are marked with red arrows.

tion in former case is applied only once on the full resolution stereo pair, rather than repeatedly learning from the corresponding downsampled images level by level. In case of the GPU memory consumption, GA-Net\_PyramidED performs better.

As for the figures, only the best performed pyramid model is visually compared with GA-Net\_Ori, e.g. GA-Net\_PyramidID+SPN on Scene Flow dataset. Accordingly, we display the master epipolar image, where the guidance information is acquired for SGA and LGA, the ground truth, and the corresponding results from each algorithm. The color bar at the end shows the disparity and error changes. In Figure 4.20, it is found that GA-Net\_Ori obtains a generally better disparity result than GA-Net\_PyramidID+SPN, with clear edges and more details included. However, our pyramid model still produces a disparity map in good quality, even including better depth results in certain regions. We discover that GA-Net\_PyramidID+SPN is capable of better reconstructing hollow-shaped objects, e.g. the barrel and the shelf as indicated by the red arrows. The finding is also supported by the following experiments on the KITTI dataset.

After finetuning the pre-trained models on part of KITTI-2012's training data, we test them on the remaining stereo pairs. In Table 4.15 and Figure 4.21, the corresponding quantitative and qualitative results are provided. Regarding the training efficiency, only the time

spent for finetuning is counted. Similar to the previous experiment, GA-Net\_Ori acquires the best accuracy, however, the pyramid models are faster and more memory friendly. SPN still improves the results of all the pyramid models, among which GA-Net\_PyramidID+SPN achieves the highest accuracy. It should be noted that, our GA-Net\_Pyramid performs better for real data leading to a further reduced accuracy gap compared with GA-Net\_Ori. From the visual inspection, the depth result of each algorithm is barely distinguishable. Moreover as mentioned before, we obtain a better depth prediction for hollow-shaped structure. KITTI-2012 doesn't provide ground truth for certain area, nevertheless, it is clear that our pyramid architecture gives a clean and more reasonable depth estimation for the regions marked by the red arrows.

Table 4.15. Accuracy and efficiency comparison between GA-Net\_Pyramid, including GA-NetPyramidED and GA-NetPyramidID, and GA-Net\_Ori on KITTI-2012 data.

	Accuracy				Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	EPE	Memory	Runtime	Memory	Runtime
GA-Net_PyramidED	86.54%	93.57%	95.76%	0.89	<b>7140M</b>	17.81h	<b>2641M</b>	28.07s
GA-Net_PyramidED+SPN	86.56%	93.53%	95.66%	0.88	7242M	18.49h	<b>2641M</b>	29.29s
GA-Net_PyramidID	83.20%	92.68%	95.12%	1.10	7546M	<b>13.77h</b>	3379M	<b>27.02s</b>
GA-Net_PyramidID+SPN	86.88%	94.13%	96.18%	0.83	7680M	15.02h	3379M	29.89s
GA-Net_Ori	<b>91.55%</b>	<b>96.64%</b>	<b>97.65%</b>	<b>0.60</b>	30514M	135.47h	6565M	165.72s

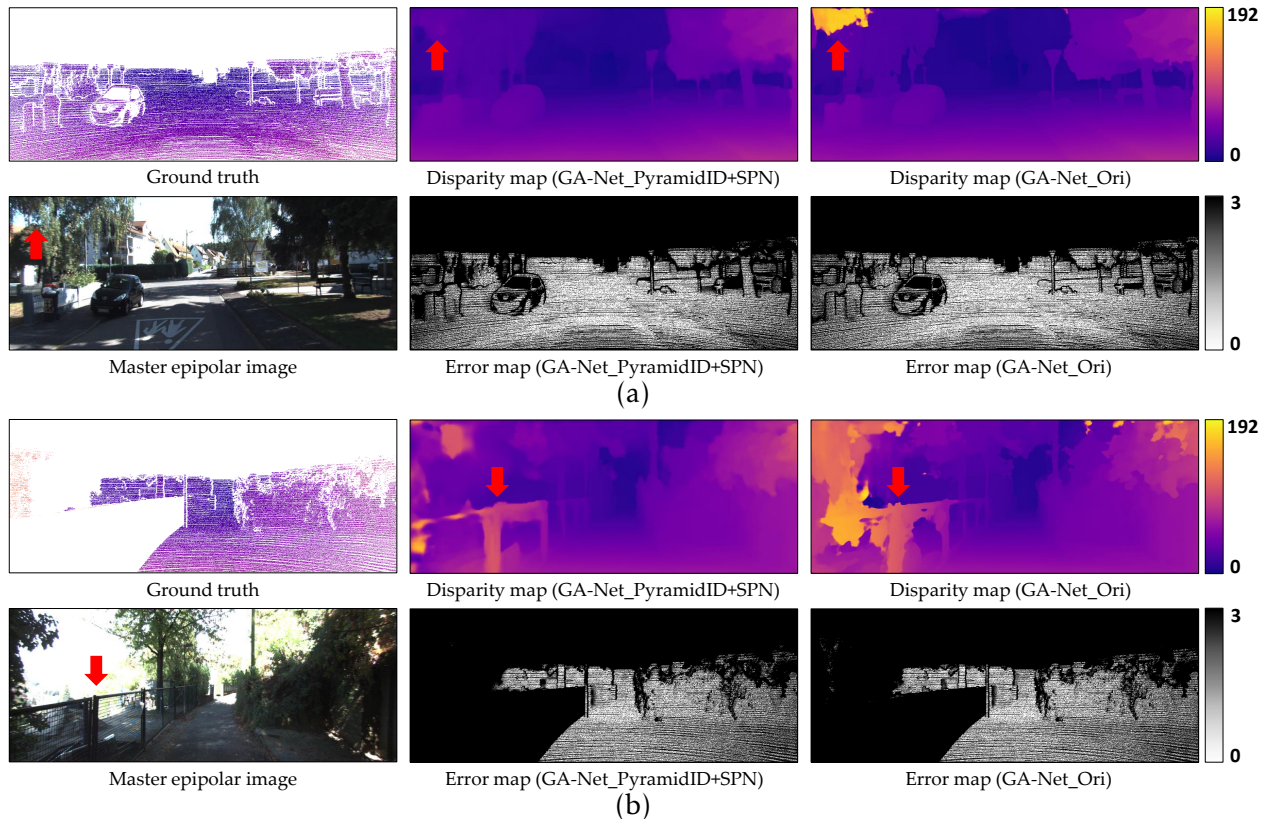


Fig. 4.21. Visual comparison between GA-Net\_PyramidID+SPN and GA-Net\_Ori on KITTI-2012 data. In each subfigure, the disparity maps from the ground truth and each network are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. Regions reconstructed better by the proposed algorithm are marked using red arrows.

Regarding the remote sensing scenarios, the networks are firstly tested on our aerial data. The airborne and satellite stereo processing is the target domain of this research, since the corresponding data is usually large in size and owns a much wider stereo baseline, which presents a higher demand on the algorithm's efficiency. The networks are trained on our

synthetic remote sensing data (854 stereo pairs) from scratch for 200 epochs. Our synthetic dataset is designed specifically for airborne and satellite stereo tasks. The dataset focuses on urban regions, via referring to six city models provided by the software CityEngine: Paris, Venice, New York, Philadelphia and two small development scenes. The models were exported and processed in Blender, to preserve the textures and relevant information. Afterwards, we used BlenderProc (Denninger et al., 2019) to render the dataset according to the geometry of the model, which included RGB images and the corresponding disparity maps. Considering both aerial and satellite platforms, the simulated camera for rendering was located at 200 m and 500 km above the cities, respectively. 854 stereo pairs in size of  $1024 \times 1024$  pixels were generated in total, with the ground sampling distance (GSD) ranging from 5 cm to 50 cm.

Afterwards, the models are finetuned on a subset (200 stereo pairs) of our aerial data for 100 epochs. As for our real aerial data, we use the 4K sensor system mounted on a helicopter for the data collection (Kurz et al., 2014). Three off-the-shelf Canon EOS cameras (one 1D-C and two 1D-X) constitute the imaging unit. The data contains geo-referenced images with a size of 17.9 megapixels, acquired over Gilching in the southwest of Munich, Germany. Equipped with 50 mm lenses looking in varying view directions, a field of view (FOV) up to  $104^\circ$  is reached. The flight height was 500 m above ground, enabling 6.9 cm nadir GSD. A multi-view stereo matching based on SGM was applied, in which the calculated heights (depths) from multiple highly overlapped images were fused to achieve a high quality digital surface model (DSM). The DSM was used to compute disparity maps for each stereo pair, which were utilized as reference data for finetuning and validation.

We randomly select another 20 aerial stereo pairs, possessing no overlap with the finetuning data, to test the trained models. Image patches in size of  $384 \times 576$  are randomly cropped for training, and the test images are  $1152 \times 1152$ . The data may contain negative or very large disparity values, hence we exclude the stereo pairs with large baselines in order to keep the disparity range processible by both GA-Net\_Pyramid and GA-Net\_Ori. Accordingly the disparity range is also set as  $[0, 192]$ . The models are trained with a batch size of two on two Quadro P6000 GPU cards.

In addition, SGM is utilized as a baseline model in our aerial and satellite experiments, since the algorithm is widely used in the field of remote sensing for dense reconstruction. We exploit Census (Zabih and Woodfill, 1994) to calculate the matching cost with a  $7 \times 7$  window. The penalty terms  $P_1$  and  $P_2$  are set to 19 and 33, respectively. The cost from 8 symmetric scanlines along horizontal, vertical and diagonal directions are accumulated to compute the disparity based on the WTA strategy, which is then further refined using a left-right consistency check.

In Table 4.16, the performance of each algorithm is recorded. We can firstly find that all the GA-Net models outperform the baseline SGM by a certain margin. Besides, our pyramidal revision leads to a very small accuracy decrease compared with the original structure, but highly improves the efficiency. Our GA-Net\_PyramidED (without SPN added) is the best performed pyramid model, which is only around 1% worse than GA-Net\_Ori in accuracy. Nevertheless, the pyramid models are about 8 and 7 times faster than GA-Net\_Ori, but only expends around 25% and 40% memory usage for training and prediction, respectively. It should be noted that for airborne data, SPN cannot improve the performance for either of the pyramid models, which is different from the close-range experiments. A visual comparison among the methods is provided in Figure 4.22.

We select two regions, one vegetation and one building area from the validation data for the visualization. It is shown that GA-Net\_PyramidED, as an intuitive modification of GA-Net\_Ori based on a pyramid architecture, archives good performance in airborne stereo matching. When the scene is relatively simple, containing fewer depth discontinuities and a

Table 4.16. Accuracy and efficiency comparison between GA-Net\_Pyramid, including GA-Net\_PyramidED and GA-Net\_PyramidID, and GA-Net\_Ori on aerial data (baseline model: SGM).

	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
GA-Net_PyramidED	77.28%	86.19%	89.70%	<b>7124M</b>	25.18h	<b>5623M</b>	<b>83.60s</b>
GA-Net_PyramidED+SPN	74.06%	86.08%	89.69%	7238M	26.19h	<b>5623M</b>	89.08s
GA-Net_PyramidID	76.35%	85.46%	89.14%	7544M	<b>20.59h</b>	6979M	84.02s
GA-Net_PyramidID+SPN	76.14%	84.82%	88.21%	7676M	21.54h	6979M	86.19s
GA-Net_Ori	<b>78.75%</b>	<b>86.99%</b>	<b>90.13%</b>	30512M	187.59h	15685M	616.74s
SGM	72.14%	75.89%	77.15%	—			

smooth depth change, the hierarchical estimation and refinement of disparity is capable of highly enhancing the efficiency, without a noteworthy sacrifice of the result’s quality.

To further understand our GA-Net\_Pyramid when applied in the field of remote sensing, we explore the impact of different pyramid architectures using our aerial data. Regarding the pyramid structure, two variants are the most important factors, the number of pyramid levels and the residual search range for disparity refinement. The main difference between GA-Net\_PyramidED and GA-Net\_PyramidID is the strategy to extract feature, which is not directly related to the above two factors. In addition, our two pyramid models achieve similar accuracy. Therefore, we select GA-Net\_PyramidED without SPN for post-processing to study the pyramid setting, since it is the more intuitive pyramidal modification of GA-Net\_Ori. As for the number of pyramid levels, we start from 2, since a 1-level GA-Net\_Pyramid will degenerate to GA-Net\_Ori, to 4 levels, with a fixed residual range  $[-6, +6]$ . The model is trained on our synthetic dataset from scratch and tested on the same validation data. We use the same hyperparameter setting as before, except that the size of the training patches changes to  $384 \times 768$  to facilitate the downsampling when more levels are applied. We train the model on one GPU card due to the less memory requirement of GA-Net\_Pyramid. The results are in Table 4.17.

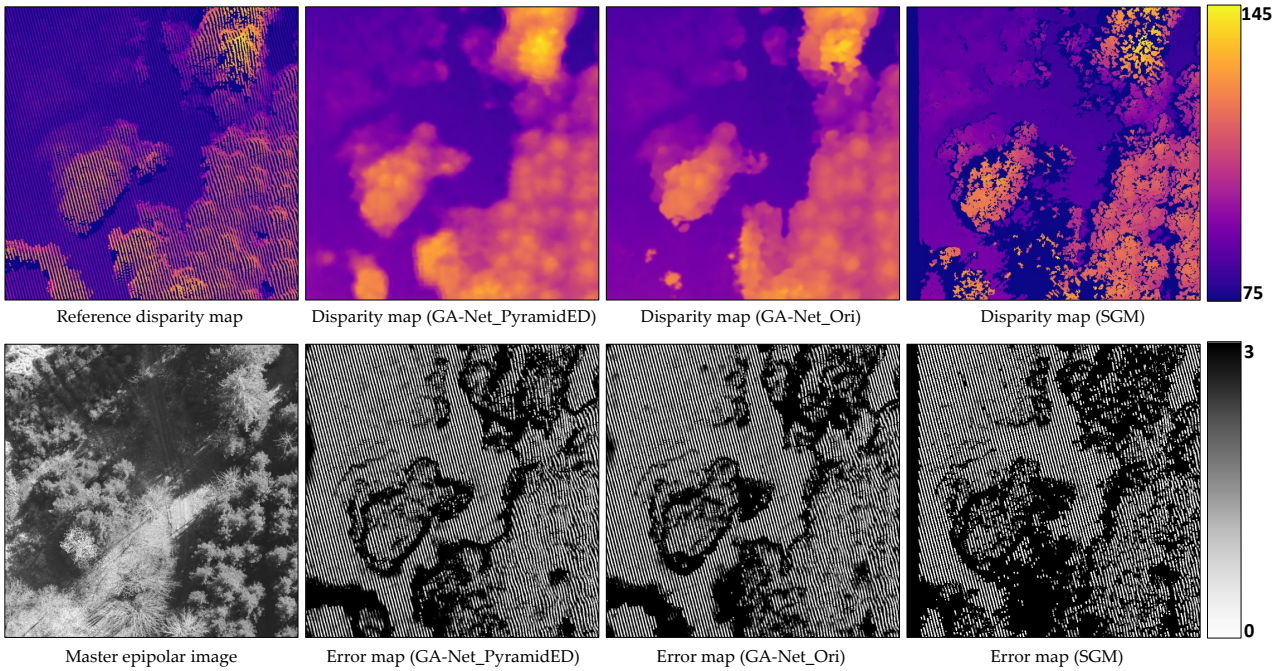
Table 4.17. Accuracy and efficiency comparison for GA-Net\_PyramidED with different pyramid levels.

Pyramid Levels	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
2	<b>72.38%</b>	80.89%	85.14%	11521M	70.25h	5813M	120.28s
3	72.17%	<b>81.22%</b>	<b>85.69%</b>	8121M	29.13h	5623M	82.11s
4	72.08%	81.19%	85.57%	<b>7647M</b>	<b>27.80h</b>	<b>5589M</b>	<b>63.92s</b>

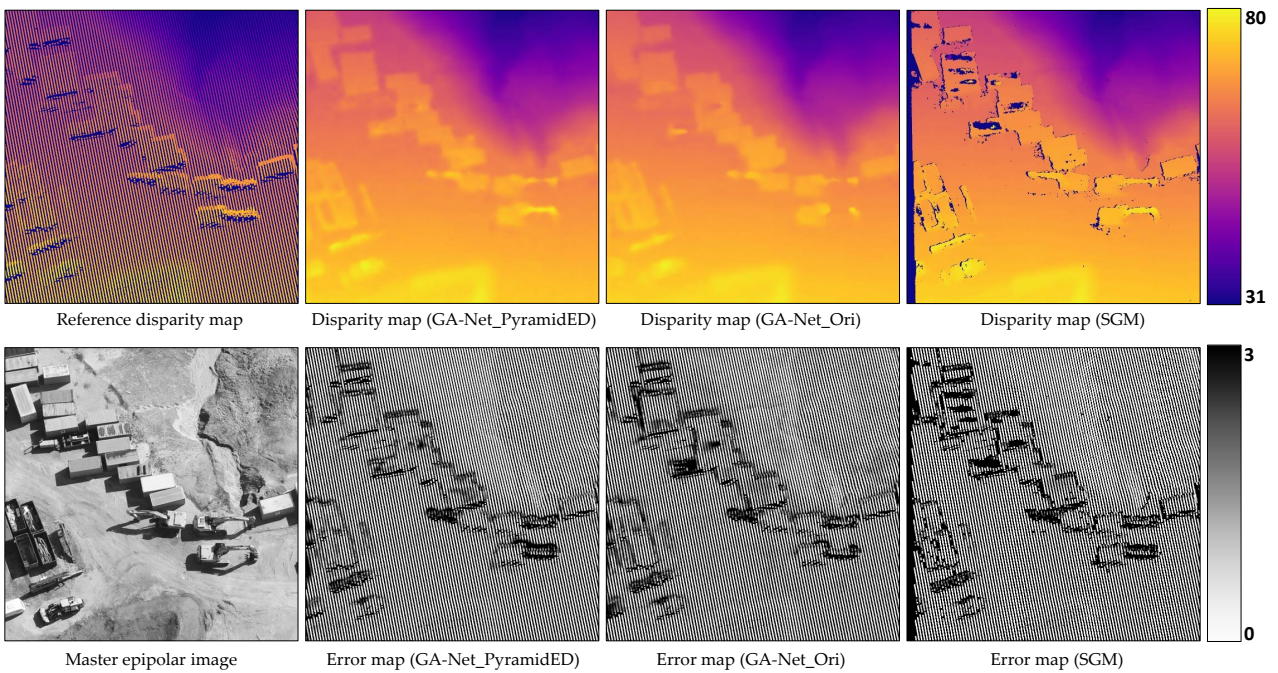
According to the table, it is found that the architecture with 4 pyramid levels acquires the best efficiency. However, with slightly increased memory and runtime, the model with 3 pyramid levels achieves better results. Along with GA-Net\_PyramidED regresses towards GA-Net\_Ori (from 3 to 2 levels), the efficiency drastically deteriorates as expected, nevertheless, without a noticeable improvement of the accuracy. Therefore, we determine to use the number of pyramid levels as 3. Then we adjust the residual search range to  $[-3, +3]$ ,  $[-6, +6]$  and  $[-12, +12]$ , respectively. The model is also trained from scratch on our synthetic dataset using one GPU card, and tested on the same 20 aerial images. We keep the training setting unchanged, except that the patch size is set back to  $384 \times 576$ . In Table 4.18, the performance for different residual search ranges is recorded.

Table 4.18 indicates that as the residual range becomes larger, the efficiency naturally decreases. Moreover, when the residual buffer expands over  $[-6, +6]$ , the accuracy cannot be





(a) A vegetation area



(b) A building area

Fig. 4.22. Visual comparison among GA-Net\_PyramidED, GA-Net\_Ori and SGM on aerial data. In each subfigure, the reference disparity map and each algorithm's stereo results are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps.

further enhanced. Hence, the structure of our pyramid is determined as 3 levels, with the maximum/minimum residual set as 6/-6. To keep the experiments consistent, the pyramid structure is used for both GA-Net\_PyramidED and GA-Net\_PyramidID in this research.

At last, we also test the proposed methods on satellite data. The flight campaign regarding our aerial images was performed during a Worldview-3 stereo acquisition of the same area. WorldView-3 is a very high resolution imaging satellite currently offering the most detailed publicly available spaceborne imagery, at a resolution of 30 cm. Due to the minimal time

Table 4.18. Accuracy and efficiency comparison for GA-Net\_PyramidED with different residual search ranges.

Residual Range	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
[-3,+3]	73.38%	81.95%	86.04%	<b>5941M</b>	<b>23.49h</b>	<b>5467M</b>	<b>55.23s</b>
[-6,+6]	<b>73.76%</b>	<b>82.21%</b>	<b>86.40%</b>	6283M	26.35h	5623M	84.50s
[-12,+12]	73.38%	82.11%	86.37%	7033M	34.96h	6489M	123.09s

difference of less than 1 hour of each aerial image from the satellite images, the higher resolution airborne data is well suited as reference data for the satellite stereo matching to finetune the models and evaluate the results. After bundle-adjustment of the satellite data with the aerial imagery and DSM as reference, we generated an epipolar rectified stereo pair using the algorithm implemented by the CARS stereo pipeline (Michel et al., 2020). Similar to the aerial imagery, a reference disparity map was calculated by projecting each point of the 4K DSM into the epipolar satellite stereo pair. The stereo pair has a dimension of  $20815 \times 28264$  pixels, which was cut into 98 tiles (in size of  $1152 \times 1152$ ) owning an overlap larger than 25% with the 4K data coverage. From them, 78 tiles were randomly selected for finetuning the pre-trained GA-Net models, with the other 20 image pairs as the validation data. As the airborne data was geo-referenced in two separate blocks using differential GPS and only few ground control points (GCPs), a slight height offset was found between the aerial and satellite data, yielding disparity differences between the aerial reference and the satellite stereo pair in the pixel range, but rising up to 4 pixels at the corner of one aerial block. Since these systematic differences strongly affected training and evaluation of the networks, a second order offset surface was fitted to the difference of the airborne reference disparity map and the satellite disparity map estimated by SGM, on each of the 98 tiles. The offset was added to the reference disparity map to remove the systematic bias.

The networks are also pre-trained on our synthetic remote sensing data for 200 epochs, and finetuned on the generated satellite training data for 150 epochs. The training conditions stay the same, including the patch size ( $384 \times 576$ ), disparity range ( $[0, 192]$ ), batch size (2), GPU usage (2 Quadro P6000 cards), etc. SGM is also used as a baseline.

In Table 4.19, we record the performance of GA-Net\_Pyramid, GA-Net\_Ori and SGM. Similar to the results of airborne data, GA-Net\_Ori achieves the highest accuracy, after which GA-Net\_PyramidED still acquires the best performance among all the other models. The 1 pixel accuracy of our GA-Net\_PyramidED, without SPN added for post-processing, is only surpassed by GA-Net\_Ori by 0.08%. However, the former is around 8 and 13 times faster than the latter, consuming only 23% and 36% GPU memory in training and test, respectively. In addition, GA-Net\_PyramidED performs better than GA-Net\_PyramidID, with less GPU memory consumption but longer training time. SPN also impairs the performance of the pyramid models, which is consistent with our experiments on aerial data. The visual comparison is in Figure 4.23, including a vegetation and a building area as well. It is found that both networks predict a smoother disparity map than SGM, with less erroneous estimation. Besides, similar results are obtained between our GA-Net\_PyramidED and GA-Net\_Ori, considering the reconstruction density and quality.

We also apply our pyramid network on a stereo pair with a large disparity range, in order to indicate the model’s ability to process large scale remote sensing data. The imagery is from WorldView-2 at a resolution of 50 cm, covering the Matterhorn mountain, Switzerland. We select a stereo pair with  $14^\circ$  conversion angle for which the disparity varies in range of thousand pixels, due to the very large ground height difference from 1800 m to 4478 m. The best performing model finetuned in our satellite stereo experiments, GA-Net\_PyramidED, is directly used for disparity prediction in this test, without fine-tuning on the mountain

Table 4.19. Accuracy and efficiency comparison between GA-Net\_Pyramid, including GA-Net\_PyramidED and GA-Net\_PyramidID, and GA-Net\_Ori on satellite data (baseline model: SGM).

	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
GA-Net_PyramidED	83.76%	90.70%	93.00%	<b>7144M</b>	23.77h	<b>5623M</b>	<b>31.53s</b>
GA-Net_PyramidED+SPN	82.99%	91.05%	93.34%	7250M	24.56h	<b>5623M</b>	35.93s
GA-Net_PyramidID	81.45%	89.58%	92.40%	7558M	<b>19.11h</b>	6979M	33.11s
GA-Net_PyramidID+SPN	80.66%	89.10%	92.00%	7700M	20.27h	6979M	32.87s
GA-Net_Ori	<b>83.84%</b>	<b>91.42%</b>	<b>93.74%</b>	30514M	179.19h	15685M	401.91s
SGM	79.98%	82.74%	83.32%				

stereo pair. Regarding the evaluation, we follow our previous processing chain, using an aerial dataset with good stereo geometry to the same area to generate reference data. The test region, the reference disparity map and our stereo results are displayed in Figure 4.24.

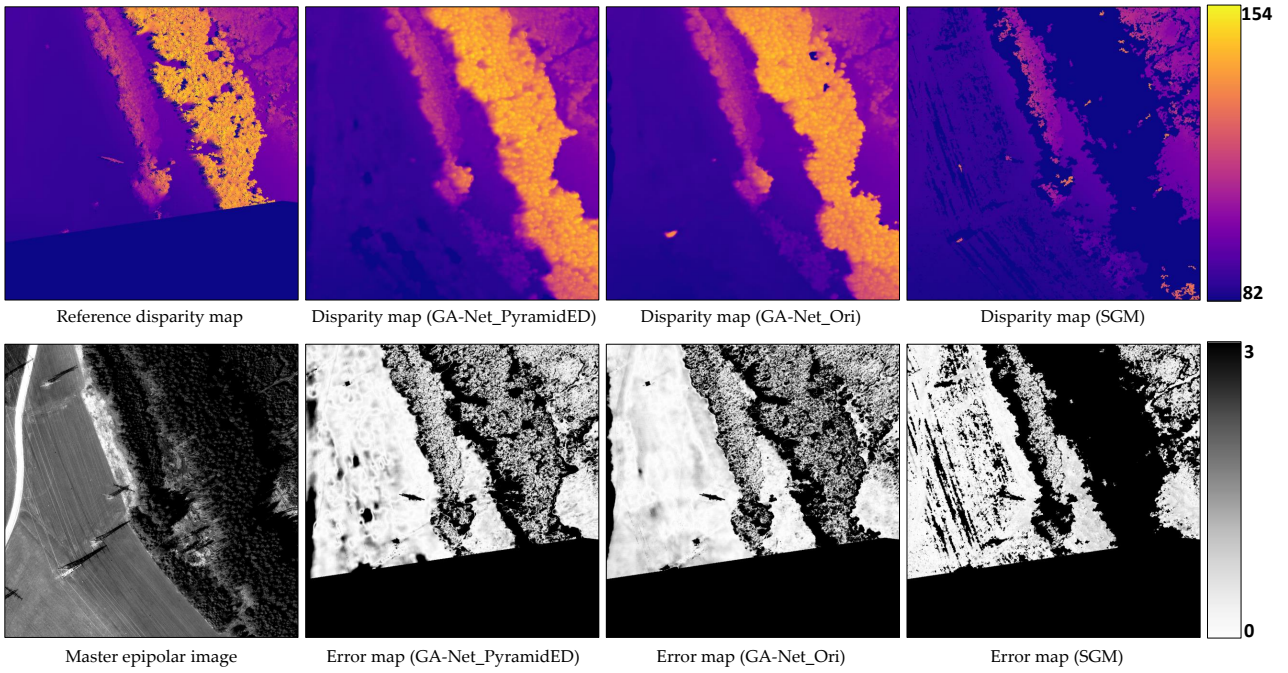
The mountain peak is located at the center of the image with a disparity of around 1250 pixels. Thus, we set the disparity range as  $[0, 1248]$ . Note that the model we use receives no supervision and knowledge, regarding the mountain area with that large disparity difference. However, we achieve a 3-pixel accuracy as 87.34%. There are temporal inconsistency between the satellite and reference data, leading to varying snow cover. Therefore, we use 3-pixel as the threshold. The visual comparison shows very similar results between our disparity prediction and the reference, considering the reconstruction density, smoothness, etc. Disparity holes are found from certain regions in our results. According to the image content, the regions are in shadow with limited texture information, where the network suffers from collecting enough information to locate the correspondences.

In the test period, the patch in size of  $768 \times 6912$  is fed to the network for disparity prediction. Considering the disparity range  $[0, 1248]$ , GA-Net\_Ori will theoretically need more than 200 GB GPU memory to process the same data. While in our pyramid implementation, GA-Net\_PyramidED consumes only around 20 GB.

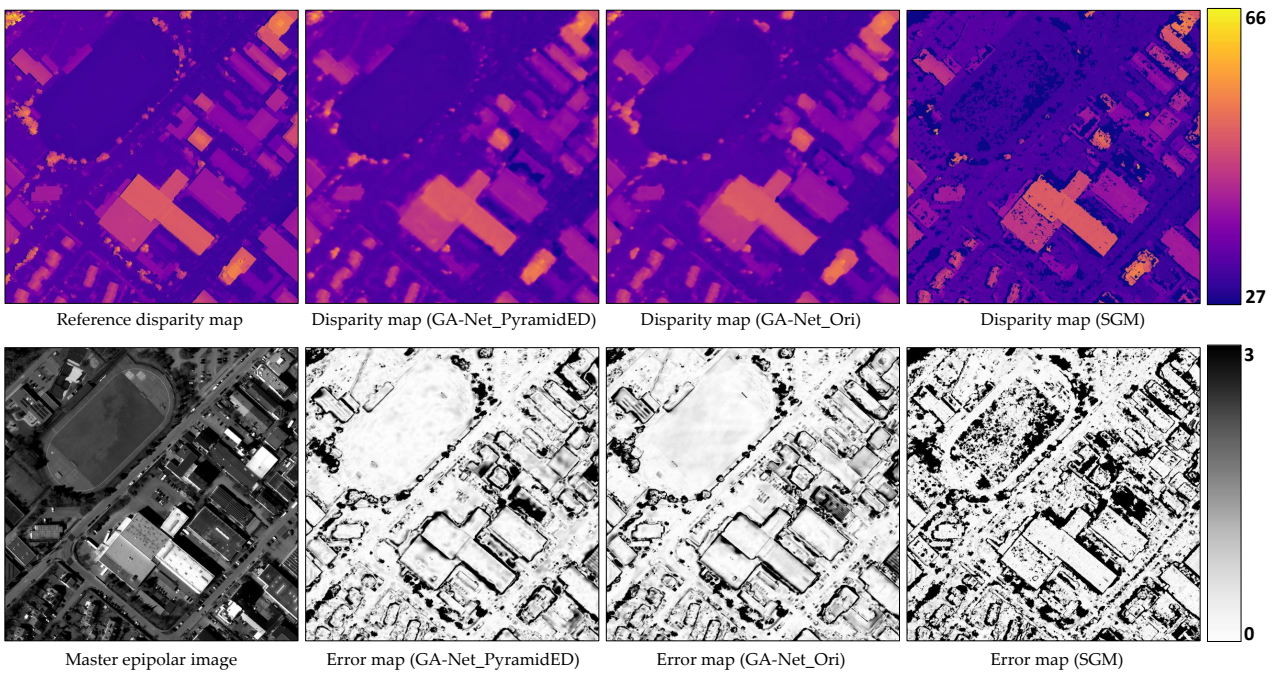
### 4.3.5 Conclusion and Outlook

Based on a pyramid architecture, our GA-Net\_Pyramid is able to roughly estimate the depth from downsampled feature, and then refine the prediction level by level until the original resolution is recovered. Thus, the efficiency is significantly enhanced with the accuracy maintained to be comparable with GA-Net\_Ori. Some technical details are found.

We firstly propose GA-Net\_PyramidED, which applies the GA-Net\_Ori model hierarchically. In our experiments on airborne and satellite data, it is demonstrated that GA-Net\_PyramidED is able to achieve similar results as GA-Net\_Ori, nevertheless, consuming much less GPU memory and runtime for both training and prediction. Considering that only the pyramid top exploits the absolute disparity range in low resolution to locate the stereo correspondence, GA-Net\_PyramidED is capable of processing stereo pairs with wider baselines if the same GPU memory for GA-Net\_Ori is available. This is particularly suitable to process large stereo pairs with high disparity search ranges in the field of remote sensing, which usually triggers the bottleneck of most memory-hungry deep neural networks. On the other hand, the aerial/satellite images mainly focus on large scale landscapes such as city areas, for which the local object heights/depths are generally smoother and regular with fewer occlusions, depth discontinuities, fine structures, etc., compared with the close-range datasets. Thus, the results from the previous level can better guide the disparity estimation on current level. More importantly, when large height variance exists within the scene, e.g.



(a) A vegetation area



(b) A building area

Fig. 4.23. Visual comparison among GA-Net\_PyramidED, GA-Net\_Ori and SGM on satellite data. In each subfigure, the reference disparity map and each algorithm's stereo results are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps.

in mountain areas, a rough depth prediction from lower resolution pyramid level is effective to limit the search range and avoid influence from ambiguous disparity candidates for higher resolution level.

Another architecture is designed as GA-Net\_PyramidID, which implicitly downsamples the input stereo pair via a U-Net feature extractor to feed each pyramid level using the intermediate feature map of its decoder. Concerning the close-range datasets, especially for Scene Flow that contains very complex and non-logical scene structures, both GA-Net\_PyramidED

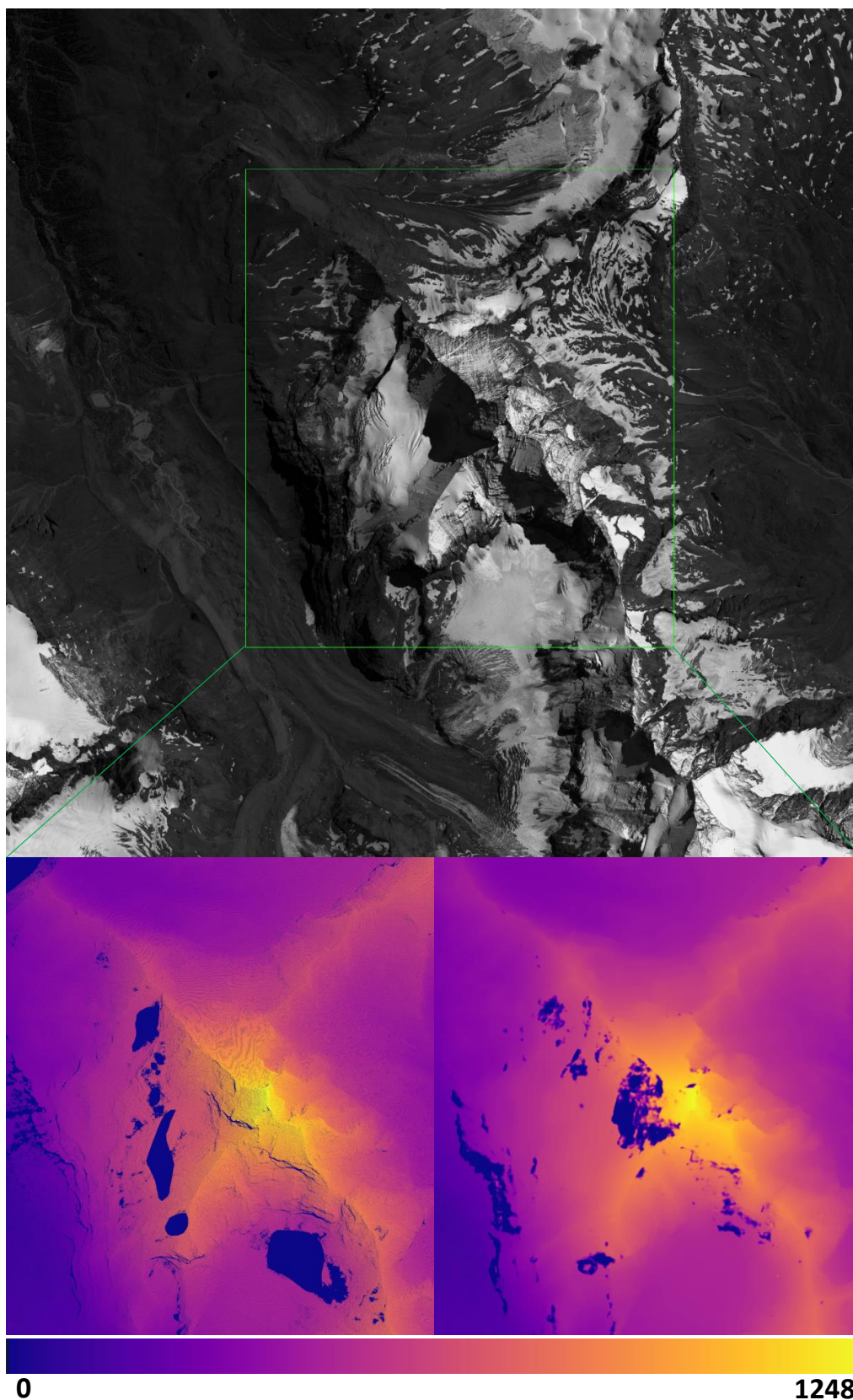


Fig. 4.24. A showcase to indicate the ability of our pyramid network in processing remote sensing stereo pair with large baseline. The reconstruction results for the region with a size of  $19791 \times 15639$  pixels highlighted by the green rectangle are shown below, from the reference disparity map (left) and our pyramid model (right). Test region: Matterhorn mountain, Switzerland. Test model: GA-Net\_PyramidED.

and GA-Net\_PyramidID are not competitive with GA-Net\_Ori. The accuracy could be influenced, when details are possibly omitted by the low resolution level. Besides, the resid-

ual search range may not support refinement for regions with rapid depth changes and discontinuities. With a SPN module added at the end to sharpen the boundaries, GA-Net\_PyramidID+SPN achieves the best result among all the pyramid models. While GA-Net\_Ori outperforms the proposed pyramid approaches on both close range datasets, Scene Flow and KITTI, the performance difference is smaller for the real-world KITTI 2012 data. Thus, we could be confident to apply this pyramidal revision in practical scenarios when the scene is natural and regular, to efficiently predict a disparity map with good quality.

SPN is applied on image segmentation to refine the object boundaries. In our experiments on close-range data, better depth estimation is achieved by our pyramid networks with SPN added, especially for GA-Net\_PyramidID. However, it is found that negative influence from SPN occurs on airborne and satellite data, for both GA-Net\_PyramidED and GA-Net\_PyramidID. The reason is that the resolution of aerial/satellite data is relatively low, with fewer details and depth discontinuities included, thus the strength of SPN is not embodied. More importantly, the training of SPN cannot be well supervised, considering that the number of valid training patches from airborne (987 millions) and satellite (934 millions) datasets is far less than the close-range datasets (18 billions). The condition to collect reference data is not as ideal as close-range scenarios using precise LiDAR scanning, structured light or synthetic labelling. In addition, SPN essentially refers to the input to improve the output, which are the master epipolar image and the disparity result in our case, respectively. The natural land texture and shadows, which are not necessarily related to ground height variation, may confuse SPN to locate the correct depth borders. The slightly changing and rolling ground height, e.g. in natural regions, could confuse the disparity post-processing as indicated by the lower 1-pixel accuracy.

Nowadays, the rapid development of deep learning and CNNs has made the technique dominate in the field of dense matching, leading to a sequence of high-rank algorithms in different close-range benchmarks. Compared to conventional approaches, the depth estimation for ill-posed areas, e.g. textureless regions, occlusions, etc, is better accomplished resulting in a considerable improvement. However, a large amount of well-annotated data and a time-consuming training are usually required, before a network reaches high performance. In the field of remote sensing, a huge amount of high definition data is supplied by unmanned aerial vehicles, helicopters, airplanes or satellites in all time. The data covers large areas with varying stereo baselines and image sizes of up to multiple gigapixels. Hence, a well-performed deep network from the field of computer vision would struggle to process the corresponding data, under the certain time and memory budget. Since that stereo datasets with reliable ground truth at sub-pixel disparity precision are not available in remote sensing, we build a dataset consisting of simultaneously acquired and carefully co-registered 30 cm satellite and 6 cm aerial imagery. The experimental results demonstrate that our proposed model can largely enhance the efficiency in training and test, while maintaining a comparable accuracy. The test on a satellite stereo pair over Matterhorn specifically highlights the significance of our method for processing large baseline stereo data.

In future research, more reference data should be collected for urban, rural and mountainous scenarios for remote sensing, in order to better supervise a learning based model in stereo prediction. Thus, we can better handle the ill-posed regions in shadows, depth boundaries, etc., and obtain high-quality geographical measurements for earth observation.

## 5 Conclusion and Outlook

### 5.1 Conclusion

Modern learning based techniques strengthen the model's ability for better understanding and expressing the scenes and data, from feature representation, analysis, to interpretation, thus, largely enhancing the corresponding algorithm's robustness to handle complex vision and semantics related problems. Regarding stereo dense matching and 3D reconstruction, tons of methods have been proposed in the field of computer vision targeting at the applications of self-driving, object recognition, etc. High performance is achieved under the support of advanced machine/deep learning techniques and rich training data sources. The methods greatly inspire the study in remote sensing for DSM generation, change detection, and so forth. However, a simple model transfer from computer vision to remote sensing does not automatically lead to competitive performance. For example, the data collection is usually more challenging, due to the varying environmental factors such as the illumination changes and clouds. More importantly, the target are mostly dynamic scenes which makes it difficult to obtain a temporarily consistent reference dataset for model supervision and evaluation. On the other hand, the data volume of remote sensing data is much higher due to the large format airborne and satellite sensors. The corresponding tasks have a higher demand on the algorithms' efficiency, especially for processing wide-baseline stereo data. Hence, the SOTA algorithms in computer vision need be adjusted before they can be applied to robustly to remote sensing data.

SGM achieves steady performance in stereo matching with reasonable computational resources, thus it is widely applied in the field of remote sensing. Therefore, the thesis aims at improving the algorithm, referring to the SOTA strategies from computer vision, with special adjustment for a more appropriate stereo processing in remote sensing. We follow the pipeline of SGM, meanwhile adapting every single module for better performance. The following conclusions are drawn with our specific contributions summarized as:

- ◇ The calculation of matching costs as the first step of dense matching determines the similarity of potentially matching pixels and forms the basis for further determination of correspondences and depth estimation. Therefore, it is promising to use machine learning and convolutional neural networks to deeply represent the individual pixels (and their surrounding neighborhood) for appropriate comparison. MC-CNN offers a light-weight network for pixel similarity measurement, however, also demands a certain amount of annotated samples to train the model which may be difficult to collect in remote sensing. With plant stereo reconstruction and drought detection as a case study, two strategies are proposed to train a MC-CNN model.

The first scheme relies on an active depth sensor to generate reference data. We use a LiDAR scanner to obtain a point cloud of the target plant and acquire disparity maps via projecting the 3D object points onto each master frame. Normally, a co-registration between the LiDAR point cloud and each stereo pair is sufficient to connect the two data sources within a common coordinate system before training. However, branches and leaves move and deform due to airflow during laser scanning, disrupting co-registration. We therefore, propose to roughly align the LiDAR and stereo point clouds acquired based on a classic stereo matching method, and then manually select the well-aligned points for further fine co-registration using GICP. At last, only well-registered points are used as ground truth. Based on this strategy, a limited amount of ground truth disparity maps could be obtained, and used to finetune a pre-trained MC-CNN model.

The second scheme uses self-supervision during the training of MC-CNN, thus avoiding the time-consuming reference data annotation. Based on SGM and a pre-trained MC-

CNN model, a disparity map is firstly generated for each of the stereo images, on which a rigid left-right consistency check is applied to exclude most outliers. Afterwards, only the disparity values that pass the consistency check are used to further fine-tune the pre-trained MC-CNN model.

In remote sensing, the self-supervised strategy is essentially more promising, considering that in many cases a reference dataset is difficult to collect or only provides low-quality training samples. The experiment results prove that the self-trained MC-CNN is capable of achieving competitive accuracy with the model trained on ground truth generated from LiDAR. When tested on our research project ForDroughtDet, the combination of SGM and a self-supervised MC-CNN model can create a detailed 3D model of wild trees in centimeter level. The deformation of leaves is clearly visible and could be used for drought stress detection.

- ◇ After the matching cost calculation, SGM applies multiple 1D SO in canonical directions and sums up the corresponding energy to compute the disparity based on the WTA strategy. Thus, the correspondence between the stereo pair is determined with both pixel similarity and 2D spatial smoothness of depth considered. SGM-Forest trains a random forest to adaptively select the best scanline for further processing, to avoid exploiting scanlines with wrong or inconsistent estimation. However regarding the scanline selection, we experimentally demonstrate that for most pixels, multiple scanlines provide a good disparity prediction. Hence, we adjust the training of the random forest, by setting the target as selecting all possible well-behaved scanlines.

Experiments on cross-domain stereo datasets, including close-range, aerial and satellite data, prove that the multiple scanlines selection further improves the SGM performance. The strategy consistently improves the dense matching results, especially on ill-posed areas, such as textureless and reflective regions with less noise.

- ◇ We finally design an end-to-end convolutional neural network based on the SOTA GA-Net to simulate SGM and estimate the disparity using a coarse to fine approach. A disparity map is directly predicted for an input stereo pair, through the network which includes feature extraction, matching cost computation and regularization, and disparity regression. The network is constructed within a pyramid architecture. On the pyramid top, a disparity map is estimated for a downsampled stereo input, for which the calculation is highly simplified considering the decreased image height, width and the disparity range. The disparity map can be upsampled to the next pyramid level, as an initial estimation for higher resolution processing. Therefore, only a refinement within a small buffer around the initial estimation is needed, instead of locating the best disparity value through an absolute range. Along with the pyramidal processing towards the pyramid bottom, only a small overhead is added at each level to refine the disparity estimation, until the original resolution is recovered. Thus, the efficiency is largely enhanced.

Two variants of the feature extractor are proposed, the first one simply feeds a downsampled stereo pair to each pyramid level and is designated as explicit downsampling. The other one applies a U-Net to extract feature from the stereo pair in original resolution, and takes the intermediate feature maps from the decoder to feed each pyramid level according to the expected resolution. Thus, an implicit downsampling is realized. Our pyramid network using both feature extractors are tested on close-range, airborne and satellite stereo datasets. Regarding the close-range experiments, our network is much faster than the baseline method GA-Net consuming lower GPU memory. However, the quality of the estimated disparity maps for close range imagery is not competitive with GA-Net, with blurred boundaries and some loss of smaller details. The two feature extractors achieve similar accuracy. As for the experiments on aerial and satellite data, our network is still much more efficient than GA-Net and acquire comparable accuracy (The



feature extractor with explicit downsampling slightly surpasses the other). We demonstrate that, our pyramid network is appropriate for reconstruction of overhead remote sensing datasets. In the field of remote sensing, aerial and satellite stereo processing is more frequent than complex close-range scenarios, underscoring the significance of our method.

We implement a series of modifications on SGM, from technical adjustment on specific modules to an overall CNN based approximation with a pyramid structure to improve the efficiency of disparity estimation. The proposed methods are tested on cross-domain stereo datasets, and proven to outperform the baseline methods and achieve SOTA performance. The thesis enriches the approaches for dense matching optimization, especially in the field of remote sensing for change detection, building reconstruction, DSM generation, etc., however, also shows its limits requiring further improvement. For example, the algorithms evaluation still needs reference samples to test the performance before releasing to practical use. A self-training strategy may simplify the model finetuning, nevertheless, still needs a pre-trained model from closely related domain and cannot provide competitive results with a network trained using high-quality ground truth. Besides, the full cost volumes from every 1D scanline is necessary to train the random forest for scanline selection in SGM, which results in extra memory consumption. High efficiency is achieved by our pyramid network, however, losing details cannot be avoided from the downsampling-upsampling processing logic. The stereo estimation for high resolution close-range data is not satisfying. Hence, extra work is needed in future research.

## 5.2 Outlook

The state-of-the-art learning based techniques have broken the rigid processing chain of traditional stereo matching, via guiding a model to analyze the problem and adaptively handle each sub-task according to the experience from the training. However, a higher demand is placed on the methods for higher accuracy, less responding time, and more intelligent decision-making, considering the more sophisticated functions required in modern scenarios. In the field of remote sensing, 3D knowledge is the most intuitive information for consistent and precise earth observation, thus naturally relying on more advanced stereo techniques to reconstruct a digital geometry of the world.

In order to better supervise a model for remote sensing stereo tasks, a synthetic airborne and satellite dataset is promising, since that the real reference data collection and annotation are usually complicated especially in extreme regions. Different rendering software are available to attach color and texture information on virtual city models with known geometric shape, position and orientation (Denninger et al., 2019). Then with a manual set of camera height, images and height maps can be simulated according to the cities' appearance and geometry, meanwhile obtaining the expected ground sampling distance. Thus, a deep neural network is well supervised for aerial and satellite stereo applications. With diverse city models to render the datasets, a robust model can be acquired for different scenarios.

The current pyramid networks for stereo matching (Wang et al., 2019; Yang et al., 2019) naturally suffer from losing details and blurring the depth edges, due to that the disparity estimation highly relies on the initial prediction using images with the coarsest resolution. The message delivered between the two consecutive pyramid levels is, therefore, very important to keep accurate results and remove the outliers for further refinement. In addition to a regular buffer set around the initial disparity estimation, statistical rules can be applied to sample the most possible disparity candidates in the following test to determine a better disparity value. For example, a probability density function (PDF) is a good option to express the possibility of each disparity candidate to be correct in the previous pyramid level.

Then a sequence of disparity candidates could be obtained with the cumulative probabilities equally sampled in range of  $[0, 1]$ , such that a certain number of candidates are available within each peak of the PDF. Thus, a bad disparity estimation from previous levels could be corrected in current level, for example at object boundaries with large depth differences.

Recent researches focus more on semantics learning for more intelligent analysis as humans to solve concrete vision tasks. The object geometry can be unconsciously perceived using semantic clues, which leads to brain technology development (Vinny and Singh, 2020). Besides, attention (Carion et al., 2020) can also assist the network to quickly concentrate on the more useful information for a fast depth sensing.

## References

- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (11): 2274–2282.
- Anandan, P., 1989. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision* 2 (3): 283–310.
- Arndt, O. J., Becker, D., Banz, C., Blume, H., 2013. Parallel implementation of real-time semi-global matching on embedded multi-core architectures. In: 2013 International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS), 56–63.
- Baker, S., Szeliski, R., Anandan, P., 1998. A layered approach to stereo reconstruction. In: Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR, Cat. No.98CB36231), 434–441.
- Banz, C., Hesselbarth, S., Flatt, H., Blume, H., Pirsch, P., 2010. Real-time stereo vision system using semi-global matching disparity estimation: Architecture and fpga-implementation. In: 2010 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, 93–101.
- Banz, C., Pirsch, P., Blume, H., 2012. Evaluation of penalty functions for semi-global matching cost aggregation. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences [XXII ISPRS Congress, Technical Commission I] 39 (2012), Nr. B3, Vol. 39, Göttingen: Copernicus GmbH, 1–6.
- Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D. B., 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 28 (3).
- Batsos, K., Cai, C., Mordohai, P., 2018. CBMV: A coalesced bidirectional matching volume for disparity estimation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2060–2069.
- Batsos, K., Mordohai, P., 2018. RecResNet: A recurrent residual CNN architecture for disparity map enhancement. In: 2018 International Conference on 3D Vision (3DV), 238–247.
- Beyer, H., 2007. Evolution strategies. *Scholarpedia* 2 (8): 1965, revision #193589.
- Biernacki, C., Celeux, G., Govaert, G., 2000. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (7): 719–725.
- Bleyer, M., Breiteneder, C., 2013. Stereo matching—state-of-the-art and research challenges. Springer London, London, 143–179.
- Bolles, R. C., 1993. The JISCT stereo evaluation. In: Proceedings of Image Understanding Workshop, 263–274.
- Bolles, R. C., Baker, H. H., Marimont, D. H., 1987. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision* 1 (1): 7–55.
- Bosch, M., Foster, K., Christie, G., Wang, S., Hager, G. D., Brown, M., 2019. Semantic stereo for incidental satellite images. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), 1524–1532.
- Boutell, M. R., Luo, J., Shen, X., Brown, C. M., 2004. Learning multi-label scene classification. *Pattern Recognition* 37 (9): 1757 – 1771.
- Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23 (11): 1222–1239.
- Breiman, L., 2001. Random forests. *Machine Learning* 45 (1): 5–32.
- Bromley, J., Bentz, J., Bottou, L., Guyon, I., LeCun, Y., Moore, C., Sackinger, E., Shah, R., 1993. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence* 7.
- Brown, M. Z., Goldberg, H., Foster, K., Leichtman, A., Wang, S., Hagstrom, S., Bosch, M., Almes, S., 2018. Large-scale public lidar and satellite image data set for urban semantic labeling. In: Defense

- + Security.
- Cai, C., Poggi, M., Mattoccia, S., Mordohai, P., 2020. Matching-space stereo networks for cross-domain generalization. arXiv preprint arXiv:2010.07347 .
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers.
- Chabra, R., Straub, J., Sweeney, C., Newcombe, R., Fuchs, H., 2019. StereoDRNet: Dilated residual stereonet. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 11778–11787.
- Chakravarty, P., Narayanan, P., Roussel, T., 2019. Gen-slam: Generative modeling for monocular simultaneous localization and mapping. In: 2019 International Conference on Robotics and Automation (ICRA), 147–153.
- Chang, J., Chen, Y., 2018. Pyramid stereo matching network. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5410–5418.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (4): 834–848.
- Chen, Z., Sun, X., Wang, L., Yu, Y., Huang, C., 2015. A deep visual correspondence embedding model for stereo matching costs. In: 2015 IEEE International Conference on Computer Vision (ICCV), 972–980.
- Cheng, J., Tsai, Y., Wang, S., Yang, M., 2017. SegFlow: Joint learning for video object segmentation and optical flow. In: 2017 IEEE International Conference on Computer Vision (ICCV), 686–695.
- Cheng, X., Wang, P., Yang, R., 2020. Learning depth with convolutional spatial propagation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (10): 2361–2379.
- Clare, A., King, R. D., 2001. Knowledge discovery in multi-label phenotype data. In: De Raedt, L., Siebes, A. (Hrsg.), *Principles of Data Mining and Knowledge Discovery*, Springer Berlin Heidelberg, Berlin, Heidelberg, 42–53.
- Cochran, S., Medioni, G., 1992. 3-D surface description from binocular stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (10): 981–994.
- Collins, R., 1996. A space-sweep approach to true multi-image matching. In: *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 358–363.
- d’Angelo, P., 2016. Improving semi-global matching: Cost aggregation and confidence measure. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLI-B1*: 299–304.
- d’Angelo, P., Reinartz, P., 2012. Semiglobal matching results on the ISPRS stereo matching benchmark. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 3819*: 79–84.
- Denninger, M., Sundermeyer, M., Winkelbauer, D., Zidan, Y., Olefir, D., Elbadrawy, M., Lodhi, A., Katam, H., 2019. Blenderproc. arXiv preprint arXiv:1911.01911 .
- Dosovitskiy, A., Fischer, P., Ilg, E., Häusser, P., Hazirbas, C., Golkov, V., v. d. Smagt, P., Cremers, D., Brox, T., 2015. FlowNet: Learning optical flow with convolutional networks. In: 2015 IEEE International Conference on Computer Vision (ICCV), 2758–2766.
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., 2017. CARLA: An open urban driving simulator. In: Levine, S., Vanhoucke, V., Goldberg, K. (Hrsg.), *Proceedings of the 1st Annual Conference on Robot Learning*, Vol. 78 of *Proceedings of Machine Learning Research (PMLR)*, 1–16.
- Duda, R. O., Hart, P. E., Stork, D. G., 2001. *Pattern Classification*, 2nd Edition. Wiley, New York.
- Duggal, S., Wang, S., Ma, W., Hu, R., Urtasun, R., 2019. DeepPruner: Learning efficient stereo matching via differentiable patchmatch. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 4383–4392.
- El-Khamy, M., Du, X., Ren, H., Lee, J., 2019. Multi-task learning of depth from tele and wide stereo image pairs. In: 2019 IEEE International Conference on Image Processing (ICIP), 4300–4304.
- Elmoataz, A., Lezoray, O., Boughleux, S., 2008. Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing. *IEEE Transactions on Image Processing* 17 (7):

- 1047–1060.
- Facciolo, G., Franchis, C. d., Meinhardt, E., 2015. MGM: A significantly more global matching for stereovision. In: Press, B. (Hrsg.), Proceedings of the British Machine Vision Conference (BMVC 2015), Swansea, United Kingdom.
- Facil, J. M., Ummenhofer, B., Zhou, H., Montesano, L., Brox, T., Civera, J., 2019. Cam-convs: Camera-aware multi-scale convolutions for single-view depth. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 11818–11827.
- Falkner, S., Klein, A., Hutter, F., 2018. BOHB: Robust and efficient hyperparameter optimization at scale. In: International Conference on Machine Learning (ICML), PMLR, 1437–1446.
- Felzenszwalb, P., Huttenlocher, D., 2004. Efficient belief propagation for early vision. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, I–I.
- Finn, C., Abbeel, P., Levine, S., 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning (ICML), PMLR, 1126–1135.
- Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D., 2018. Deep ordinal regression network for monocular depth estimation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2002–2011.
- Fu, Z., Fard, M. A., 2018. Learning confidence measures by multi-modal convolutional neural networks. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 1321–1330.
- Fua, P., 1993. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications* 6: 35–49.
- Fusiello, A., Roberto, V., Trucco, E., 1997. Efficient stereo with multiple windowing. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 858–863.
- Garg, R., B.G., V. K., Carneiro, G., Reid, I., 2016. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Hrsg.), *Computer Vision – ECCV 2016*, Springer International Publishing, Cham, 740–756.
- Gehrig, S. K., Eberli, F., Meyer, T., 2009. A real-time low-power stereo vision engine using semi-global matching. In: Fritz, M., Schiele, B., Piater, J. H. (Hrsg.), *Computer Vision Systems*, Springer Berlin Heidelberg, Berlin, Heidelberg, 134–143.
- Gehrig, S. K., Rabe, C., 2010. Real-time semi-global matching on the cpu. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops (CVPRW), 85–92.
- Gehrke, S., Morin, K., Downey, M., Boehrer, N., Fuchs, T., 2010. Semi-global matching: An alternative to LIDAR for DSM generation. In: Proceedings of the 2010 Canadian Geomatics Conference and Symposium of Commission I, Vol. 2.
- Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR).
- Geman, S., Geman, D., 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6* (6): 721–741.
- Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., Dorn, S., et al., 2020. A2D2: Audi autonomous driving dataset. arXiv preprint arXiv:2004.06320 .
- Gidaris, S., Komodakis, N., 2017. Detect, replace, refine: Deep structured prediction for pixel wise labeling. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 7187–7196.
- Girardeau-Montaut, D., Roux, M., Marc, R., Thibault, G., 2005. Change detection on points cloud data acquired with a ground laser scanner. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36 (part 3): W19.
- Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. *Deep Learning*. Vol. 1. MIT press Cambridge.
- Görner, M., Chilian, A., Hirschmüller, H., 2010. Towards an autonomous walking robot for planetary surfaces. In: *i-SAIRAS 2010*, Sapporo, Japan.

- Guo, X., Yang, K., Yang, W., Wang, X., Li, H., 2019. Group-wise correlation stereo network. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 3268–3277.
- Haala, N., 2014. Dense image matching final report. EuroSDR Publication Series, Official Publication 64: 115–145.
- Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A. C., 2015. Matchnet: Unifying feature and metric learning for patch-based matching. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3279–3286.
- Hannah, M. J., 1974. Computer Matching of Areas in Stereo Images. Stanford University.
- Hartley, R., Zisserman, A., 2004. Multiple View Geometry in Computer Vision, 2nd Edition. Cambridge University Press.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778.
- Hengstler, S., Prashanth, D., Fong, S., Aghajan, H., 2007. MeshEye: A hybrid-resolution smart camera mote for applications in distributed intelligent surveillance. In: 2007 6th International Symposium on Information Processing in Sensor Networks, 360–369.
- Hirschmüller, H., 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, 807–814 vol. 2.
- Hirschmüller, H., 2008. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2): 328–341.
- Hirschmüller, H., 2011. Semi-global matching-motivation, developments and applications. *Photogrammetric Week 11* : 173–184.
- Hirschmüller, H., Innocent, P. R., Garibaldi, J., 2002. Real-time correlation-based stereo vision with reduced border errors. *International Journal of Computer Vision* 47 (1): 229–246.
- Hirschmüller, H., Scharstein, D., 2007. Evaluation of cost functions for stereo matching. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1–8.
- Hoiem, D., Efros, A. A., Hebert, M., 2005. Automatic photo pop-up. In: ACM SIGGRAPH 2005 Papers, SIGGRAPH '05, Association for Computing Machinery, New York, NY, USA, 577–584.
- Holschneider, M., Kronland-Martinet, R., Morlet, J., Tchamitchian, P., 1990. A real-time algorithm for signal analysis with the help of the wavelet transform. In: Combes, J.-M., Grossmann, A., Tchamitchian, P. (Hrsg.), *Wavelets*, Springer Berlin Heidelberg, Berlin, Heidelberg, 286–297.
- Hosni, A., Bleyer, M., Gelautz, M., Rhemann, C., 2009. Local stereo matching using geodesic support weights. In: 2009 16th IEEE International Conference on Image Processing (ICIP), 2093–2096.
- Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E., 2020. Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (8): 2011–2023.
- Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q., 2017. Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2261–2269.
- Huang, X., Wang, P., Cheng, X., Zhou, D., Geng, Q., Yang, R., 2020. The ApolloScape open dataset for autonomous driving and its application. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (10): 2702–2719.
- Hutter, F., Kotthoff, L., Vanschoren, J., 2019. Automated Machine Learning: Methods, Systems, Challenges. Springer Nature.
- Ilg, E., Saikia, T., Keuper, M., Brox, T., 2018. Occlusions, motion and depth boundaries with a generic network for disparity, optical flow or scene flow estimation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Hrsg.), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 626–643.
- Intille, S. S., Bobick, A. F., 1994. Disparity-space images and large occlusion stereo. In: Eklundh, J.-O. (Hrsg.), *Computer Vision – ECCV '94*, Springer Berlin Heidelberg, Berlin, Heidelberg, 179–186.
- Jiang, H., Sun, D., Jampani, V., Lv, Z., Learned-Miller, E., Kautz, J., 2019. SENSE: A shared encoder network for scene-flow estimation. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 3194–3203.

- Jie, Z., Wang, P., Ling, Y., Zhao, B., Wei, Y., Feng, J., Liu, W., 2018. Left-right comparative recurrent model for stereo matching. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 3838–3846.
- Joung, S., Kim, S., Ham, B., Sohn, K., 2017. Unsupervised stereo matching using correspondence consistency. In: 2017 IEEE International Conference on Image Processing (ICIP), 2518–2522.
- Kanade, T., Kano, H., Kimura, S., Yoshida, A., Oda, K., 1995. Development of a video-rate stereo machine. In: Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots, Vol. 3, IEEE, 95–100.
- Kelly, R., McConnell, P., Mildenerger, S., 1977. The gestalt photomapping system. *Photogrammetric Engineering and Remote Sensing* 43: 1407–1417.
- Kendall, A., Gal, Y., 2017. What uncertainties do we need in bayesian deep learning for computer vision? In: Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Hrsg.), *Advances in Neural Information Processing Systems*, Vol. 30, Curran Associates, Inc.
- Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A., 2017. End-to-end learning of geometry and context for deep stereo regression. In: 2017 IEEE International Conference on Computer Vision (ICCV), 66–75.
- Khamis, S., Fanello, S., Rhemann, C., Kowdle, A., Valentin, J., Izadi, S., 2018. StereoNet: Guided hierarchical refinement for real-time edge-aware depth prediction. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Hrsg.), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 596–613.
- Kim, S., Min, D., Kim, S., Sohn, K., 2017. Feature augmentation for learning confidence measure in stereo matching. *IEEE Transactions on Image Processing* 26 (12): 6019–6033.
- Kim, S., Min, D., Kim, S., Sohn, K., 2019. Unified confidence estimation networks for robust stereo matching. *IEEE Transactions on Image Processing* 28 (3): 1299–1313.
- Kingma, D. P., Ba, J., 2017. Adam: A method for stochastic optimization.
- Knöbelreiter, P., Pock, T., 2019. Learned collaborative stereo refinement. In: German Conference on Pattern Recognition (GCPR), Springer, 3–17.
- Knöbelreiter, P., Reinbacher, C., Shekhovtsov, A., Pock, T., 2017. End-to-end training of hybrid cnn-crf models for stereo. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1456–1465.
- Knöbelreiter, P., Vogel, C., Pock, T., 2018. Self-supervised learning for stereo reconstruction on aerial images. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, 4379–4382.
- Kolmogorov, V., Zabih, R., 2002. Multi-camera scene reconstruction via graph cuts. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (Hrsg.), *Computer Vision — ECCV 2002*, Springer Berlin Heidelberg, Berlin, Heidelberg, 82–96.
- Krizhevsky, A., Sutskever, I., Hinton, G. E., 2012. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, 1097–1105.
- Kurz, F., Rosenbaum, D., Meynberg, O., Mattyus, G., Reinartz, P., 2014. Performance of a real-time sensor and processing system on a helicopter. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XL-1*: 189–193.
- Kusch, G., d’Angelo, P., Qin, R., Poli, D., Reinartz, P., Cremers, D., 2014. Dsm accuracy evaluation for the ISPRS Commission I image matching benchmark. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 40: 195–200.
- Kutulakos, K., Seitz, S., 1999. A theory of shape by space carving. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV)*, Vol. 1, 307–314 vol.1.
- Lafferty, J. D., McCallum, A., Pereira, F. C. N., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, ICML ’01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 282–289.
- Laga, H., Jospin, L. V., Boussaid, F., Bennamoun, M., 2020. A survey on deep learning techniques for stereo-based depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

- Lawrence, S., Giles, C., Tsoi, A. C., Back, A., 1997. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks* 8 (1): 98–113.
- Le Saux, B., Yokoya, N., Hansch, R., Brown, M., Hager, G., 2019. 2019 data fusion contest [technical committees]. *IEEE Geoscience and Remote Sensing Magazine* 7 (1): 103–105.
- Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86 (11): 2278–2324.
- Lee, D. C., Hebert, M., Kanade, T., 2009. Geometric reasoning for single image structure recovery. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2136–2143.
- Li, Z., Snavely, N., 2018. MegaDepth: Learning single-view depth prediction from internet photos. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2041–2050.
- Liang, Z., Feng, Y., Guo, Y., Liu, H., Chen, W., Qiao, L., Zhou, L., Zhang, J., 2018. Learning for disparity estimation through feature constancy. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2811–2820.
- Liu, H., Simonyan, K., Yang, Y., 2018. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*.
- Liu, S., De Mello, S., Gu, J., Zhong, G., Yang, M.-H., Kautz, J., 2017. Learning affinity via spatial propagation networks. In: Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Hrsg.), *Advances in Neural Information Processing Systems*, Vol. 30, Curran Associates, Inc., 1520–1530.
- Liu, X., Wu, J., Zhou, Z., 2009. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39 (2): 539–550.
- Liu, Y., Cheng, M., Hu, X., Bian, J., Zhang, L., Bai, X., Tang, J., 2019. Richer convolutional features for edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41 (8): 1939–1946.
- Lu, Z., Wang, J., Li, Z., Chen, S., Wu, F., 2021. A resource-efficient pipelined architecture for real-time semi-global stereo matching. *IEEE Transactions on Circuits and Systems for Video Technology* : 1–1.
- Luo, W., Schwing, A. G., Urtasun, R., 2016. Efficient deep learning for stereo matching. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 5695–5703.
- Matthies, L., Kanade, T., Szeliski, R., 1989. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision* 3 (3): 209–238.
- Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T., 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 4040–4048.
- McCallum, A. K., 1999. Multi-label text classification with a mixture model trained by EM. In: *AAAI 99 workshop on text learning*, Citeseer.
- Mehrtretter, M., 2020. Uncertainty estimation for end-to-end learned dense stereo matching via probabilistic deep learning. *arXiv preprint arXiv:2002.03663*.
- Mei, X., Sun, X., Zhou, M., Jiao, S., Wang, H., Xiaopeng Zhang, 2011. On building an accurate stereo matching system on graphics hardware. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 467–474.
- Menze, M., Geiger, A., 2015. Object scene flow for autonomous vehicles. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3061–3070.
- Michael, M., Salmen, J., Stallkamp, J., Schlipsing, M., 2013. Real-time stereo vision: Optimizing semi-global matching. In: 2013 IEEE Intelligent Vehicles Symposium (IV), 1197–1202.
- Michel, J., Sarrazin, E., Youssefi, D., Cournet, M., Buffe, F., Delvit, J. M., Emilien, A., Bosman, J., Melet, O., L’Helguen, C., 2020. A new satellite imagery stereo pipeline designed for scalability, robustness and performance. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences V-2-2020*: 171–178.
- Milanfar, P., 2013. A tour of modern image filtering: New insights and methods, both practical and theoretical. *IEEE Signal Processing Magazine* 30 (1): 106–128.
- Min, D., Choi, S., Lu, J., Ham, B., Sohn, K., Do, M. N., 2014. Fast global image smoothing based on



- weighted least squares. *IEEE Transactions on Image Processing* 23 (12): 5638–5653.
- Mitchell, T. M., 1997. *Machine Learning*, 1st Edition. McGraw-Hill, Inc., USA.
- Murray, D., Little, J. J., 2000. Using real-time stereo vision for mobile robot navigation. *autonomous robots* 8 (2): 161–171.
- Ni, J., Li, Q., Liu, Y., Zhou, Y., 2018. Second-order semi-global stereo matching algorithm based on slanted plane iterative optimization. *IEEE Access* 6: 61735–61747.
- Nie, G., Cheng, M., Liu, Y., Liang, Z., Fan, D., Liu, Y., Wang, Y., 2019. Multi-level context ultra-aggregation for stereo matching. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 3278–3286.
- Okutomi, M., Kanade, T., 1991. A multiple-baseline stereo. In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 63–69.
- Okutomi, M., Kanade, T., 1993. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15 (4): 353–363.
- Pang, J., Sun, W., Ren, J. S., Yang, C., Yan, Q., 2017. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 878–886.
- Pang, J., Sun, W., Yang, C., Ren, J., Xiao, R., Zeng, J., Lin, L., 2018. Zoom and learn: Generalizing deep stereo matching to novel domains. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2070–2079.
- Park, H., Lee, K. M., 2017. Look wider to match image patches with convolutional neural networks. *IEEE Signal Processing Letters* 24 (12): 1788–1792.
- Park, M., Yoon, K., 2015. Leveraging stereo matching with learning-based confidence measures. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 101–109.
- Poggi, M., Mattocchia, S., 2016. Learning a general-purpose confidence measure based on  $O(1)$  features and a smarter aggregation strategy for semi global matching. In: 2016 Fourth International Conference on 3D Vision (3DV), 509–518.
- Poggi, M., Mattocchia, S., 2016. Learning from scratch a confidence measure. In: *BMVC*.
- Poggi, M., Tosi, F., Batsos, K., Mordohai, P., Mattocchia, S., 2020. On the synergies between machine learning and stereo: a survey.
- Qin, R., Huang, X., Gruen, A., Schmitt, G., 2015. Object-based 3-D building change detection on multitemporal stereo images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8 (5): 2125–2137.
- Rosu, A.-M., Pierrot-Deseilligny, M., Delorme, A., Binet, R., Klinger, Y., 2015. Measurement of ground displacement from optical satellite image correlation using the free open-source software micmac. *ISPRS Journal of Photogrammetry and Remote Sensing* 100: 48–59, high-Resolution Earth Imaging for Geospatial Information.
- Rothermel, M., 2017. Development of a SGM-based Multi-View Reconstruction Framework for Aerial Imagery. Ph.D. thesis, University of Stuttgart, Stuttgart.
- Ruder, S., 2017. An overview of multi-task learning in deep neural networks.
- Ryan, T. W., Gray, R. T., Hunt, B. R., 1980. Prediction of correlation errors in stereo-pair images. *Optical Engineering* 19 (3): 312.
- Saikia, T., Marrakchi, Y., Zela, A., Hutter, F., Brox, T., 2019. Autodispnet: Improving disparity estimation with automl. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 1812–1823.
- Savva, M., Chang, A. X., Hanrahan, P., 2015. Semantically-enriched 3d models for common-sense knowledge. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 24–31.
- Schapire, R. E., Singer, Y., 2000. BoosTexter: A boosting-based system for text categorization. *Machine Learning* 39 (2): 135–168.
- Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P., 2014. High-resolution stereo datasets with subpixel-accurate ground truth. In: *German Conference on Pattern Recognition (GCPR)*, Springer, 31–42.

- Scharstein, D., Pal, C., 2007. Learning conditional random fields for stereo. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1–8.
- Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision* 47 (1): 7–42.
- Scharstein, D., Szeliski, R., 2003. High-accuracy stereo depth maps using structured light. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2003. Proceedings., Vol. 1, I–I.
- Scharstein, D., Tanai, T., Sinha, S. N., 2017. Semi-global stereo matching with surface orientation priors. In: 2017 International Conference on 3D Vision (3DV), 215–224.
- Schmid, K., Tomic, T., Ruess, F., Hirschmüller, H., Suppa, M., 2013. Stereo vision based indoor/outdoor navigation for flying robots. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3955–3962.
- Schönberger, J. L., Sinha, S. N., Pollefeys, M., 2018. Learning to fuse proposals from multiple scan-line optimizations in semi-global matching. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Hrsg.), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 758–775.
- Schöps, T., Schönberger, J. L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A., 2017. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2538–2547.
- Schumacher, F., Greiner, T., 2014. Matching cost computation algorithm and high speed FPGA architecture for high quality real-time semi global matching stereo vision for road scenes. In: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), 3064–3069.
- Schuster, R., Wasenmüller, O., Unger, C., Stricker, D., 2019. SDC-stacked dilated convolution: A unified descriptor network for dense matching tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2556–2565.
- Segal, A., Haehnel, D., Thrun, S., 2009. Generalized-icp. In: *Robotics: science and systems*, Vol. 2, Seattle, WA, 435.
- Seitz, S., Curless, B., Diebel, J., Scharstein, D., Szeliski, R., 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, 519–528.
- Seitz, S., Dyer, C., 1997. Photorealistic scene reconstruction by voxel coloring. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 1067–1073.
- Seki, A., Pollefeys, M., 2016. Patch based confidence prediction for dense disparity map. In: *BMVC*, Vol. 2, 4.
- Seki, A., Pollefeys, M., 2017. Sgm-nets: Semi-global matching with neural networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6640–6649.
- Shaked, A., Wolf, L., 2017. Improved stereo matching with constant highway networks and reflective confidence learning. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6901–6910.
- Shean, D. E., Alexandrov, O., Moratto, Z. M., Smith, B. E., Joughin, I. R., Porter, C., Morin, P., 2016. An automated, open-source pipeline for mass production of digital elevation models (DEMs) from very-high-resolution commercial stereo satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* 116: 101–117.
- Simoncelli, E., Adelson, E., Heeger, D., 1991. Probability distributions of optical flow. In: Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 310–315.
- Slabaugh, G. G., Culbertson, W. B., Malzbender, T., Stevens, M. R., Schafer, R. W., 2004. Methods for volumetric reconstruction of visual scenes. *International Journal of Computer Vision* 57 (3): 179–199.
- Song, X., Yang, G., Zhu, X., Zhou, H., Wang, Z., Shi, J., 2020a. Adastereo: a simple and efficient approach for adaptive stereo matching. *arXiv preprint arXiv:2004.04627*.
- Song, X., Zhao, X., Fang, L., Hu, H., Yu, Y., 2020b. EdgeStereo: An effective multi-task learning network for stereo matching and edge detection. *International Journal of Computer Vision* 128: 910–

- 930.
- Springenberg, J., Dosovitskiy, A., Brox, T., Riedmiller, M., 2015. Striving for simplicity: The all convolutional net. In: International Conference on Learning Representations (ICLR, workshop track).
- Spyropoulos, A., Komodakis, N., Mordohai, P., 2014. Learning to detect ground control points for improving the accuracy of stereo matching. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1621–1628.
- Tasdizen, T., Whitaker, R., 2004. Higher-order nonlinear priors for surface reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (7): 878–891.
- Tonioni, A., Poggi, M., Mattocchia, S., Di Stefano, L., 2017. Unsupervised adaptation for deep stereo. In: 2017 IEEE International Conference on Computer Vision (ICCV), 1614–1622.
- Tonioni, A., Rahnama, O., Joy, T., Di Stefano, L., Ajanthan, T., Torr, P. H., 2019. Learning to adapt for stereo. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 9653–9662.
- Tonioni, A., Tosi, F., Poggi, M., Mattocchia, S., Stefano, L. D., 2019. Real-time self-adaptive deep stereo. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 195–204.
- Tosi, F., Poggi, M., Benincasa, A., Mattocchia, S., 2018. Beyond local reasoning for stereo confidence estimation with deep learning. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Hrsg.), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 323–338.
- Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y., 2004. Support vector machine learning for interdependent and structured output spaces. In: *Proceedings of the Twenty-First International Conference on Machine Learning (ICML)*, ICML '04, Association for Computing Machinery, New York, NY, USA, 104.
- Tsoumakas, G., Katakis, I., 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)* 3 (3): 1–13.
- Tulyakov, S., Ivanov, A., Fleuret, F., 2017. Weakly supervised learning of deep metrics for stereo reconstruction. In: 2017 IEEE International Conference on Computer Vision (ICCV), 1348–1357.
- Tulyakov, S., Ivanov, A., Fleuret, F., 2018. Practical deep stereo (PDS): Toward applications-friendly deep stereo matching.
- Veksler, O., 2002. Stereo correspondence with compact windows via minimum ratio cycle. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (12): 1654–1660.
- Vinny, M., Singh, P., 2020. Review on the artificial brain technology: Bluebrain. *Journal of Informatics Electrical and Electronics Engineering* 1 (1): 3.
- Viola, P., Wells, W., 1995. Alignment by maximization of mutual information. In: *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 16–23.
- Vogiatzis, G., Torr, P., Cipolla, R., 2005. Multi-view stereo via volumetric graph-cuts. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, 391–398 vol. 2.
- Wang, R., Pizer, S. M., Frahm, J.-M., 2019. Recurrent neural network for (un-)supervised learning of monocular video visual odometry and depth. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5550–5559.
- Wang, Y., Lai, Z., Huang, G., Wang, B. H., van der Maaten, L., Campbell, M., Weinberger, K. Q., 2019. Anytime stereo image depth estimation on mobile devices. In: 2019 International Conference on Robotics and Automation (ICRA), 5893–5900.
- Wu, Z., Wu, X., Zhang, X., Wang, S., Ju, L., 2019. Semantic stereo matching with pyramid cost volumes. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 7483–7492.
- Wulff, J., Butler, D. J., Stanley, G. B., Black, M. J., 2012. Lessons and insights from creating a synthetic optical flow benchmark. In: Fusiello, A., Murino, V., Cucchiara, R. (Hrsg.), *Computer Vision – ECCV 2012. Workshops and Demonstrations*, Springer Berlin Heidelberg, Berlin, Heidelberg, 168–177.
- Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F., Reinartz, P., 2020. Multi-label learning based semi-global matching forest. *Remote Sensing* 12 (7).
- Xia, Y., Tian, J., d'Angelo, P., Reinartz, P., 2018. Dense matching comparison between census and a

- convolutional neural network algorithm for plant reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2*: 303–309.
- Xie, C., Zhou, H., Wu, J., 2018. Vortex pooling: Improving context representation in semantic segmentation.
- Xu, D., Ouyang, W., Wang, X., Sebe, N., 2018. PAD-Net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 675–684.
- Xu, L., Yan, Q., Jia, J., 2013. A sparse control model for image and video editing. *ACM Transactions on Graphics* 32 (6): 1–10.
- Yang, G., Manela, J., Happold, M., Ramanan, D., 2019. Hierarchical deep stereo matching on high-resolution images. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5510–5519.
- Yang, G., Zhao, H., Shi, J., Deng, Z., Jia, J., 2018. SegStereo: Exploiting semantic information for disparity estimation. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, 636–651.
- Yang, R., Pollefeys, M., 2003. Multi-resolution real-time stereo on commodity graphics hardware. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2003. *Proceedings.*, Vol. 1, I–I.
- Ye, X., Li, J., Wang, H., Huang, H., Zhang, X., 2017. Efficient stereo matching leveraging deep local and context information. *IEEE Access* 5: 18745–18755.
- Yin, Z., Shi, J., 2018. GeoNet: Unsupervised learning of dense depth, optical flow and camera pose. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 1983–1992.
- Yoon, K.-J., Kweon, I.-S., 2005. Locally adaptive support-weight approach for visual correspondence search. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 2, 924–931 vol. 2.
- Yu, F., Wang, D., Shelhamer, E., Darrell, T., 2018. Deep layer aggregation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2403–2412.
- Yu, L., Wang, Y., Wu, Y., Jia, Y., 2018. Deep stereo matching with explicit cost aggregation sub-architecture.
- Zabih, R., Woodfill, J., 1994. Non-parametric local transforms for computing visual correspondence. In: Eklundh, J.-O. (Hrsg.), *Computer Vision – ECCV '94*, Springer Berlin Heidelberg, Berlin, Heidelberg, 151–158.
- Zagoruyko, S., Komodakis, N., 2015. Learning to compare image patches via convolutional neural networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4353–4361.
- Zbontar, J., LeCun, Y., 2015. Computing the stereo matching cost with a convolutional neural network. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1592–1599.
- Zbontar, J., LeCun, Y., 2016. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research* 17: 1–32.
- Zhan, W., Ou, X., Yang, Y., Chen, L., 2019. DSNet: Joint learning for scene segmentation and disparity estimation. In: 2019 International Conference on Robotics and Automation (ICRA), 2946–2952.
- Zhang, F., Prisacariu, V., Yang, R., Torr, P. H. S., 2019. GA-Net: Guided aggregation net for end-to-end stereo matching. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 185–194.
- Zhang, F., Qi, X., Yang, R., Prisacariu, V., Wah, B., Torr, P., 2020. Domain-invariant stereo matching networks. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (Hrsg.), *Computer Vision – ECCV 2020*, Springer International Publishing, Cham, 420–439.
- Zhang, F., Wah, B. W., 2018. Fundamental principles on learning new features for effective dense matching. *IEEE Transactions on Image Processing* 27 (2): 822–836.
- Zhang, L., Wang, Q., Lu, H., Zhao, Y., 2019. End-to-end learning of multi-scale convolutional neural network for stereo matching.
- Zhao, C., Sun, Q., Zhang, C., Tang, Y., Qian, F., 2020. Monocular depth estimation based on deep

- learning: An overview. *Science China Technological Sciences* : 1–16.
- Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6230–6239.
- Zhong, Y., Li, H., Dai, Y., 2018. Open-world stereo video matching with deep rnn. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Hrsg.), *Computer Vision – ECCV 2018*, Springer International Publishing, Cham, 104–119.
- Zhou, C., Zhang, H., Shen, X., Jia, J., 2017. Unsupervised learning of stereo matching. In: 2017 IEEE International Conference on Computer Vision (ICCV), 1576–1584.
- Zitnick, C. L., Kang, S. B., Uyttendaele, M., Winder, S., Szeliski, R., 2004. High-quality video view interpolation using a layered representation. In: *SIGGRAPH 2004*.
- Zitzler, E., Thiele, L., 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation* 3 (4): 257–271.

# Acknowledgement

The past several years of my Ph.D built a solid foundation for my research career, making me stronger to face every challenge through my life. Here, I would like to attribute all my harvest in this unforgettable period to my supervisors, colleagues, families and friends. Without them, I would not have succeeded.

Foremost, I would like to express my warmest gratitude to Prof. Peter Reinartz, for granting me this great opportunity to work at the German Aerospace Center (DLR) and pursue my Ph.D. He gave me not only the technical supervision, but also the support in every aspect for my work and life. The same gratitude goes to my principle supervisor Prof. Richard Bamler, who provided me with precious guidance throughout my Ph.D to avoid detours on my way, and my second supervisor Prof. Friedrich Fraundorfer for his valuable comments and corrections on my work in every discussion. I am honored to have Prof. Urs Hugentobler to chair my Ph.D colloquium.

I would like to sincerely thank Dr. Pablo d'Angelo for his helpful instructions on my Ph.D work. He was always waiting to help me for figuring out a new way, whenever I got stuck on research problems. His strict scientific attitude and rich experience deeply impressed me, and gave me a positive effect to continuously improve my work and deal with various challenges. This thesis would not have been completed without his support and constant encouragement.

My heartfelt gratitude goes to Dr. Jiaojiao Tian. She was the first colleague I met, who opened up my Ph.D life and kept helping me in the past few years. My rookie year would not have gone that smoothly without her invaluable suggestions. She encouraged me to participate scientific conferences, summer schools and courses, which helped me to extend my view and build close relationship with different outstanding researchers. Besides, I benefited a lot in our discussions to improve my scientific ability. With her continuous support, I never lost my confidence and enthusiasm on my research.

It is such a pleasure to work in the department of Photogrammetry and Image Analysis. I would be always grateful to my amazing colleagues Dr. Xiangyu Zhuo, Dr. Ksenia Bittner, Ms. Sabine Knickl, Ms. Guichen Zhang, Mr. Xiangtian Yuan, Mr. Yuxing Xie, Mr. Mario Fuentes Reyes, Mr. Peter Schwind, Mr. Maximilian Langheinrich, Dr. Emiliano Carmona and Dr. Franz Kurz. I would never forget their beautiful cakes, joyful talks, tasty coffee and talented shows in every Christmas party. Many thanks to my project partners Dr. Thomas Schneider, Mr. Emanuel Jachmann and Mr. Christian Kempf from Technische Universität München, and Prof. Joachim Hill and Dr. Henning Buddenbaum from Universität Trier. The collaboration with them was productive and happy to smoothly finish our project. I would also like to thank Prof. Marc Pollefeys and Dr. Viktor Larsson, for offering me the opportunity to have my academic visit in Swiss Federal Institute of Technology in Zürich.

Last but not the least, I gave my deep gratitude to my friends Dr. Song Liu, Ms. Meijie Fu, Dr. Jian Xu, Dr. Jingliang Hu, Dr. Danfeng Hong and Mr. Zhibin Cheng. I was so benefited from their research advice and really relaxed from every coffee break and party with them. I would like to specifically thank my parents and my girlfriend Chaonan Ji. They always stand with me to face the difficulties in my work and life in every second. Any word is so pale to express my gratitude. I am luck to have you in my life.

Danke!

谢谢!

## List of Figures

- 1.1 Stereo results of an indoor tree. The two images on the left constitute a rectified stereo pair, in which the corresponding points lie on the same row. The stereo disparity map is displayed on the right, from which the depth of the scene can be perceived. The color bar at the bottom represents the disparity range. 2
- 1.2 Challenges in stereo matching, including repetitive texture as the grids on the flag, part of which is occluded by the tree crown (occlusion also happens among leaves), reflective surface on the window, and textureless region on the wall. 3
- 2.1 Binocular stereo 3D reconstruction (Bleyer and Breiteneder, 2013). Stereo matching locates the corresponding pixels from two 2D images. The disparity is thus calculated as the relative displacement in between, from which a 3D view of the scene could be recovered. 6
- 2.2 Disparity determination through the disparity space for locating the correspondences. The element  $(x_0, y_0, d_0)$  in the disparity space should have the highest probability within the DSI, assuming  $(x_0, y_0)$  and  $(x_0 - d_0, y_0)$  are the corresponding projections from the object point  $(X, Y, Z)$  on the reference and target frame, respectively. 7
- 2.3 Stereo epipolar geometry. The corresponding point for  $q_1$  could be located exclusively through the epipolar line  $L_2$  in another image. Thus, the stereo matching is highly simplified from 2D to 1D. 8
- 2.4 The ordering rule in stereo matching. Normally, the order of the image points from the same targets should be consistent in each of the image, as (a). However, the rule is not valid for transparent objects, e.g. in (b). 8
- 2.5 The cost aggregation along a single scanline and the strategy in SGM to visit each pixel through multiple scanlines, assuming 16 scanlines are used (Hirschmüller, 2008). From the image border, the disparity is continuously predicted for each pixel to support the next pixel's estimation along a directed path for smoothness. With the same procedure repeated along multiple scanlines, the 2D smoothness is approximated for each pixel. 12
- 2.6 Deep learning assisted stereo matching (Zbontar and LeCun, 2016; Gidaris and Komodakis, 2017; Seki and Pollefeys, 2017; Schönberger et al., 2018; Knöbelreiter and Pock, 2019). Along the conventional processing pipeline, each module can be supervised for better feature representation to calculate the matching cost, smarter strategy to aggregate neighboring pixels and penalize disparity inconsistency, and learning based post-processing to refine the results. 13
- 2.7 Representative end-to-end neural networks for stereo matching (Kendall et al., 2017; Chang and Chen, 2018; Zhang et al., 2019). The entire stereo matching procedure is fully differentiable and trainable, from feature extraction, cost volume generation and regularization, to disparity prediction at the end. Thus, a disparity map can be output directly from a stereo pair. 14

- 2.8 Monocular and multi-view stereo acquisition. In monocular stereo, only a certain view of the scene is obtained, which makes the depth sensing an ill-posed problem. In multi-view stereo, a sequence of images (at least two) are captured around the target scene or object. Detailed 3D reconstruction can be achieved with fewer occlusions and reduced stereo errors thanks to additional views. 15
- 3.1 Basic network architecture for matching cost computation. A Siamese network is usually used as a backbone, with two branches sharing the same architecture and weights to extract features from each of the patches. Afterwards, the feature can be directly compared for matching cost calculation, e.g. via a correlation layer, or concatenated together to feed a comparison sub-network to adaptively learn a similarity measure. 18
- 3.2 Matching constraints along an epipolar line (Tulyakov et al., 2017), including epipolar constraint, disparity range constraint, uniqueness constraint, continuity constraint and ordering constraint. 20
- 3.3 The disparity estimation using a single scanline. Although the raw estimation from each single scanline is noisy, it can be found that the results from the "left to right" scanline is slightly better than "top to bottom", within the marked region. The reason is that the depth distribution is more stable horizontally than vertically. 22
- 4.1 The architecture of MC-CNN. With the same sub-networks, composed of a series of convolutional layers and rectified linear units, a feature vector can be generated for each of the input image patches. Then, a similarity score is computed at the end, either simply based on a dot product of the normalized feature vectors or another sub-network to learn the similarity measure during training. The latter architecture achieves better accuracy at the cost of relatively higher complexity. 44
- 4.2 The flow chart for ground truth disparity map generation using LiDAR point cloud. Starting from the experiment stereo images, the disparity maps are generated using SGM with Census and MC-CNN pre-trained on the Middlebury data sets, respectively. Afterwards, a pixel-wise average of the two maps is computed, and projected into the object space to obtain a point cloud. The laser point cloud is registered to this newly generated point cloud. Thus, the ground truth disparity map is acquired via projecting the registered LiDAR point cloud onto the epipolar image planes. 46
- 4.3 The flow chart for the self-training strategy. Based on SGM and a pre-trained MC-CNN on Middlebury datasets, a disparity map is generated. Afterwards, a rigid left-right consistency check is applied to remove most outliers. Only the pixels left are regarded as accurate estimation (artificial ground truth) to further train the MC-CNN model, which is finally used to predict the disparity results. 47
- 4.4 The epipolar image pair for the first experiment. MicMac was utilized for camera calibration, relative orientation and epipolar image generation. 48
- 4.5 The disparity maps generated based on SGM with different strategies, Census, MC-CNN-Pre, MC-CNN-LiDAR and MC-CNN-SelfT for matching cost. Inconsistent matching (IM) is represented by the color white. 49



- 4.6 The reconstruction details of several selected leaves. From left to right in each subset: the first row includes the master epipolar image and disparity maps for Census and MC-CNN-Pre. The second row includes the ground truth and disparity maps for MC-CNN-LiDAR and MC-CNN-SelfT. In order to enhance the contrast of the disparity within each single leaf, we have used a different color bar for each leaf. Pixels invalidated by the left-right check are shown in white. 50
- 4.7 The epipolar image pair for the second experiment, which is collected from the test region of our project. 56
- 4.8 The disparity map generated using self-trained MC-CNN. Inconsistent matching (IM) is represented by the color white. 56
- 4.9 Leaves under drought stress. From left to right in each subset: the master epipolar image, the disparity map of the self-trained MC-CNN matching scheme, and the disparity profile along the red line. The color represents the disparity. From blue to yellow, the targets get closer to the camera. Pixels with inconsistent matching (IM) are shown in white color. 57
- 4.10 The comparison between each single scanline's disparity prediction and the ground truth, for pixels marked green in (a). It is found that both SO3 and SO7 accomplish good prediction and should be selected for further processing. 62
- 4.11 Stereo pair, cost cube and the corresponding aggregated cost cube in SGM. As more scanlines are considered, the memory usage is highly increased. 64
- 4.12 Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: Census). 68
- 4.13 Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: MC-CNN-acrt). 69
- 4.14 The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: Census). The red rectangles marked in the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM. 70
- 4.15 The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: MC-CNN-acrt). The red rectangles marked in the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM. 71
- 4.16 Stereo matching results on EuroSDR benchmark datasets (Vaihingen/Enz). The master epipolar image and the corresponding disparity results are displayed. The green rectangle marks the region for detailed comparison. 72
- 4.17 Results on stereo datasets from the 2019 IEEE GRSS data fusion contest (Track 2, pairwise semantic stereo challenge). The disparity and error maps are displayed, with the red rectangles highlighting the performance difference between SGM-ForestS and SGM-ForestM. 73
- 4.17 Results on stereo datasets from the 2019 IEEE GRSS data fusion contest (Track 2, pairwise semantic stereo challenge). The disparity and error maps are displayed, with the red rectangles highlighting the performance difference between SGM-ForestS and SGM-ForestM. (cont.) 74

- 4.18 GA-Net\_Pyramid with explicit downsampling. The input stereo pair is downsampled explicitly according to the resolution required by each pyramid level. At the pyramid top, the stereo correspondences are located within an absolute disparity range in low resolution. The following pyramid levels perform disparity refinement within a pre-defined residual disparity range until the original resolution is recovered at the pyramid bottom. (SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement.) 79
- 4.19 GA-Net\_Pyramid with implicit downsampling. The feature extractor is applied on the stereo pair in original resolution, with the intermediate feature maps from its decoder to feed each pyramid level according to the expected resolution. Thus, an implicit downsampling is achieved. (SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement.) 81
- 4.20 Visual comparison between GA-Net\_PyramidID+SPN and GA-Net\_Ori on Scene Flow data. In each subfigure, the disparity maps from the ground truth and each network are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. Regions where the proposed algorithm outperforms GA-Net\_Ori are marked with red arrows. 83
- 4.21 Visual comparison between GA-Net\_PyramidID+SPN and GA-Net\_Ori on KITTI-2012 data. In each subfigure, the disparity maps from the ground truth and each network are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. Regions reconstructed better by the proposed algorithm are marked using red arrows. 84
- 4.22 Visual comparison among GA-Net\_PyramidED, GA-Net\_Ori and SGM on aerial data. In each subfigure, the reference disparity map and each algorithm's stereo results are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. 87
- 4.23 Visual comparison among GA-Net\_PyramidED, GA-Net\_Ori and SGM on satellite data. In each subfigure, the reference disparity map and each algorithm's stereo results are displayed in the first row. The second row provides the master epipolar image and the corresponding error maps. 90
- 4.24 A showcase to indicate the ability of our pyramid network in processing remote sensing stereo pair with large baseline. The reconstruction results for the region with a size of 19791 x 15639 pixels highlighted by the green rectangle are shown below, from the reference disparity map (left) and our pyramid model (right). Test region: Matterhorn mountain, Switzerland. Test model: GA-Net\_PyramidED. 91

# List of Tables

4.1	The image acquisition parameters.	48
4.2	Mean of the disparity difference between dense matching and ground truth.	53
4.3	Median of the disparity difference between dense matching and ground truth.	53
4.4	STD of the disparity difference between dense matching and ground truth.	53
4.5	MAD of the disparity difference between dense matching and ground truth.	54
4.6	Evaluation of reconstruction completeness and accuracy for each dense matching scheme.	55
4.7	The image acquisition parameters.	55
4.8	The percentage of pixels with more than one scanline achieving good prediction for Middlebury and ETH3D benchmarks.	62
4.9	Train/validation splits for Middlebury benchmark.	66
4.10	Train/validation splits for ETH3D benchmark.	66
4.11	The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: Census. '-5dirs' for 5 scanlines version and '-8dirs' for 8 scanlines version).	67
4.12	The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: MC-CNN-acrt).	67
4.13	The benchmark results of SGM-ForestM on ETH3D datasets (Matching cost: Census).	68
4.14	Accuracy and efficiency comparison between GA-Net_Pyramid, including GA-Net_PyramidED and GA-Net_PyramidID, and GA-Net_Ori on Scene Flow data.	82
4.15	Accuracy and efficiency comparison between GA-Net_Pyramid, including GA-NetPyramidED and GA-NetPyramidID, and GA-Net_Ori on KITTI-2012 data.	84
4.16	Accuracy and efficiency comparison between GA-Net_Pyramid, including GA-Net_PyramidED and GA-Net_PyramidID, and GA-Net_Ori on aerial data (baseline model: SGM).	86
4.17	Accuracy and efficiency comparison for GA-Net_PyramidED with different pyramid levels.	86
4.18	Accuracy and efficiency comparison for GA-Net_PyramidED with different residual search ranges.	88
4.19	Accuracy and efficiency comparison between GA-Net_Pyramid, including GA-Net_PyramidED and GA-Net_PyramidID, and GA-Net_Ori on satellite data (baseline model: SGM).	89



## Appendices

- A Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2019. Self-supervised convolutional neural networks for plant reconstruction using stereo imagery. *Photogrammetric Engineering & Remote Sensing*, 85(5), pp.389-399.**

<https://www.ingentaconnect.com/content/asprs/pers/2019/00000085/00000005/art00016>



1        **Self-Supervised Convolutional Neural Networks for Plant**  
2                    **Reconstruction Using Stereo Imagery**

3  
4  
5  
6  
7  
8  
9

10    **Dense matching strategies combining convolutional neural**  
11    **networks and semi-global matching for plant reconstruction.**

12  
13  
14  
15  
16  
17  
18  
19  
20  
21

22 **Abstract:**

23 Stereo matching can provide complete and dense 3D reconstruction to study plant  
24 growth. Recently, high-quality stereo matching results were achieved combining semi-  
25 global matching with deep learning. However, due to a lack of suitable training data, this  
26 technique is not readily applicable for plant reconstruction. We propose a self-supervised  
27 MC-CNN scheme to calculate matching cost and test it for plant reconstruction. The MC-  
28 CNN network is re-trained using the initial matching results obtained from the standard  
29 MC-CNN weights. For the experiment, close-range photogrammetric imagery of an in-  
30 house plant is used. The results show that the performance of self-supervised MC-CNN is  
31 superior to the Census algorithm and comparable to MC-CNN trained by a LiDAR point  
32 cloud. Another experiment is performed using stereo imagery of a field beech tree. The  
33 proposed self-training strategy is tested and has proved capable of identifying the drought  
34 condition of trees from the reconstructed leaves.

35 **1 Introduction**

36 Forest management is an interdisciplinary topic involved in numerous fields such as  
37 environment, politics, economics, climate and ecology (Strigul, 2012). Remote sensing,  
38 as a technique to take measurements from a distance, is appropriate to assist forest  
39 management because it can observe the target with no need to approach it and provide  
40 time series data sets for constant monitoring. Spaceborne and airborne remote sensing  
41 instruments offer broad observation of trees to estimate the biomass, monitor the living  
42 condition, measure the forest canopy cover, etc. (Ahmed et al., 2014; Freeman et al.,



43 2016; Wu et al., 2016). Some high-resolution stereo imaging sensors are capable of  
44 deriving detailed digital surface models to acquire geometric parameters of the forest,  
45 however, only some large scale properties such as forest canopy height can actually be  
46 estimated (Tian et al., 2017).

47 In order to obtain detailed information about the forest, single tree growth patterns should  
48 be observed. The size, shape, color and leaf distribution of individual trees are all  
49 important factors and worth measuring in detail so that the health situation of the tree and  
50 even the whole ecosystem can be better understood (Levin, 1999; Gatziolis et al., 2015).  
51 The terrestrial Light Detection and Ranging (LiDAR) technique can provide accurate and  
52 dense point clouds of trees to support the geometric survey for tree-level parameters  
53 estimation (Kankare et al., 2013; Tao et al., 2015). Nevertheless, the data acquisition can  
54 require considerable manpower and material resources and can even be dangerous in  
55 extreme terrain. In the past decade, dense matching using optical stereo images has been  
56 widely used for 3D reconstruction. Among the different techniques, Semi-Global  
57 Matching (SGM) has outperformed most existing approaches in accuracy and efficiency  
58 (especially in remote sensing), and is used in many applications, for example building  
59 reconstruction, digital surface model generation, robot navigation and driver assistance  
60 (Hirschmüller, 2011; Kuschik et al., 2014; Qin et al., 2015). However, the performance  
61 varies when different matching cost calculation approaches are adopted. Many local  
62 features (e.g. Census, Mutual Information) have been used for the matching cost  
63 calculation (Hirschmüller, 2008; Hirschmüller and Scharstein, 2009). But, tree leaf

64 matching remains very difficult due to the lack of unique features, many occlusions and  
65 repetitive structure.

66 Convolutional Neural Networks (CNN) (LeCun et al., 1998) are a popular topic in  
67 computer vision and have been used to solve many vision problems. Recently, an  
68 algorithm computing Matching Cost based on CNN (MC-CNN) was proposed (Zbontar  
69 and LeCun, 2016) in which a net is trained with supervised learning based on pairs of  
70 small image patches with known true disparity. Combined with SGM, MC-CNN has  
71 proved to outperform most previous algorithms thanks to a good extraction of the local  
72 image features and a trained similarity measure to compare the extracted feature  
73 descriptors. However, the ground truth collection is always a bottleneck for deep neural  
74 network based algorithms, which require huge amount of labeled data to train the net  
75 (Krizhevsky et al., 2012; Knöbelreiter et al., 2018). Ground truth acquisition for tree  
76 reconstruction via LiDAR sensors is complicated by the long scanning time required for  
77 capturing a dense point cloud. Any tiny movement of the leaf or branch during the laser  
78 scanning will cause the scanned point cloud to be inconsistent with the images, which  
79 limits its use for further training and evaluation. Hence, in this paper we follow the work  
80 of (Knöbelreiter et al., 2018) and propose a dense matching strategy combining SGM and  
81 a self-trained MC-CNN for plant reconstruction.

82 This paper is organized as follows: The MC-CNN based dense matching and the  
83 proposed training schemes are described in Section 2. Section 3 describes an indoor and  
84 an outdoor experiment, which demonstrate the feasibility of the proposed self-training

85 strategy. Conclusions are drawn and an outlook for future research is provided in Section  
86 4.

## 87 **2 Methodology**

### 88 **2.1 Dense Matching**

89 Dense matching attempts at establishing correspondences between every pixel in the  
90 image pair (Scharstein and Szeliski, 2002). Together with the known camera orientations,  
91 a dense point cloud can be obtained. Most dense stereo matching algorithms consist of  
92 the following four steps: Firstly, a similarity measure between two potentially matching  
93 pixels is computed to evaluate the matching cost. Then as the matching cost can be  
94 ambiguous, costs are usually aggregated in a local neighborhood. Global stereo methods  
95 then apply regularization to the aggregated costs, while local methods simply select the  
96 correspondence with the lowest matching cost. SGM combines local and global methods  
97 by regularizing the aggregated costs before determining each correspondence. Afterwards  
98 for rectified stereo pairs, a disparity map containing the horizontal shifts between the  
99 images is obtained (Bolles et al., 1987; Okutomi and Kanade, 1993). Finally, subpixel  
100 interpolation, left-right consistency check and outlier filtering are applied by most stereo  
101 algorithms.

### 102 **2.2 CNN**

103 CNNs (LeCun et al., 1998) have been used to solve several vision problems such as  
104 classification (Krizhevsky et al., 2012), recognition (Lawrence et al., 1997), etc. It is

105 basically a feed-forward artificial neural network constructed by a sequence of layers  
106 with learnable weights and biases. A volume of activations are transformed into another  
107 when going through the layers, and finally certain scores are obtained as output at the end  
108 of the network, e.g. class scores for classification. Four types of layers are frequently  
109 used: (a) convolutional layers, in which each neuron is related to a local region of the  
110 input; (b) pooling layers, used to downsample the previous volume; (c) rectified linear  
111 units applying an elementwise activation function; and (d) fully-connected layers, which  
112 calculate the output by connecting each neuron to all the neurons of the previous volume  
113 for high-level reasoning. The network can be trained to reach its best performance with a  
114 sufficient amount of training samples.

## 115 **2.3 MC-CNN**

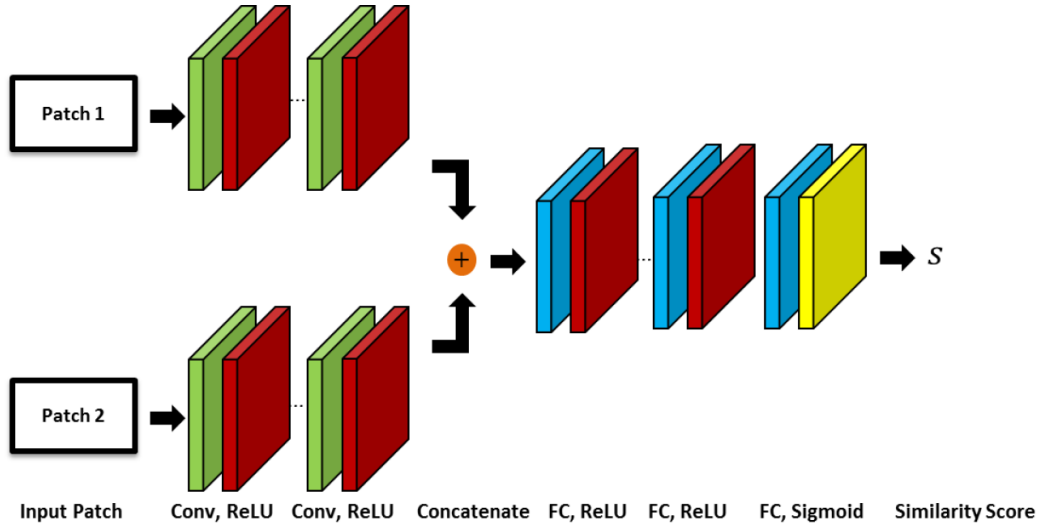
116 CNNs provide a new possibility in dense matching (Luo et al., 2016; Zbontar and LeCun,  
117 2016). Zbontar and LeCun (2016) proposed a dense stereo algorithm using a CNN based  
118 matching cost combined with SGM and additional post-processing steps, which  
119 outperformed most previous stereo matching algorithms. Therefore this algorithm is  
120 utilized as the main framework in this paper.

### 121 **2.3.1 Data Term**

122 A binary classification data set is constructed for training the net, based on either the  
123 KITTI (Geiger et al., 2013; Menze and Geiger, 2015) or the Middlebury (Scharstein and  
124 Szeliski, 2002, 2003; Scharstein and Pal, 2007; Hirschmüller and Scharstein, 2009;  
125 Scharstein et al., 2014) stereo data sets with available ground truth disparity maps. At

126 each image location, a positive and a negative training example are extracted. The  
127 positive example is a pair of patches from the left and right image respectively with the  
128 central pixels projected from the same object point, while the negative example is from a  
129 pair of patches where this geometric condition is not satisfied.

130 Two network architectures are designed and trained on the extracted training examples.  
131 Both of them are siamese networks with two sub-networks sharing the same weights  
132 (Bromley et al., 1993). The first two sub-networks transform a pair of image patches into  
133 two feature vectors describing the structure of each patch. The siamese network consists  
134 of several convolutional layers, each of which is followed by a rectified linear unit. The  
135 second part of the network computes the similarity measure using the two feature vectors.  
136 The first architecture uses the dot product of the normalized feature vectors as similarity  
137 measure. Therefore, it has a lower runtime and is called fast architecture. The second  
138 architecture, shown in Figure 1 and named accurate architecture, learns the similarity  
139 measure during training. The outputs of the two subnets are concatenated and passed  
140 through a number of fully-connected layers with a rectified linear unit following each of  
141 them. At the end, there is one more fully-connected layer which uses the sigmoid  
142 nonlinearity to produce the similarity score. In this paper, the accurate architecture is  
143 adopted due to the high-quality demand of plant reconstruction.



144

145 Figure 1. The accurate architecture computes the similarity score using fully connected  
 146 network layers.

147 The binary cross-entropy loss used for training is defined as

$$148 \quad l = t \cdot \log s + (1 - t) \cdot \log(1 - s), \quad (1)$$

149 in which  $l$  is the binary cross-entropy loss.  $s$ , the similarity score, represents the output of  
 150 the net. The value of  $t$  depends on the category of the training example being used, which  
 151 is equal to 1 for positive examples and 0 for negative examples. The hyperparameters  
 152 include the number of convolutional layers in each subnet (5), the number of feature  
 153 maps in each layer (112), the convolutional kernel size (3), the number of fully-connected  
 154 layers (3), the corresponding number of units in each full-connected layer (384), and the  
 155 input patch size (11×11). Zbontar and LeCun (2016) acquire the hyperparameters based  
 156 on manual search and simple scripts to help automate the process, which are also applied  
 157 in this paper.

158 **2.3.2 Smoothness Term**

159 SGM is used to regularize the disparity estimation using a piecewise constant smoothness  
160 term. SGM is a combination of local and global stereo matching methods (Hirschmüller,  
161 2008), and approximates a global 2D smoothness term by summation of 1 dimensional  
162 smoothness constraints on 8 or 16 directions. For each direction, assuming the target  
163 pixel is at location  $p$ , the cost is computed as:

$$L_r(p, d) = C(p, d) + \min(L_r(p - r, d), L_r(p - r, d - 1) + P_1,$$

164  $L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2),$  (2)

165 where  $L_r(p, d)$  is the cost along the path traversed in direction  $r$  for the pixel  $p$  at  
166 disparity  $d$  and  $C(p, d)$  is the matching cost.  $P_1$  represents a penalty when the previous  
167 pixel has a disparity difference of 1.  $P_2$  penalizes larger disparity differences. For each  
168 pixel  $p$ ,  $S(p, d) = \sum_r L_r(p, d)$  is computed and the disparity with the minimum  $S$  is  
169 selected.

170 SGM is selected as smoothness term due to its good performance and efficiency, its  
171 runtime is proportional to the reconstructed volume (d'Angelo and Reinartz, 2011;  
172 d'Angelo, 2016).  $C(p, d)$  is calculated using MC-CNN and then aggregated based on  
173 Cross-Based Cost Aggregation (CBCA) (Mei et al., 2011; Zbontar and LeCun, 2016). It  
174 should be noticed that  $S(p, d)$  undergoes CBCA once more before the final disparity  
175 determination.

176 **2.3.3 Disparity Computation and Refinement**

177 The disparity for each pixel is determined using the winner-takes-all strategy to generate  
178 a disparity map. Referring to Zbontar and LeCun (2016) and Mei et al. (2011), some  
179 post-processing steps are implemented to refine the quality of the disparity map,  
180 including interpolation, subpixel enhancement, a median filter, and a bilateral filter.

## 181 **2.4 Training Details**

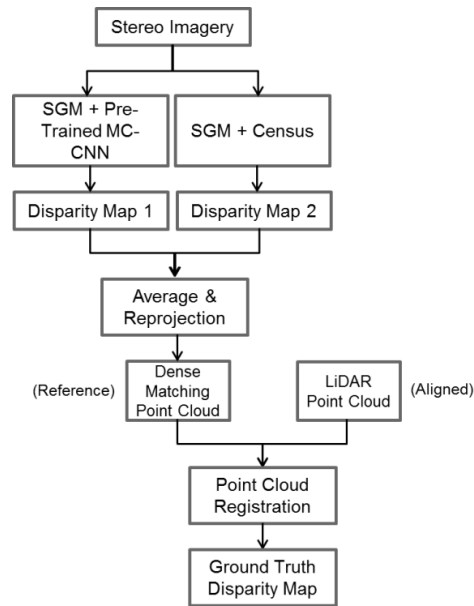
182 As for the training, two schemes are designed, of which one utilizes the ground truth  
183 from a LiDAR scanner to construct training data, while the self-training scheme directly  
184 uses the dense matching results of MC-CNN, pre-trained on the Middlebury data sets, to  
185 re-train the network. The reason for the two schemes is to test how the performance of  
186 MC-CNN can be improved by self-training and training with ground truth, respectively.

### 187 **2.4.1 LiDAR Training Scheme**

188 Zbontar and LeCun (2016) provide several nets pre-trained on the KITTI 2012, KITTI  
189 2015 and Middlebury data sets, respectively. The KITTI data sets focus on street views  
190 which do not fully match with our application. However, the Middlebury data focuses on  
191 static objects and the scenes exhibit a similar structure as our plant images, e.g. both  
192 concentrate on a certain target. Therefore, as one option we start from the pre-trained net  
193 on the Middlebury data sets and further train the net using the ground truth from LiDAR.  
194 In other words, we re-use the net pre-trained on the Middlebury data, and refine the  
195 network for plant reconstruction by further training. Thus the learning ability of the net  
196 for objects from a different category could also be tested.



197 As for the LiDAR scanning, a point cloud of the plant is generated to obtain the ground  
198 truth disparity map. As the image orientation and the LiDAR point cloud use different  
199 coordinate systems, a co-registration step is needed before the point cloud can be used.  
200 Besides, the main target is to test the performance of MC-CNN trained with different  
201 strategies for plant reconstruction and compare with a classic Census algorithm to  
202 demonstrate the effectiveness of MC-CNN. Hence as shown in Figure 2, we first generate  
203 two disparity maps based on SGM with Census and MC-CNN pre-trained on the  
204 Middlebury data sets. A pixel-wise average of both maps is computed and projected into  
205 3D space to obtain a point cloud. Then, the point cloud from the laser scanner is  
206 registered to this newly generated point cloud. The ground truth disparity map is obtained  
207 by projecting the registered laser scanning point cloud onto the epipolar image planes.  
208 We use CloudCompare (Girardeau-Montaut et al., 2005) to roughly align the two point  
209 clouds first, by scale matching, rotation, translation and manual point pair picking  
210 alignment. After the rough alignment, some objects (in our case, leaves), which are  
211 reconstructed well by both dense matching and LiDAR, and aligned close to each other  
212 already, are selected for a further fine registration based on the Generalized Iterative  
213 Closest Point (GICP) method (Segal et al., 2009). GICP is more robust and performs  
214 better than the standard ICP without loss of efficiency. Afterwards, only well registered  
215 leaves are kept to generate the ground truth as described in detail by section 3.1.3.



216

217

Figure 2. Flow chart for ground truth generation.

#### 218 2.4.2 Self-Training Scheme

219 Huge amounts of data are available to meet the need of CNN for training. However in  
 220 most cases, high performance is accomplished at the cost of substantial pre-processing  
 221 workloads to label the training examples. Therefore, many self-supervised concepts have  
 222 been proposed to avoid the time-consuming manual annotation (Joung et al., 2017; Zhou  
 223 et al., 2017; Knöbelreiter et al., 2018). Joung et al. (2017) exploited the correspondence  
 224 consistency between stereo images to pick samples during the training and guide the  
 225 network to compute matching cost. Zhou et al. (2017) randomly initialized a network and  
 226 adopted left-right consistency check to select suitable matching to train the net.  
 227 Knöbelreiter et al. (2018) constructed the training data using a pre-trained version of their  
 228 hybrid CNN-CRF model followed by a conservative consistency check to reject most

229 outliers. Based on that, their self-supervised network is able to improve the completeness  
230 and accuracy of the stereo reconstruction results on aerial imagery.

231 Very high resolution LiDAR point clouds are very difficult and expensive to capture  
232 especially in an outdoor environment. In addition, it is almost impossible to obtain  
233 perfectly matching image and LiDAR data due to the long scanning time and changes in  
234 the plant shape due to wind and other effects. Therefore, instead of using LiDAR data, a  
235 self-training procedure is applicable even to scenarios where ground truth acquisition is  
236 difficult or impossible. We use the MC-CNN as described in section 2.3, pre-trained on  
237 Middlebury, to generate disparity maps used for self-training. A left-right consistency  
238 check with a threshold of 1 pixel is used to filter most outliers:

$$239 \quad |d_p^L + d_q^R| \leq 1 \quad q = p - d_p^L, \quad (3)$$

240 where  $d_p^L$  is the disparity for pixel at location  $p$  in the disparity map regarding the left  
241 epipolar image as the master epipolar plane, while similarly  $d_q^R$  is calculated via dense  
242 matching regarding the right epipolar image as the master epipolar plane. Only pixels  
243 where left-right matching differs by less than 1 pixel are used as ground truth to further  
244 train MC-CNN.

### 245 **3 Experiments**

246 Two experiments demonstrate the feasibility of self-trained MC-CNN for plant  
247 reconstruction. The first experiment was carried out in an indoor laboratory environment.  
248 In this experiment, an 8-meter high tree standing in the atrium of a building was

249 photographed from above. At the same time, a LiDAR point cloud was captured from a  
250 similar position. The second experiment investigated stereoscopic images from the crown  
251 of a beech tree growing in a typical European forest.

## 252 **3.1 Experiment I**

### 253 **3.1.1 Data Set**

254 The main objective of this work is the three-dimensional reconstruction of trees and their  
255 leaves in the forest. In order to minimize the influence of environmental conditions, the  
256 first experiment investigates an 8-meter high deciduous tree inside a building. A digital  
257 high-resolution handheld camera (NIKON D5500) equipped with an 18 mm lens is used  
258 to acquire images from a bridge over the crown of the tree. An exposure time of 1/20  
259 seconds and an ISO speed rating of 400 was used. The acquired images are 4000 pixels in  
260 height and 6000 pixels in width. A stereo image pair with a baseline length of  
261 approximately 0.1 meters is taken from a distance of approximately 1 meter from the tree.  
262 Details about the image acquisition are available in Table 1. A Leica HDS7000 laser  
263 scanner is used to obtain a point cloud of the plant from a similar position. Capturing the  
264 point cloud with a point distance of 6.3 mm and a depth error of 0.4 mm RMS at a  
265 distance of 10 meters took about 10 minutes.

266 Table 1. The image acquisition parameters.

Camera model	NIKON D5500
Height	4000 pixels
Width	6000 pixels
Exposure time	1/20 sec
ISO speed rating	400

Focal length	18.0 mm
Object distance	$\approx 1$ m
GSD	0.02 cm/pixel
Baseline length	$\approx 0.1$ m

267

### 268 3.1.2 3D Reconstruction

269 The proposed dense matching approach requires epipolar images, where corresponding  
 270 pixels are located on the same image row. MicMac (Rosu et al., 2015) was utilized for  
 271 camera calibration, relative orientation and epipolar image rectification. The epipolar  
 272 images generated based on the stereo pair mentioned above are shown in Figure 3.



273

274 Figure 3. The epipolar image pair for dense matching.

275 Disparity maps have been calculated using the method described in sections 2.2 and 2.3  
 276 using 4 different matching costs:

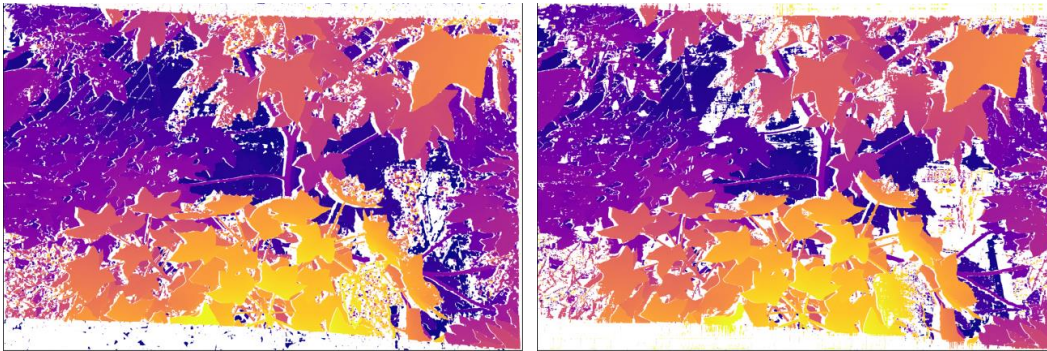
277 Census: Using only Census as matching cost;

278 MC-CNN-Pre: Using MC-CNN matching cost pre-trained on the Middlebury data sets;

279 MC-CNN-LiDAR: Using MC-CNN further trained on the LiDAR ground truth for  
 280 matching cost, as described in section 2.4.1;

281 MC-CNN-SelfT: Using MC-CNN further trained using the disparity maps of MC-CNN-  
282 Pre, as described in section 2.4.2.

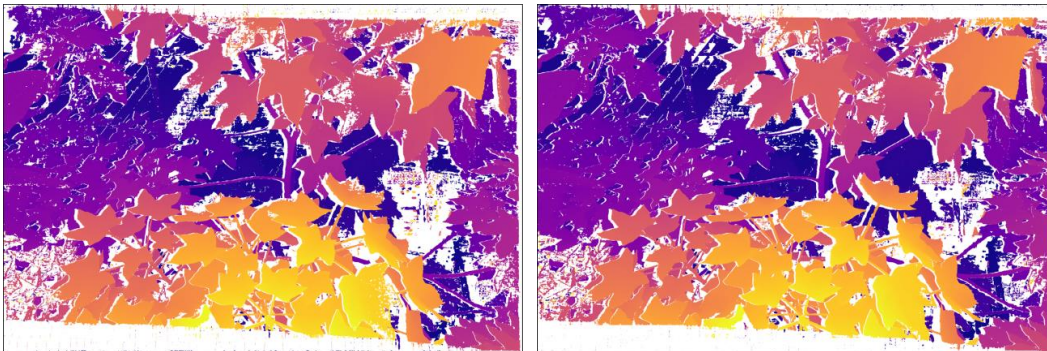
283 After the processing as described in section 2.3 and applying the left-right consistency  
284 check as described in section 2.4.2, the generated disparity maps for the epipolar image  
285 pair in Figure 3 are shown in Figure 4. For pixels with valid matching, the calculated  
286 disparity values from -91 to +42 are represented by the color from blue to yellow  
287 accordingly.



288  
289

(a) Census

(b) MC-CNN-Pre



290  
291

(c) MC-CNN-LiDAR

(d) MC-CNN-SelfT



292

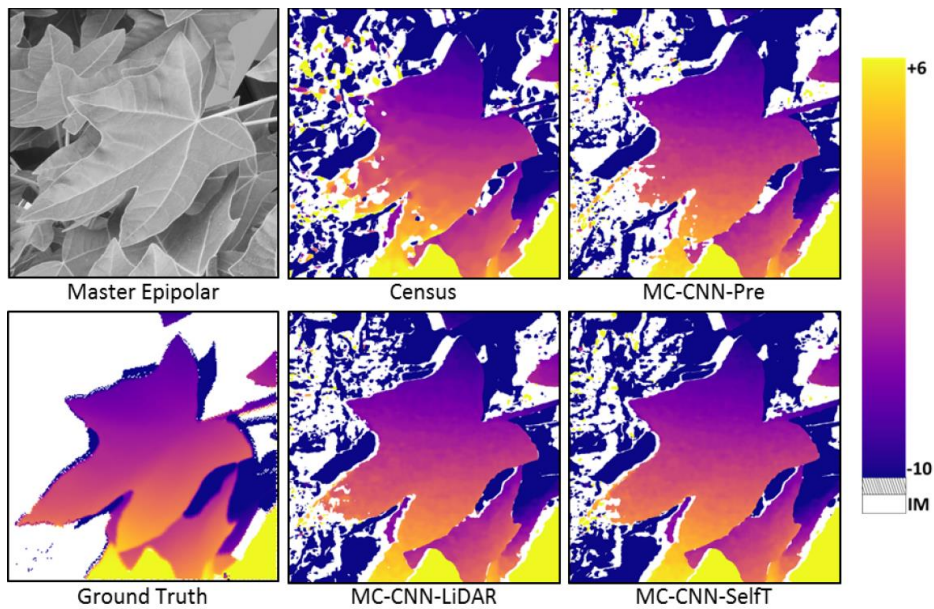
293 Figure 4. The disparity maps generated based on SGM with different strategies for  
294 matching cost. Inconsistent matching (IM) is represented by the color white.

### 295 **3.1.3 Evaluation and Discussion**

296 Training and evaluation of the different methods is hampered by systematic differences  
297 between LiDAR and stereo pairs. Due to the automatic air conditioning of the building  
298 there were small movements of the branches and leaves during LiDAR recording which  
299 took around 10 minutes. These led to slightly different leaf positions between LiDAR and  
300 stereo images. During the generation of the ground truth disparity map, some errors are  
301 included unavoidably when picking up point pairs to align the point clouds initially. The  
302 fine registration with GICP can improve the co-registration but errors still exist. Due to  
303 these problems, the point cloud registration is not perfect which influences the use of the  
304 ground truth disparity map generated from the LiDAR data. This is also the reason that  
305 we determine to only focus on some selected leaves after rough alignment to do GICP, as  
306 mentioned in section 2.4.1. Afterwards the relatively well registered leaves by GICP, that  
307 visually show merely small shift between the point clouds, are utilized for training and  
308 evaluation of the methods, which alleviates the problem mentioned above. This is in  
309 accordance with our application, as the shape of the leaves is the major indicator of plant  
310 health. Compared with images from the Middlebury data sets with sizes of around  
311  $300 \times 200$  to  $3000 \times 2000$  pixels, our images are larger ( $6000 \times 4000$  pixels), and the  
312 masked leaves can still provide a good amount of application specific training data. Thus,  
313 we use 13 well registered leaves together with Jadeplant and Sword1 data (containing a

314 plant, belonging to the Middlebury data sets 2014) as training data. The reason for adding  
315 the Middlebury data into the newly generated data sets is to increase the amount of  
316 training data from limited selected leaves.

317 A visual comparison of the results in Figure 4 shows that the tree was well reconstructed  
318 by all matching schemes. The results of five independent leaves not used during training  
319 on the LiDAR ground truth are shown in Figure 5. While most parts of the leaves are well  
320 reconstructed, some differences in completeness and amount of outliers are visible.

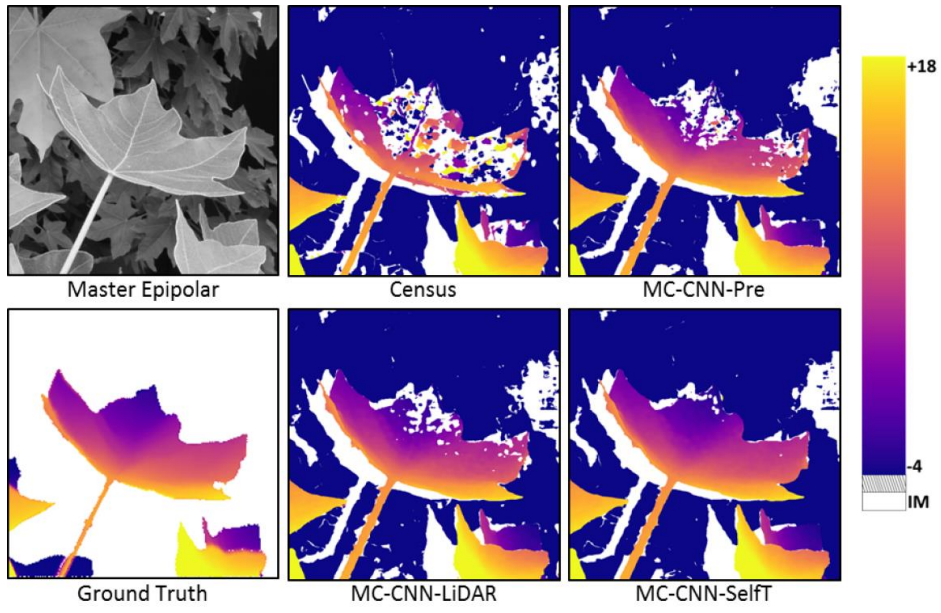


321

322

Leaf (a)

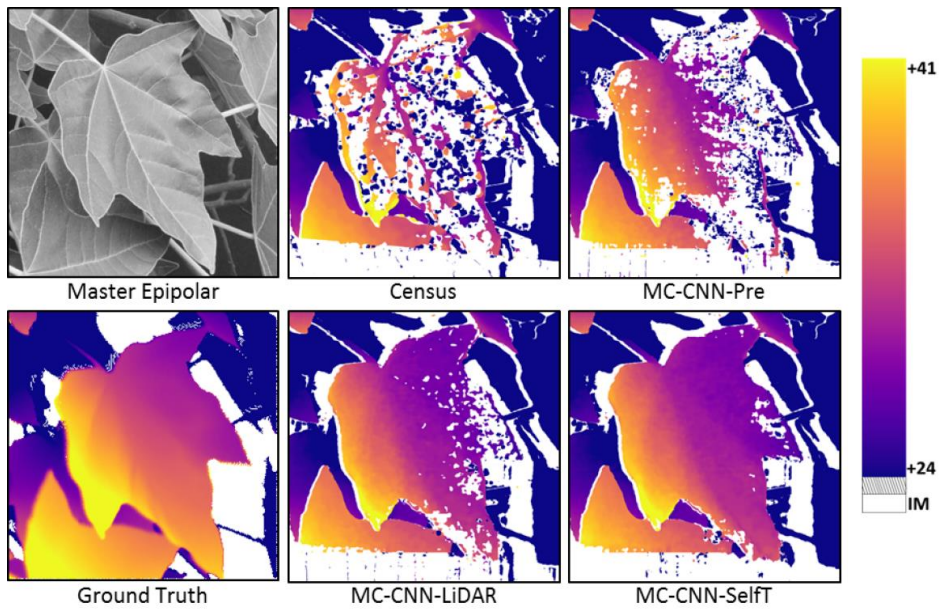




323

324

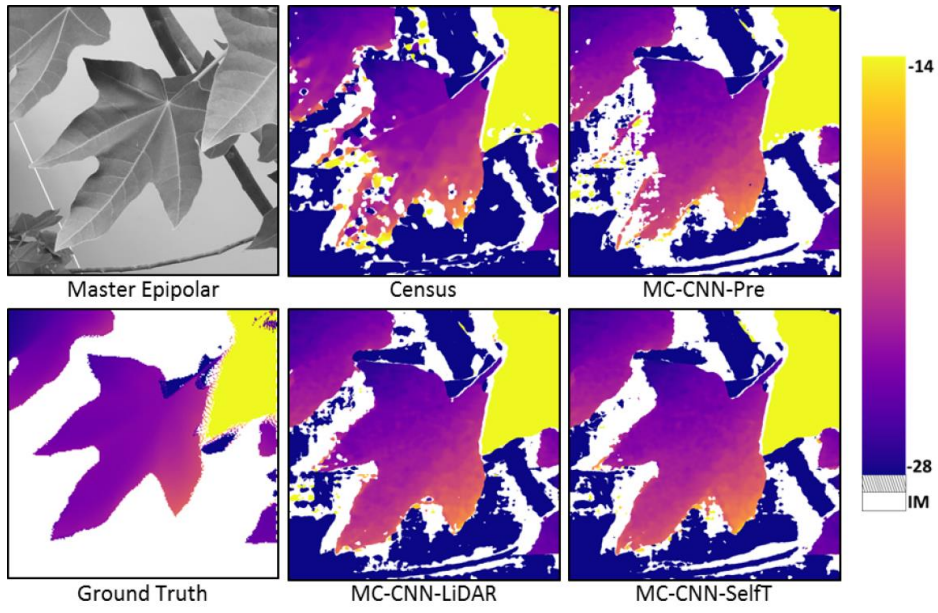
Leaf (b)



325

326

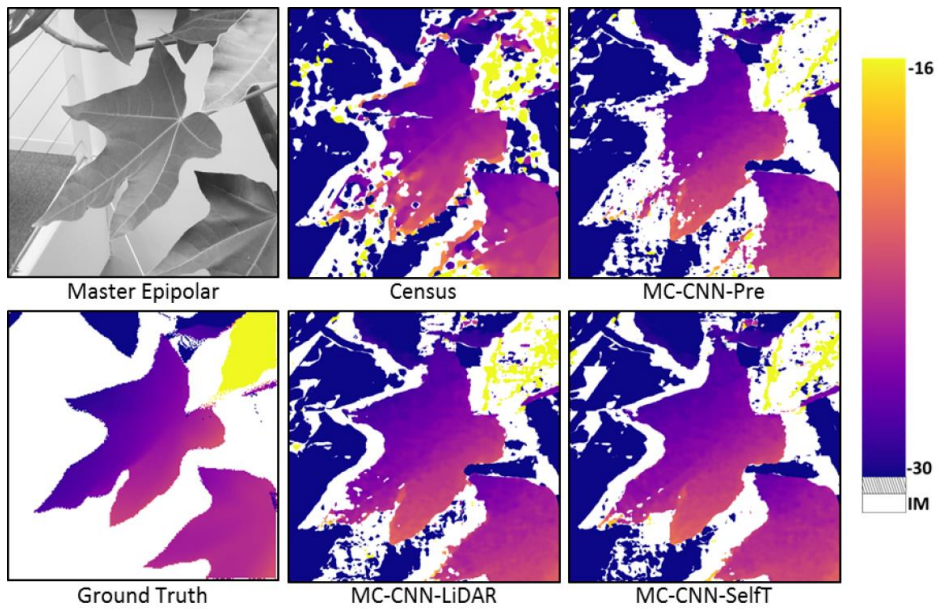
Leaf (c)



327

328

Leaf (d)



329

330

Leaf (e)

331 Figure 5. The reconstruction details of several selected leaves. From left to right in each

332 subset: the first row includes the master epipolar image and dense matching results for

333 Census and MC-CNN-Pre. The second row includes the ground truth and dense matching  
334 results for MC-CNN-LiDAR and MC-CNN-SelfT. In order to enhance the contrast of the  
335 disparity within each single leaf, we have used a different colorbar for each leaf. Pixels  
336 invalidated by the left-right check are shown in white.

337 From a visual inspection, it is found that the disparity values obtained by all four  
338 strategies match with the ground truth. With Census as matching cost, the main shape of  
339 the leaf is reconstructed but with considerable noise and low completeness. MC-CNN-Pre  
340 results in low completeness, cf. leaf (e), but shows less noise. However when fed with  
341 specific data for further training, MC-CNN-LiDAR and MC-CNN-SelfT achieve higher  
342 reconstruction completeness. MC-CNN-SelfT results in a slightly better leaf  
343 reconstruction than MC-CNN-LiDAR and fewer gaps. We would like to point out two  
344 reasons for this behavior: Firstly, in self-training more training samples are available for  
345 the net to develop the ability to learn new feature and calculate the similarity score. In  
346 Figure 4, it can be seen that all leaves are reconstructed or partially reconstructed in MC-  
347 CNN-Pre. Hence, the further trained MC-CNN can learn from each single leaf during the  
348 training and recover more area. Besides the rigid left-right consistency check, applied to  
349 the dense matching results of MC-CNN-Pre to construct training samples, guarantees a  
350 reasonable training procedure for MC-CNN-SelfT.

351 A quantitative evaluation is performed by comparing the generated disparity maps with  
352 the disparity maps obtained from LiDAR. The leaves (a) – (e) shown above are used for  
353 comparison. Firstly, the disparity difference  $D_p$  is calculated as below in units of pixels:

354 
$$D_p = d_p - d_p^G \quad p \in N_p, \quad (4)$$

355 where  $d_p$  denotes the disparity value of a pixel at location  $p$  calculated using one of the  
 356 four dense matching schemes.  $d_p^G$  is the corresponding ground truth disparity value.  $N_p$  is  
 357 the set of pixels where both dense matching and ground truth provide disparity values.  
 358 The mean ( $D_{mean}$ ), median ( $D_{median}$ ), standard deviation ( $D_{STD}$ ) and median absolute  
 359 deviation ( $D_{MAD}$ ) of the disparity differences are computed for comparison.

360 
$$D_{mean} = mean(D_p) \quad (5)$$

361 
$$D_{median} = median(D_p) \quad (6)$$

362 
$$D_{STD} = \sqrt{mean((D_p - D_{mean})^2)} \quad (7)$$

363 
$$D_{MAD} = median(|D_p - D_{median}|). \quad (8)$$

364 The results are reported in Tables 2 to 5.

365 Table 2. Mean of the disparity difference between dense matching and ground truth.

	$D_{mean}$ (pixels)			
leaf	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
(a)	0.28	-0.23	<b>0.05</b>	0.17
(b)	-6.78	-4.96	-2.32	<b>-1.88</b>
(c)	-13.88	-14.32	-3.73	<b>-3.13</b>
(d)	<b>0.35</b>	0.72	0.50	0.64
(e)	-0.15	<b>0.14</b>	0.30	0.46

366

367 Table 3. Median of the disparity difference between dense matching and ground truth.

	$D_{median}$ (pixels)			
leaf	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
(a)	0.11	-0.11	-0.10	<b>-0.00</b>
(b)	-1.78	-1.72	-2.02	<b>-1.57</b>
(c)	-3.91	-3.30	-3.54	<b>-3.12</b>
(d)	<b>0.32</b>	0.48	0.40	0.57
(e)	<b>0.06</b>	0.29	0.28	0.40

368

369 Table 4. STD of the disparity difference between dense matching and ground truth.

	$D_{STD}$ (pixels)			
leaf	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
(a)	4.49	4.48	<b>2.37</b>	2.76
(b)	19.61	15.02	1.29	<b>1.28</b>
(c)	25.53	30.65	7.86	<b>6.38</b>
(d)	2.73	3.16	<b>1.06</b>	1.13
(e)	5.35	2.84	<b>0.70</b>	0.86

370

371 Table 5. MAD of the disparity difference between dense matching and ground truth.

	$D_{MAD}$ (pixels)			
leaf	Census	MC-CNN-Pre	MC-CNN-LiDAR	MC-CNN-SelfT
(a)	0.76	<b>0.57</b>	<b>0.57</b>	0.63
(b)	3.03	0.51	0.42	<b>0.40</b>
(c)	3.49	0.64	<b>0.63</b>	<b>0.63</b>
(d)	0.73	0.67	<b>0.60</b>	0.65
(e)	0.50	0.46	<b>0.43</b>	0.51

372

373 By comparing the results in Table 2 and Table 3, it can be observed that the median is as  
374 expected more robust to outliers than the mean (e.g. for leaf (c), all the  $D_{median}$  are  
375 around 3 pixels). Leaf (b) and (c) show a relatively large systematic disparity difference.  
376 This can be attributed to the systematic error caused by the shape change and imperfect  
377 point cloud registration of the ground truth disparity map.

378 The  $D_{STD}$  values in Table 4 show the robustness of MC-CNN-LiDAR and MC-CNN-  
379 SelfT, as they exhibit much lower  $D_{STD}$  than Census and MC-CNN-Pre.

380  $D_{MAD}$  has been widely used for depth map evaluation, as it is more robust to outliers than  
381  $D_{STD}$ . The disparity map generated from Census has a relatively high  $D_{MAD}$  for the leaves  
382 (b) and (c). This is due to the large amount of noise in the Census results, as visible in  
383 Figure 5.

384 In addition to the pixel-based direct comparison, the reconstruction completeness and the  
385 percentage of the accurately measured pixels are calculated. The reconstruction  
386 completeness is calculated using the formula (9).

$$387 \quad Cpl = \frac{n_{DM/G}}{n_G} \times 100\%, \quad (9)$$

388 where  $n_G$  denotes the number of pixels with a valid disparity value provided by the  
389 ground truth in each leaf.  $n_{DM/G}$  denotes the number of pixels where both dense matching  
390 and ground truth provide disparity values. Thus the completeness  $Cpl$  will be the  
391 percentage of pixels in ground truth which are reconstructed by the dense matching as  
392 well.

393 However due to the systematic error, the disparity difference  $D_p$  between dense matching  
394 and ground truth cannot be directly utilized for evaluation. Therefore, we remove the  
395 systematic disparity shift for each leaf before computing the percentage of accurate  
396 pixels.

397 
$$Acc = \frac{n_{pass}}{n_G} \times 100\% \quad (10)$$

398 
$$n_{pass} = \text{the \# of pixels if: } |D_p - D_{median\_mean}| \leq \varepsilon \quad (11)$$

399 
$$D_{median\_mean} = \text{mean}(D_{median\_scheme\ i}) \quad i \in \{1, 2, 3, 4\}, \quad (12)$$

400 where  $D_{median\_mean}$  is the mean of  $D_{median}$  calculated using each of the four matching  
 401 schemes for each leaf.  $n_{pass}$  counts the number of pixels with the deviation below  $\varepsilon$ , a  
 402 pre-defined threshold to evaluate the corresponding accuracy. In this paper,  $\varepsilon$  is set as 0.5  
 403 and 1 pixel respectively for the test. The results are shown in Table 6.

404 Table 6. Evaluation of reconstruction completeness and accuracy for each dense  
 405 matching scheme.

Algorithm	(a)			(b)			(c)			(d)			(e)		
	Cpl	Acc		Cpl	Acc		Cpl	Acc		Cpl	Acc		Cpl	Acc	
		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p		0.5 p	1 p
Census	92.0	31.8	57.0	63.0	14.8	23.9	49.7	7.6	14.0	92.0	36.4	56.9	89.7	43.3	71.0
MC-CNN-Pre	91.1	42.1	67.3	82.0	39.0	62.5	59.8	23.6	37.0	91.5	37.6	63.3	85.0	45.6	72.9
MC-CNN-LiDAR	96.9	<b>43.8</b>	<b>72.1</b>	89.2	<b>51.9</b>	70.7	86.4	34.5	60.5	<b>99.4</b>	<b>44.3</b>	<b>69.4</b>	97.1	<b>55.6</b>	<b>82.5</b>
MC-CNN-SelfT	<b>97.9</b>	41.0	67.0	<b>98.6</b>	51.0	<b>81.4</b>	<b>95.7</b>	<b>39.7</b>	<b>62.2</b>	<b>99.4</b>	41.9	67.8	<b>99.5</b>	47.9	77.4

406  
 407 MC-CNN-SelfT consistently obtains a slightly higher completeness than MC-CNN-  
 408 LiDAR, while MC-CNN-LiDAR obtains slightly higher accuracy values for most leaves,  
 409 except for leaves (b) and (c), where MC-CNN-SelfT shows significantly better  
 410 completeness and 1 pixel accuracy values. Both re-trained methods consistently

411 outperform Census and MC-CNN-Pre. This shows that especially MC-CNN-SelfT,  
412 which does not require additional LiDAR ground truth data, is a good approach for  
413 significantly improving the leaf reconstruction.

414 In this experiment, MC-CNN-LiDAR is handicapped due to imperfect ground truth,  
415 leading to disadvantages compared to the MC-CNN-SelfT method. We therefore assume  
416 that the scores for MC-CNN-LiDAR could be improved slightly by using a perfectly  
417 registered ground truth. However due to different registration errors for each leaf (cf.  
418 Table 3), the LiDAR trained network is not able to learn and correct for a systematic  
419 error between the LiDAR point cloud and the image data. We thus believe that the  
420 evaluation does not favor a specific method.

## 421 **3.2 Experiment II**

422 This work was performed as part of a project aiming at detecting the physiological and  
423 morphological status of trees under drought stress and studying the adaptation of forest  
424 areas to climate change. A major part of the project focuses on constructing a detailed  
425 and accurate 3D model of tree leaves in order to monitor the shape change when facing  
426 drought.

427 For this purpose, two nadir-viewing cameras are mounted on a crane system for stereo  
428 measurement. When the system is lifted above the trees, a stereo image pair of the tree  
429 crowns can be obtained. In order to test the feasibility of the stereo method described in  
430 this paper, a stereo image pair above a beech tree subject to slightly artificial drought



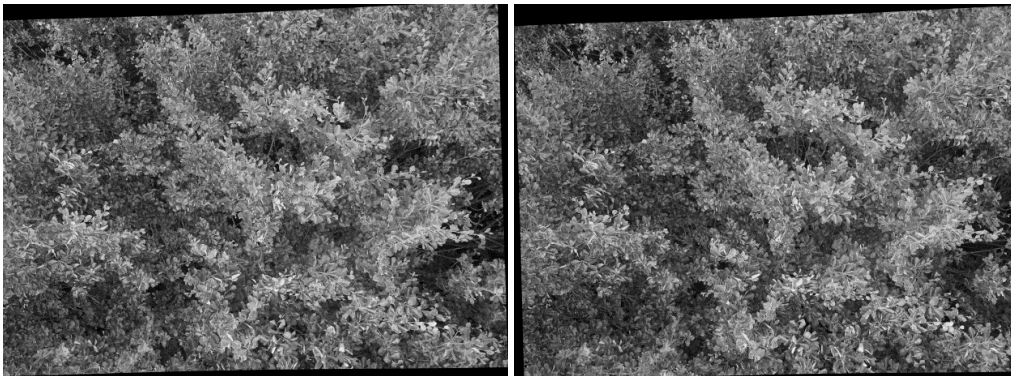
431 stress is collected. Some information about the images and the camera setting is shown in  
432 Table 7.

433 Table 7. Details about the image acquisition.

Camera model	SONY ILCE-5100
Height	4000 pixels
Width	6000 pixels
Exposure time	1/60 sec
ISO speed rating	125
Focal length	19.0 mm
Object distance	$\approx 3$ m
GSD	0.06 cm/pixel
Baseline length	$\approx 0.25$ m
Acquisition date	June 19 <sup>th</sup> , 2018

434

435 The corresponding epipolar image pair is shown in Figure 6. In this experiment, no  
436 LiDAR data is available, thus only Census, MC-CNN-Pre and MC-CNN-SelfT can be  
437 applied. The disparity map computed using MC-CNN-SelfT is shown in Figure 7.



438

439 Figure 6. An epipolar image pair from the test region of our project.

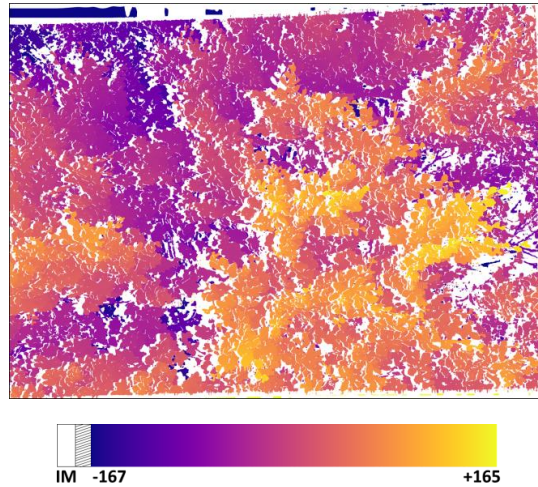
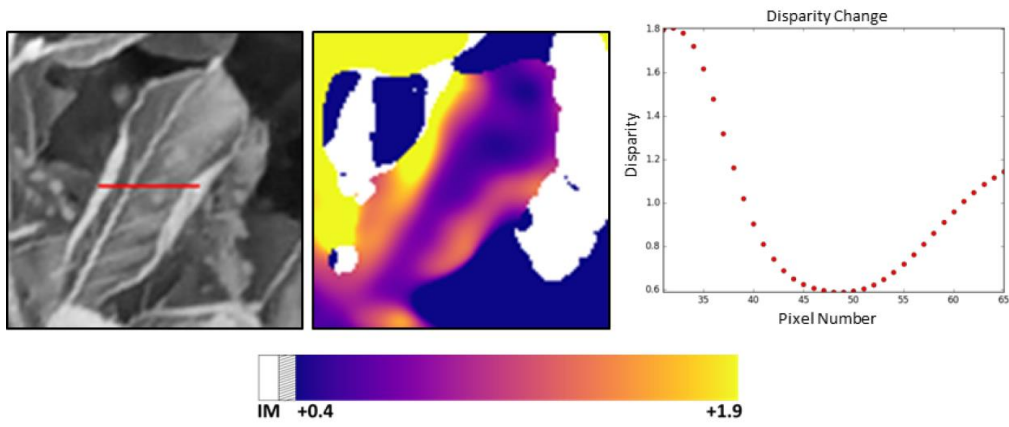


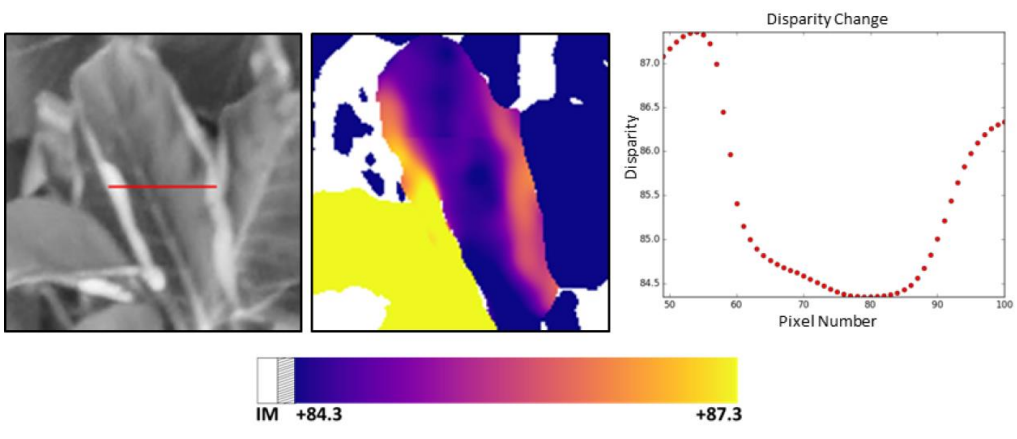
Figure 7. The disparity map generated using self-trained MC-CNN. Inconsistent matching is represented by the color white.

444 Figure 6 shows that the large beech tree crown is much more complex, and has much  
 445 smaller leaves than the indoor tree used in the first experiment. The slight drought stress  
 446 leads to multiple different leaf shapes. Under the hypothesis that curved leaves are an  
 447 indicator for drought stress, the stereo method should enable a clear separation of planar  
 448 and curved leaves. The generated disparity map provides a dense reconstruction of the  
 449 tree crown, and individual leaves are separable. The reconstruction completeness for MC-  
 450 CNN-Pre and MC-CNN-SelfT, are 76.0% and 78.7%, respectively. Due to the lack of  
 451 ground truth, the value is computed as the ratio of pixel passing the left-right check to the  
 452 number of valid pixels in the rectified image. Some leaves under drought stress are  
 453 selected for visual comparison. As shown in Figure 8, the curled shape of the leaves is  
 454 clearly visible in the disparity image and the profile plot.



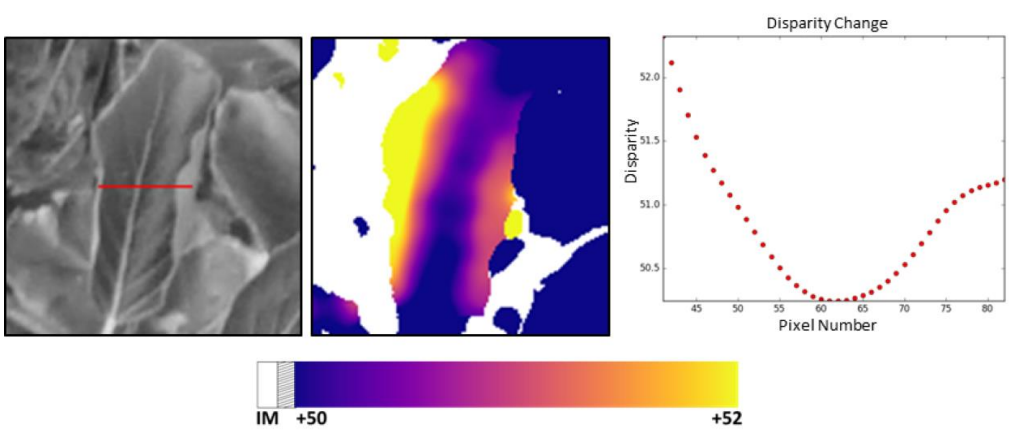
455  
456

(1)



457  
458

(2)



459  
460

(3)

461 Figure 8. Leaves under drought stress. From left to right in each subset: the master  
462 epipolar image, the disparity map of the self-trained MC-CNN matching scheme, and the  
463 disparity profile along the red line. The color represents the disparity. From blue to  
464 yellow, the targets get closer to the camera. Pixels with inconsistent matching are shown  
465 in white color.

466 It can be found that all the profiles are roughly U shaped, similar to the true shape of the  
467 leaves.

#### 468 **4 Conclusion**

469 Plant reconstruction from stereo imagery is difficult due to the complexity of leaves  
470 which exhibit similar shape and intensity information. Hence the matching cost  
471 computation should be accurate to adequately represent the similarity between patches as  
472 the basis for the final disparity computation. SGM combined with MC-CNN has proved  
473 to outperform most previous algorithms; however, in practice it is extremely difficult to  
474 capture a large amount of high-quality training data. In this paper, a self-trained MC-  
475 CNN without the use of ground truth is tested to reconstruct the plant. Based on the dense  
476 matching results of MC-CNN pre-trained on the Middlebury data sets, a rigid left-right  
477 consistency check is applied to limit the outliers and the filtered results are utilized to  
478 further train the net. The reconstructed plant shows superior performance for the self-  
479 trained version than for the pre-trained one and the classic Census algorithm. Compared  
480 with MC-CNN further trained using the ground truth from LiDAR, the self-trained net  
481 behaves slightly worse in accuracy but better in reconstruction completeness. The self-

482 training strategy of MC-CNN is also applied to the stereo imagery of a natural forest tree  
483 under drought condition. The resultant disparity map is capable of showing the  
484 deformation of leaves, which highlights the possibility of the self-trained MC-CNN to  
485 monitor the tree health situation.

486 In future research, more approaches will be tested to capture the ground truth for outdoor  
487 experiments, for instance the structured light technique (Scharstein and Szeliski, 2003).  
488 Also the reconstruction of other more stable objects like buildings could be attempted.  
489 Furthermore, multi-viewed dense matching can be used to improve the self-training.  
490 Multiple images can in fact provide denser reconstruction results; meanwhile a  
491 consistency check among more than two images is able to further remove outliers which  
492 guarantees more reasonable training data. The self-training strategy of MC-CNN  
493 provides the possibility of detailed plant reconstruction and avoids the complexity of  
494 collecting ground truth especially in extreme situations.

## 495 **References**

496 Ahmed, O.S., S.E. Franklin, and M.A. Wulder, 2014. Integration of lidar and landsat data  
497 to estimate forest canopy cover in coastal British Columbia, *Photogrammetric*  
498 *Engineering & Remote Sensing*, 80(10): 953-961.

499 Bolles, R.C., H.H. Baker, and D.H. Marimont, 1987. Epipolar-plane image analysis: An  
500 approach to determining structure from motion, *International Journal of Computer*  
501 *Vision*, 1(1): 7-55.

502 Bromley, J., J.W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R.  
503 Shah, 1993. Signature verification using a siamese time delay neural network,  
504 International Journal of Pattern Recognition and Artificial Intelligence, 7(4): 669-688.

505 d'Angelo, P., 2016. Improving semi-global matching: cost aggregation and confidence  
506 measure, International Archives of the Photogrammetry, Remote Sensing and Spatial  
507 Information Sciences, 41(B1): 299-304.

508 d'Angelo, P., and P. Reinartz, 2011. Semiglobal matching results on the ISPRS stereo  
509 matching benchmark, Proceedings of ISPRS Workshop, Hannover, Germany, 38-  
510 4(W19): 79-84.

511 Freeman, M.P., D.A. Stow, and D.A. Roberts, 2016. Object-based image mapping of  
512 conifer tree mortality in San Diego county based on multitemporal aerial ortho-imagery,  
513 Photogrammetric Engineering & Remote Sensing, 82(7): 571-580.

514 Gatziolis, D., J.F. Lienard, A. Vogs, and N.S. Strigul, 2015. 3D tree dimensionality  
515 assessment using photogrammetry and small unmanned aerial vehicles, Public Library of  
516 Science ONE, 10(9): e0137765.

517 Geiger, A., P. Lenz, C. Stiller, and R. Urtasun, 2013. Vision meets robotics: The KITTI  
518 dataset, International Journal of Robotics Research, 32(11): 1231-1237.

519 Girardeau-Montaut, D., M. Roux, R. Marc, and G. Thibault, 2005. Change detection on  
520 points cloud data acquired with a ground laser scanner, International Archives of  
521 Photogrammetry, Remote Sensing and Spatial Information Sciences, 36(part 3): W19.

522 Hirschmüller, H., 2008. Stereo processing by semiglobal matching and mutual  
523 information, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):  
524 328-341.

525 Hirschmüller, H., 2011. Semi-global matching - motivation, developments and  
526 applications, *Proceedings of Photogrammetric Week*.

527 Hirschmüller, H., and D. Scharstein, 2009. Evaluation of stereo matching costs on images  
528 with radiometric differences, *IEEE Transactions on Pattern Analysis and Machine*  
529 *Intelligence*, 31(9): 1582-1599.

530 Joung, S., S. Kim, B. Ham, and K. Sohn, 2017. Unsupervised stereo matching using  
531 correspondence consistency, *IEEE International Conference on Image Processing*, pp.  
532 2518-2522.

533 Kankare, V., M. Holopainen, M. Vastaranta, E. Puttonen, X. Yu, J. Hyypä, M. Vaaja, H.  
534 Hyypä, and P. Alho, 2013. Individual tree biomass estimation using terrestrial laser  
535 scanning, *ISPRS Journal of Photogrammetry and Remote Sensing*, 75: 64-75.

536 Knöbelreiter, P., C. Vogel, and T. Pock, 2018. Self-supervised learning for stereo  
537 reconstruction on aerial images, *IEEE International Geoscience and Remote Sensing*  
538 *Symposium*, pp. 4383-4386.

539 Krizhevsky, A., I. Sutskever, and G.E. Hinton, 2012. Imagenet classification with deep  
540 convolutional neural networks, *Proceedings of Advances in Neural Information*  
541 *Processing Systems*, pp. 1097-1105.

542 Kuschik, G., P. d'Angelo, R. Qin, D. Poli, P. Reinartz, and D. Cremers, 2014. DSM  
543 accuracy evaluation for the ISPRS Commission I image matching benchmark,  
544 International Archives of the Photogrammetry, Remote Sensing and Spatial Information  
545 Sciences, 40(1): 195-200.

546 Lawrence, S., C.L. Giles, A.C. Tsoi, and A.D. Back, 1997. Face recognition: A  
547 convolutional neural network approach, IEEE Transactions on Neural Networks, 8(1):  
548 98-113.

549 LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner, 1998. Gradient-based learning applied  
550 to document recognition, Proceedings of the IEEE, 86(11): 2278-2324.

551 Levin, S.A., 1999. Fragile Dominion: Complexity and the Commons, Perseus Books,  
552 Cambridge, Massachusetts.

553 Luo, W., A.G. Schwing, and R. Urtasun, 2016. Efficient deep learning for stereo  
554 matching, Proceedings of IEEE Conference on Computer Vision and Pattern  
555 Recognition, Las Vegas, Nevada, USA, pp. 5695-5703.

556 Mei, X., X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, 2011. On building an  
557 accurate stereo matching system on graphics hardware, Proceedings of IEEE  
558 International Conference on Computer Vision Workshops, pp. 467-474.

559 Menze, M., and A. Geiger, 2015. Object scene flow for autonomous vehicles,  
560 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston,  
561 Massachusetts, USA, pp. 3061-3070.



562 Okutomi, M., and T. Kanade, 1993. A multiple-baseline stereo, IEEE Transactions on  
563 Pattern Analysis and Machine Intelligence, 15(4): 353-363.

564 Qin, R., X. Huang, A. Gruen, and G. Schmitt, 2015. Object-based 3-D building change  
565 detection on multitemporal stereo images, IEEE Journal of Selected Topics in Applied  
566 Earth Observations and Remote Sensing, 8(5): 2125-2137.

567 Rosu, A.M., M. Pierrot-Deseilligny, A. Delorme, R. Binet, and Y. Klinger, 2015.  
568 Measurement of ground displacement from optical satellite image correlation using the  
569 free open-source software MicMac, ISPRS Journal of Photogrammetry and Remote  
570 Sensing, 100: 48-59.

571 Scharstein, D., H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P.  
572 Westling, 2014. High-resolution stereo datasets with subpixel-accurate ground truth,  
573 German Conference on Pattern Recognition, Münster, Germany.

574 Scharstein, D., and C. Pal, 2007. Learning conditional random fields for stereo,  
575 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,  
576 Minneapolis, Minnesota, USA, pp. 1-8.

577 Scharstein, D., and R. Szeliski, 2002. A taxonomy and evaluation of dense two-frame  
578 stereo correspondence algorithms, International Journal of Computer Vision, 47(1-3): 7-  
579 42.

580 Scharstein, D., and R. Szeliski, 2003. High-accuracy stereo depth maps using structured  
581 light, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,  
582 Madison, Wisconsin, USA, 1: 195-202.

583 Segal, A., D. Haehnel, and S. Thrun, 2009. Generalized-icp, Proceedings of Robotics:  
584 Science and Systems.

585 Strigul, N., 2012. Individual-based models and scaling methods for ecological forestry:  
586 implications of tree phenotypic plasticity, Sustainable Forest Management-Current  
587 Research, pp. 359-384.

588 Tao, S., Q. Guo, S. Xu, Y. Su, Y. Li, and F. Wu, 2015. A geometric method for wood-  
589 leaf separation using terrestrial and simulated lidar data, Photogrammetric Engineering &  
590 Remote Sensing, 81(10): 767-776.

591 Tian, J., T. Schneider, C. Straub, F. Kugler, and P. Reinartz, 2017. Exploring digital  
592 surface models from nine different sensors for forest monitoring and change detection,  
593 Remote Sensing, 9(3): 287.

594 Wu, Z., D. Dye, J. Vogel, and B. Middleton, 2016. Estimating forest and woodland  
595 aboveground biomass using active and passive remote sensing, Photogrammetric  
596 Engineering & Remote Sensing, 82(4): 271-281.

597 Zbontar, J., and Y. LeCun, 2016. Stereo matching by training a convolutional neural  
598 network to compare image patches, Journal of Machine Learning Research, 17: 1-32.

- 599 Zhou, C., H. Zhang, X. Shen, and J. Jia, 2017. Unsupervised learning of stereo matching,  
600 Proceedings of IEEE International Conference on Computer Vision, 2(8): 1567-1575.



## Appendices

- B Xia, Y., d'Angelo, P., Tian, J., Fraundorfer, F. and Reinartz, P., 2020. Multi-label learning based semi-global matching forest. Remote Sensing, 12(7), p.1069.**

<https://www.mdpi.com/2072-4292/12/7/1069>



Article

# Multi-Label Learning based Semi-Global Matching Forest

Yuanxin Xia <sup>1,\*</sup> , Pablo d'Angelo <sup>1</sup> , Jiaojiao Tian <sup>1</sup>, Friedrich Fraundorfer <sup>1,2</sup> and Peter Reinartz <sup>1</sup>

<sup>1</sup> Department of Photogrammetry and Image Analysis, Remote Sensing Technology Institute, German Aerospace Center (DLR), 82234 Wessling, Germany; Pablo.Angelo@dlr.de; Jiaojiao.Tian@dlr.de; Peter.Reinartz@dlr.de

<sup>2</sup> Institute of Computer Graphics and Vision, Graz University of Technology (TU Graz), 8010 Graz, Austria; fraundorfer@icg.tugraz.at

\* Correspondence: Yuanxin.Xia@dlr.de; Tel.: +49-8153-2816-37

Version submitted to Remote Sens.

**Abstract:** Semi-Global Matching (SGM) approximates a 2D Markov Random Field (MRF) via multiple 1D scanline optimizations, which serves as a good trade-off between accuracy and efficiency in dense matching. Nevertheless, the performance is limited due to the simple summation of the aggregated costs from all 1D scanline optimizations for the final disparity estimation. SGM-Forest improves the performance of SGM by training a random forest to predict the best scanline according to each scanline's disparity proposal. The disparity estimated by the best scanline acts as reference to adaptively adopt close proposals for further post-processing. However, in many cases more than one scanline is capable of providing a good prediction. Training the random forest with only one scanline labeled may limit or even confuse the learning procedure when other scanlines can offer similar contributions. In this paper, we propose a multi-label classification strategy to further improve SGM-Forest. Each training sample is allowed to be described by multiple labels (or zero label) if more than one (or none) scanline gives a proper prediction. We test the proposed method on stereo matching datasets, from Middlebury, ETH3D, EuroSDR image matching benchmark, and the 2019 IEEE GRSS data fusion contest. The result indicates that under the framework of SGM-Forest, the multi-label strategy outperforms the single-label scheme consistently.

**Keywords:** Semi-Global Matching (SGM); random forests; scanline; multi-label classification; disparity; learning

## 1. Introduction

Dense matching recovers depth information from the dense correspondence between stereo imagery. Focusing on the similarity of patches to locate corresponding points is the most intuitive strategy (local stereo methods) and requires less computational effort [1]. The performance, however, is not competitive with methods considering spatial smoothness simultaneously (global stereo methods) at the cost of efficiency [1]. Semi-Global Matching (SGM) provides a good trade-off between accuracy and efficiency [2–5]. It regularizes disparity estimation by performing 1D Scanline Optimization (SO) [6] in multiple canonical directions, typically 8 or 16, and then summing up the corresponding energy functions. Thus, 2D SO is approximated and the disparity value corresponding to the minimum energy is selected based on the winner-take-all (WTA) strategy.

SGM has been applied in numerous fields, including building reconstruction, digital surface model generation, robot navigation, driver assistance etc. [7–9]. However, the energy summation from all scanlines and the corresponding WTA strategy are empirical steps without a theoretical background, which is essentially inadequate when different scanlines propose inconsistent solutions

32 [10]. Schönberger et al. [10] proposed SGM-Forest, which trained a random forest to predict a scanline  
33 with the best disparity proposal. Accordingly, a confidence value is obtained for each scanline, allowing  
34 for a confidence-based weighted average of the corresponding disparity prediction to refine the result.  
35 The algorithm is robust and performs steadily better than standard SGM in multiple stereo matching  
36 benchmark datasets [11–13].

37 However, in practice, there can be more than one scanline with good disparity prediction. It  
38 appears when multiple scanlines properly perceive the scene structure, therefore, are capable of  
39 predicting accurate disparity values simultaneously. For example, on a slanted plane extending  
40 horizontally, the two vertical scanlines (from bottom to top, and inversely), along which the slope is  
41 not explicitly expressed, should have better disparity estimation than the horizontal ones but achieve  
42 similar performance. Thus, the random forest gets confused when only a single best has to be selected.

43 In our project, we define a standard to determine good or bad scanlines, aiming at guiding the  
44 random forest to select as many good scanlines as possible for disparity prediction. The samples with  
45 zero scanline selection (all regarded as bad) are included for training, so that a more comprehensive  
46 prediction is obtained. The structure of the paper is as follows: Firstly, related work for improving SGM  
47 is described in Section 2. Afterwards, the standard SGM and SGM-Forest are recapped in Section 3,  
48 followed by our extension of SGM-Forest based on multi-label classification. In Section 4, the methods  
49 are tested on two close-range stereo matching datasets, Middlebury and ETH3D benchmarks [13–16],  
50 an airborne dataset, EuroSDR image matching benchmark [17], and a satellite dataset from the 2019  
51 IEEE GRSS data fusion contest [18,19]. The comparison is recorded between original SGM-Forest based  
52 on single-label classification (termed SGM-ForestS for the follow-up) and our proposed implementation  
53 based on multi-label classification (termed SGM-ForestM). The results indicate higher performance of  
54 the latter. Finally, the conclusion is drawn in Section 5 with an outlook for future work.

## 55 2. Related Work

56 Inspired by global stereo methods, SGM applies a matching cost measure (as data term) to check  
57 the photo consistency between potentially matching pixels, and designs a new strategy (as smoothness  
58 term) based on dynamic programming (DP) [20] to accomplish the spatial harmony among neighboring  
59 points. It is widely used for its good accuracy-efficiency balance and extendibility to various stereo  
60 systems, therefore, the algorithm has been optimizing for higher performance [10,21–28]. Regarding  
61 the data term, Ni et al. [21] combined three measures to calculate the matching cost for SGM, to  
62 keep robust in non-ideal radiometric conditions. Zbontar and LeCun [22] initiated a convolutional  
63 neural networks (CNN) based method to measure a similarity score between image patches, for a  
64 further process via SGM which achieved the state-of-the-art. Luo et al. [23] accelerated the cost volume  
65 generation using a faster Siamese network [29], and obtained good efficiency.

66 As for the smoothness term, Seki and Pollefeys [24] designed a CNN to adaptively penalize  
67 conflicting disparity prediction between neighboring pixels, to control the smoothness of the resultant  
68 disparity map. Their approach performed well in various situations, e.g. flat plane, slanted plane, and  
69 border. Scharstein et al. [25] enhanced SGM's ability for processing untextured or weakly-textured  
70 slanted area. They adjusted the penalty term based on the prior knowledge of the depth change which  
71 was obtained by precomputed surface orientation priors. Michael et al. [26] found out that the disparity  
72 map generated using a single scanline exhibited varying qualities as different canonical directions were  
73 adopted. Depending upon the global scene structure, the scanlines accounted for different significance  
74 for 2D SO approximation. Therefore, they proposed to assign a specific weight to each scanline for  
75 deriving a weighted summation before WTA. Poggi and Mattocchia [27] further extended this idea.  
76 According to the disparity map estimated by a single scanline, a feature vector was extracted for each  
77 pixel which indicated the statistical dispersion of disparity within the surrounding patch. The feature  
78 vector was then fed to a random forest to predict a confidence measure for the corresponding path,  
79 allowing for a weighted summation processing. Schönberger et al. [10] inferred that the upper bound  
80 of the matching accuracy can be approached by always selecting the best disparity proposal from all



81 the scanlines. They trained a random forest for the best scanline selection, which was more efficient via  
 82 simply using the disparity proposed by each scanline and the corresponding costs as input, instead of  
 83 handcrafting feature to feed random forest. Moreover, each scanline's estimation was better delivered.  
 84 Then based on the disparity predicted by the best scanline, other close disparity proposals were also  
 85 adopted for a weighted average according to the corresponding confidence measures. Thus, the higher  
 86 performance was achieved.

87 Recently, Zhang et al. [28] proposed a semi-global aggregation layer as a differentiable  
 88 approximation of SGM to accomplish an end-to-end network. Together with a local guided aggregation  
 89 layer for thin structures refinement, the network was capable of improving the dense matching  
 90 performance significantly for a challenging situation, e.g. occlusion, textureless area, etc.

### 91 3. Methodology

#### 92 3.1. SGM

93 As mentioned above, global stereo methods explicitly consider the smoothness demand in  
 94 addition to photo consistency. Accordingly, an energy function is defined for which a disparity map  
 95 should be optimized to properly balance the two claims and approach the energy minimization. This  
 96 optimization, however, cannot be achieved in 2D because the disparity determination for each pixel  
 97 will affect every other pixel under the smoothness assumption, which causes an np-complete problem  
 98 [1].

99 SGM starts from the image boundaries and aggregates the energy towards the target pixel along  
 100 a 1D path (scanline). Thus, for each pixel, the previous points are already considered during the  
 101 energy aggregation, which contributes to 1D smoothness. By summing up the aggregated energy from  
 102 multiple 1D paths, the disparity corresponding to the minimum energy is found based on the WTA  
 103 strategy and 2D smoothness is approximated. For a pixel located at image position  $p$  with a sampled  
 104 disparity  $d$  from the disparity space, the energy along the path traversing in direction  $r$  is defined as  
 105 follows:

$$L_r(p, d) = C(p, d) + \min ( L_r(p - r, d), L_r(p - r, d - 1) + P_1, \\ L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2 ), \quad (1)$$

106 in which  $L_r(p, d)$  represents the energy.  $C(p, d)$  is the photo inconsistency under the current parallax  
 107 and the rest of Equation (1) controls the smoothness by imposing a penalty term for a conflicting  
 108 disparity setting between  $p$  and its previous neighbor  $p - r$ . A small penalty  $P_1$  is applied for only 1  
 109 pixel difference, otherwise a larger penalty term  $P_2$  is used.

Considering several canonical directions  $r$ , the energy is summed up as follows:

$$S(p, d) = \sum_r L_r(p, d), \quad (2)$$

from which the disparity is computed according to the WTA strategy as:

$$d_p = \operatorname{argmin}_d S(p, d). \quad (3)$$

110 Thus, SGM is able to derive a suitable disparity for each pixel with spatial smoothness considered,  
 111 meanwhile spending reasonable runtime proportional to the reconstructed volume [3,4].

#### 112 3.2. SGM-Forests

113 SGM approximates energy minimization of a 2D Markov Random Field (MRF) via multiple  
 114 SOs, however, the summation of the aggregated cost along each scanline is not necessarily effective,  
 115 especially when different scanlines propose inconsistent estimation. In this case, an adaptive scanline

116 selection strategy is promising. Hence, Schönberger et al. [10] adopt a random forest to select the best  
 117 scanline based on a classification framework.

The input feature for the random forest is constructed in this way: Assuming a pixel at location  $p$  has a WTA winner  $d_p^{r'}$  along a certain path  $r'$  as:

$$d_p^{r'} = \operatorname{argmin}_d L_{r'}(p, d), \quad (4)$$

118 the corresponding costs  $K_p^r(r')$  on  $d_p^{r'}$  along all  $N$  scanlines are calculated, where  $N$  is the number of  
 119 directions considered.

$$K_p^r(r') = L_r(p, d_p^{r'}), \quad r = 1 \dots N. \quad (5)$$

$N + 1$  elements  $\{d_p^{r'}, K_p^{r=1}(r'), \dots, K_p^{r=N}(r')\}$  are obtained for the current scanline of  $r'$ . Thus for all the scanlines, a feature vector with a length of  $(N + 1) * N$  is acquired for  $p$  which is then fed into a random forest for the best scanline prediction  $r^*$  and a posterior probability  $\rho^*$ . In order to achieve a more robust estimation, the corresponding disparity  $d_p^{r^*}$  acts as a baseline to select other scanlines with a close prediction for a weighted averaging computation:

$$\hat{d}_p = \frac{\sum_r d_p^r * \rho_p^r}{\sum_r \rho_p^r}, \quad (6)$$

where  $d_p^{r'}$  is selected from a set of WTA winners differing  $d_p^{r^*}$  by less than  $\epsilon_d$ , and  $\rho_p^r$  is the corresponding posterior probability predicted by the random forest as:

$$D_p = \left\{ \left( d_p^r, \rho_p^r \right) \mid |d_p^r - d_p^{r^*}| < \epsilon_d \right\}, \quad r = 1 \dots N. \quad (7)$$

The sum of selected posterior probabilities  $\hat{\rho}_p = \sum_r \rho_p^r$  is the confidence measure of  $\hat{d}_p$ .  $\hat{\rho}_p$  is then used for a confidence-based median filtering within an adaptive local neighborhood  $\mathcal{N}_p$  centered around  $p$  as follows:

$$\bar{d}_p = \operatorname{median}(\hat{d}_q) \quad \text{and} \quad \bar{\rho}_p = \operatorname{median}(\hat{\rho}_q), \quad q \in \mathcal{N}_p \quad (8)$$

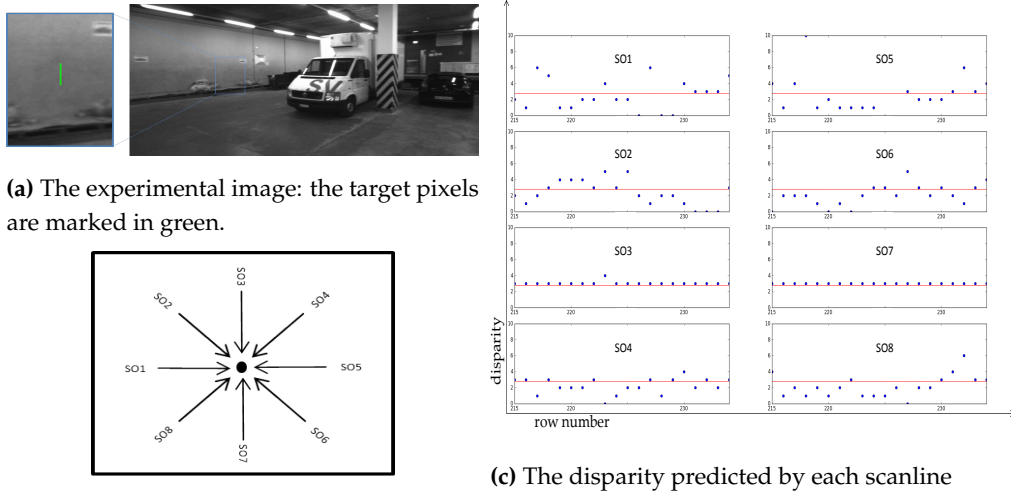
$$\mathcal{N}_p = \{q \mid \|q - p\| < \epsilon_p \wedge |I_q - I_p| < \epsilon_I \wedge \hat{\rho}_q > \epsilon_\rho\}, \quad (9)$$

120 where  $\|q - p\|$  measures the Euclidean distance between  $q$  and  $p$ .  $I$  is the image intensity.  $\epsilon_p$ ,  $\epsilon_I$  and  $\epsilon_\rho$   
 121 are the corresponding pre-defined thresholds.

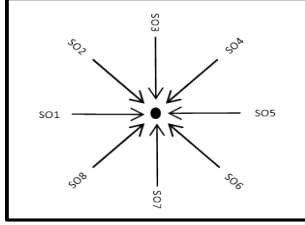
As for the training procedure, assuming the pixel at location  $p$  has the ground truth disparity available as  $d_p^{GT}$ , the label for this training sample is set as:

$$\tilde{r} = \operatorname{argmin}_r |d_p^r - d_p^{GT}|, \quad r = 1 \dots N. \quad (10)$$

122 However, this label assignment is problematic in some cases because multiple scanlines can  
 123 predict a disparity value very close to the ground truth. Figure 1 provides such an example. SO1 –  
 124 SO8 represent the disparity estimation through a single scanline in each of the 8 canonical directions.  
 125 Along the green line in (a), the disparities predicted by each scanline (defined in (b)) are shown in (c)  
 126 (blue dots), compared with the ground truth (red line). It is found that SO3 and SO7 accomplish better  
 127 solution than the other scanlines, however, barely differ from each other. In this case, both scanlines  
 128 should be selected.



(a) The experimental image: the target pixels are marked in green.



(b) The definition of each scanline: SO1 - SO8. in comparison with the ground truth.

(c) The disparity predicted by each scanline

**Figure 1.** The comparison between each single scanline’s disparity prediction and the ground truth, for pixels marked green in (a).

129 To further analyze this problem, we investigate Middlebury (2005 and 2006) [14,15] and ETH3D  
 130 [13] benchmark datasets, recording the percentage of pixels with multiple ( $\geq 2$ ) scanlines predicting  
 131 disparities close to the ground truth (differing by less than 1 pixel) in Table 1. The percentage of pixels  
 132 with at least one well-predicting scanline is appended below, which indicates the theoretical upper  
 133 bound of the performance, for SGM based on the random forest to select scanlines. Census [30] is used  
 134 here as the matching cost. It is found that, for most pixels (75.52% in Middlebury, 81.69% in ETH3D),  
 135 more than one scanline potentially achieves a good disparity estimation.

**Table 1.** The percentage of pixels with more than one scanline achieving good prediction for Middlebury and ETH3D benchmarks.

	Middlebury	ETH3D
Good scanline $\geq 2$	75.52%	81.69%
Good scanline $\geq 1$	83.83%	90.65%

136 Although SGM-ForestS further refines the disparity prediction by considering other scanlines  
 137 with close proposals, it’s supposed to be more reasonable if the random forest learns to select all the  
 138 proper scanlines directly in training. Therefore, we adjust the scanline selection based on a multi-label  
 139 classification strategy and propose SGM-ForestM.

### 140 3.3. SGM-ForestM

#### 141 3.3.1. Multi-Label Classification

142 Traditional pattern recognition focuses on classification tasks with each class defined mutually  
 143 exclusive [31]. For some scenarios, however, there are samples with multiple properties among  
 144 different classes, e.g. a movie categorized into comedy and action film, which may confuse the  
 145 classifier during training. In order to handle these samples properly, the first issue is label assignment.  
 146 The most intuitive solution is to label a sample by the class it most likely belongs to. This strategy,  
 147 nevertheless, is ambiguous and may result in a subjective judgment. An alternative is to neglect the  
 148 samples related to multiple classes and concentrate only on the rest with a distinct definition. Yet, the  
 149 classifier trained in this way is not able to deal with multi-label samples in the test period.

150 The two schemes above simply ignore the multi-label attribute of the samples and still treat the  
 151 problem based on a single label classification strategy, therefore, the performance is limited. To cover

152 all the corresponding labels of each sample, a new option is to define some ‘composite’ classes, of  
 153 which each class includes a certain combination of base classes, e.g. ‘building + plant’ from ‘building’  
 154 and ‘plant’. Then each composite class is allocated with a new label number above the original  
 155 range for training. The samples categorized as composite classes, however, are normally too sparse  
 156 to train a well-behaved classifier [32]. Hence, Boutell et al. [32] propose a ‘cross-training’ strategy  
 157 which simultaneously trains multiple binary classifiers. Each binary classifier aims at determining  
 158 the existence of a certain base class, and regards the corresponding multi-label samples as positive  
 159 examples for training. For example, the samples of ‘building + plant’ are regarded as ‘building’ and  
 160 ‘plant’, respectively, when training the ‘building classifier’ and ‘plant classifier’. Thus, all the labels of  
 161 each training sample are considered, meanwhile the training data are explored more effectively. In  
 162 this paper, the ‘cross-training’ scheme is applied for training the random forest based on a multi-label  
 163 classification strategy, in order to process pixels with more than one scanline predicting appropriate  
 164 disparities. With the cost aggregation applied along a certain path as equation (4), if the estimated  
 165 disparity is close to the ground truth, the corresponding pixel should be regarded as a positive sample  
 166 for training the binary classifier of the path. Regarding the pixels marked green in Figure 1 (a) as an  
 167 example, the label should be set as positive for the classifier of SO3 and SO7, and as negative for the  
 168 others. The multi-label strategy is appropriate for classification when overlap exists among different  
 169 categories. The label assignment is more reasonable for non-mutually exclusive classes, in which  
 170 one sample can be essentially related to multiple labels. It applies not only to computer vision, e.g.  
 171 semantic scene classification, but also in many other fields including document analysis (e.g. text  
 172 categorization), medicine (e.g. disease diagnosis), etc. [32–36].

### 173 3.3.2. Theoretical Background and Implementation Details

The feature for our SGM-ForestM is extracted in the same way as SGM-ForestS described in Section 3.2., however, the label setting is adjusted to satisfy our multi-label concept. Instead of selecting the best scanline with the closest prediction to the ground truth as Equation (10), we define a threshold  $\epsilon_{dso}$  to extract all the promising scanlines as:

$$\mathcal{R}_p = \left\{ r \mid |d_p^r - d_p^{GT}| < \epsilon_{dso} \right\}, \quad r = 1 \dots N. \quad (11)$$

174 Thus, the pixel  $p$  is a positive example when training the binary classifiers of all the corresponding  
 175 scanlines contained by  $\mathcal{R}_p$ . Otherwise,  $p$  is regarded as negative.

176 Afterwards in the test period, the trained random forest gives  $N$  predictions and  $N$  probabilities  
 177 for each pixel, indicating which scanlines should be regarded as good disparity proposals (with the  
 178 corresponding probability,  $\rho_p^r$ , larger than 0.5). It should be noted that a probability value is calculated  
 179 exclusively for a certain scanline with no dependency on the others. Unlike the single label classifier  
 180 that the probabilities for all classes should be sum-to-one, the multi-label classifier is not restricted to  
 181 follow the rule.

With multiple (or zero) scanlines proposed by the random forest, the one with the highest probability,  $r^*$ , is considered as a baseline to refine the disparity estimation as given in Equation (12) and (13) below:

$$\hat{d}_p = \frac{\sum d_p^r * \rho_p^r}{\sum \rho_p^r}, \quad d_p^r, \rho_p^r \in D_p \quad (12)$$

$$D_p = \left\{ (d_p^r, \rho_p^r, r) \mid |d_p^r - d_p^{r^*}| < \epsilon_d \right\}, \quad r = 1 \dots N. \quad (13)$$

Here  $D_p$  is constructed via selecting disparity estimation close to  $d_p^{r^*}$  from the WTA winners as SGM-ForestS. Thus, we limit the influence from the outliers, and ensure that one disparity value is available for further processing. As Equation (6) and (7), we refer to SGM-ForestS’s strategy to consider scanlines with close disparity proposals, however, it should be pointed out that the disparity

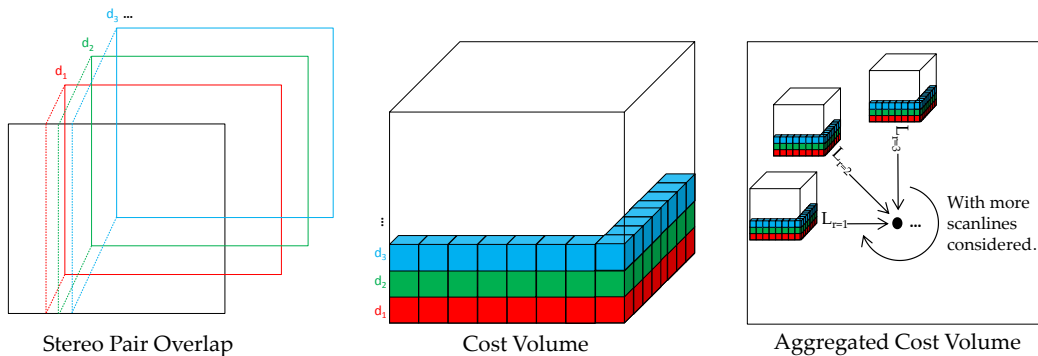
refinement of our SGM-ForestM is based on more reasonable prediction,  $r^*$ , owing to multi-label classification. In addition, the confidence measure should be adjusted accordingly as:

$$\hat{\rho}_p = \frac{\sum_{r \in D_p} \rho_p^r}{\sum_{r=1}^N \rho_p^r}, \quad (14)$$

182 in which the nominator is still the sum of probabilities for selected scanlines as SGM-ForestS. The  
 183 denominator, on the other hand, is the sum of all scanlines' probabilities in order to confine the  
 184 confidence in the range of [0, 1]. Following SGM-ForestS, a confidence-based median filter is exploited  
 185 as well. We test our proposed algorithm on multiple datasets. The results indicate superior performance  
 186 of SGM-ForestM, as shown in Section 4.

### 187 3.4. Efficiency and Memory Usage

188 SGM approximates global energy function by summing up the aggregated costs along multiple  
 189 1D paths. The number of paths is determined according to application demands, hardware constraints  
 190 or quality requirements [37]. With more paths considered, e.g. 8 or 16, better results are obtained  
 191 incurring reduced streaking artifacts, however, at the expense of high computational complexity [4,37].  
 192 As shown in Figure 2, SGM-Forest requires storing the full aggregated cost volumes for all aggregation  
 193 directions, leading to increased memory usage over standard SGM. Thus, resource efficient solutions  
 194 and high resolution data processing are hampered as the number of paths increases.



**Figure 2.** Stereo pair, cost cube and the corresponding aggregated cost cube in SGM.

195 Hence, we test different implementations of SGM, SGM-ForestS, and SGM-ForestM, as indicated  
 196 in Section 4, by varying the number of scanlines considered for further processing. We aim at  
 197 observing how the SGM-Forest algorithms are influenced, when fewer scanline proposals are applied.  
 198 A particularly interesting case is the configuration with 5 scanlines starting from left, top-left, top,  
 199 top-right and right, as this allows a memory efficient top down sweep implementation which only  
 200 requires storing two lines of the  $C$  and  $L_r$  volumes, greatly reducing the amount of required memory.  
 201 This enables the processing of very large stereo pairs with sizes of 200 to 2000 Megapixels, as typically  
 202 occurring in aerial and satellite data. Thus, the potential of SGM-Forest for efficient systems can be  
 203 explored, such as real-time designs in CPU and GPU systems, or embedded modules on e.g. embedded  
 204 multi-core architectures and Field-Programmable Gate Arrays (FPGAs) [37–41].

## 205 4. Experiments

206 In order to show the benefits of our multi-label classification strategy for training the random  
 207 forest, we refer to [10] and apply the same implementation for both SGM-ForestS and SGM-ForestM.  
 208 All the processing details are controlled, including the matching cost computation, SGM setting, etc., for  
 209 the sake of an unbiased comparison. As for the matching cost, both Census [30] and MC-CNN-acrt [22]  
 210 are tested. Census, as a non-learning based method, performs generally well in many stereo algorithms,

211 while MC-CNN-acrt represents the current state of the art for CNN based matching cost calculation.  
 212 Therefore, the two algorithms are appropriate for our SGM and SGM-Forest implementation. With  
 213 regard to Census, a  $7 \times 7$  window size is set. For MC-CNN-acrt, the original network architecture is  
 214 used: The number of convolutional layers is 5, with 112 feature maps and  $3 \times 3$  kernel size for each;  
 215 The number of fully-connected layers is 3, with the corresponding number of units as 384.

216 Regarding SGM, the matching cost is scaled to be in the range of  $[0, 1023]$ , and  $P_1$  and  $P_2$  are set as  
 217 400 and 700, respectively, to compute  $L_r(p, d)$ . We perform SO along 8 canonical directions ( $N = 8$  with  
 218 2 horizontal, 2 vertical, and 4 diagonal scanlines, as Figure 1b) in order to generate input proposals to  
 219 train the random forest for SGM-ForestS and SGM-ForestM. The 8 scanlines are also used to conduct  
 220 a standard SGM as a baseline comparison. In addition, as described in Section 3.4., we adjust the  
 221 implementation of SGM, SGM-ForestS, SGM-ForestM by applying 5 SOs instead of 8, in order to  
 222 check the influence when using fewer scanlines. 2 horizontal, 1 vertical (pointing downwards), and 2  
 223 diagonal (pointing downwards) scanlines are included, which accomplish a top-down sweep of the  
 224 scene to enable single-pass algorithms and consume less aggregation buffer [37]. As for the 8-scanlines  
 225 version, both Census and MC-CNN-acrt are employed as matching cost, for a general comparison  
 226 among the three SGM related algorithms. As the 5 scanlines version targets fast implementation, it is  
 227 only tested using the faster Census data term.

228 Considering SGM-Forest, we exploit the same parameter setting as proposed in [10]. For both  
 229 SGM-Forest versions, the same forest structure is adopted comprising 128 trees with the maximum  
 230 depth of each as 25, based on *Gini impurity* to measure the split quality. Before feeding to the  
 231 random forest, we preprocess the disparity proposals  $d_p^r$  via normalizing to relative values for feature  
 232 vectors construction, in order to generalize across different datasets. The disparity estimates are then  
 233 denormalized to absolute values for further confidence based filtering.  $\epsilon_d, \epsilon_p, \epsilon_l$ , and  $\epsilon_\rho$  are respectively  
 234 set as 2, 5, 10, and 0.1, which are determined according to parameter grid search and 3-fold cross  
 235 validation based on Middlebury 2014 training datasets [10].  $\epsilon_{dso}$  is set as 1 pixel in SGM-ForestM. All  
 236 our implementations are based on Python and C.

#### 237 4.1. Close-Range Datasets Experiments

238 The experiment contains the usage of two benchmark datasets, Middlebury and ETH3D, which  
 239 supply a certain number of stereo pairs with ground truth disparity maps available. We rigidly split  
 240 the provided datasets into non-overlapping training and validation sets (as shown below), in order to  
 241 train our proposed algorithm and test the performance according to the validation accuracy. From the  
 242 manually split training set, 500K pixels are randomly selected for training the random forest, while  
 243 all the pixels are used to train MC-CNN-acrt. As for the Middlebury benchmark, the training set is  
 244 acquired from 2005 and 2006 scenes, while 2014 scenes provide the validation set, as shown in Table 2.  
 245 Each dataset from Middlebury 2005 and 2006 consists of 7 views under 3 illumination and 3 exposure  
 246 conditions (63 images in total). Ground truth disparity maps are provided for view-2 and view-6.  
 247 We regard the former as the master epipolar frame, and randomly select illumination and exposure  
 248 condition for two images to construct stereo pairs for further processing.

**Table 2.** Train/validation splits for Middlebury benchmark.

	<b>Train</b>	<b>Validation</b>
<b>Middlebury 2005</b>	Books	Adirondack
	Dolls	ArtL
	Laundry	Jadeplant
	Moebius	Motorcycle
	Reindeer	MotorcycleE
	Aloe	Piano
	Baby1	PianoL
	Baby2	Pipes
	Baby3	Playroom
	Bowling1	Playtable
	Bowling2	PlaytableP
	Cloth1	Recycle
	Cloth2	Shelves
<b>Middlebury 2006</b>	Cloth3	Teddy
	Cloth4	Vintage
	Flowerpots	
	Lampshade1	
	Lampshade2	
	Midd1	
	Midd2	
	Monopoly	
	Plastic	
	Rocks1	
	Rocks2	

249 ETH3D stereo benchmark contains various indoor and outdoor views with ground truth extracted  
250 using a high-precision laser scanner. The images are acquired using a Digital Single-Lens Reflex (DSLR)  
251 camera synchronized with a multi-camera rig capturing varying field-of-views. The benchmark  
252 provides high-resolution multi-view stereo imagery, low-resolution many-view stereo on video data,  
253 and low-resolution two-view stereo images that are used in this paper. There are 27 frames with  
254 ground truth for training and 20 for test. We exploit the former for train/validation splits, as shown in  
255 Table 3. For some scenes, the data include two different sizes. Both focus on the same target, however,  
256 with one contained in the field of view from the other (e.g. `delivery_area_1s` and `delivery_area_1l`).  
257 Therefore, we manually divide the datasets for training and validation, in order to avoid images taken  
258 for the same scene appearing in both splits.

**Table 3.** Train/validation splits for ETH3D benchmark.

<b>Train</b>	<b>Validation</b>
<code>delivery_area_1s</code>	<code>delivery_area_2s</code>
<code>delivery_area_1l</code>	<code>delivery_area_2l</code>
<code>delivery_area_3s</code>	<code>electro_1s</code>
<code>delivery_area_3l</code>	<code>electro_1l</code>
<code>electro_2s</code>	<code>facade_1s</code>
<code>electro_2l</code>	<code>forest_2s</code>
<code>electro_3s</code>	<code>playground_2s</code>
<code>electro_3l</code>	<code>playground_2l</code>
<code>forest_1s</code>	<code>playground_3s</code>
<code>playground_1s</code>	<code>playground_3l</code>
<code>playground_1l</code>	<code>terrace_1s</code>
<code>terrains_2s</code>	<code>terrace_2s</code>
<code>terrains_2l</code>	<code>terrains_1s</code>
	<code>terrains_1l</code>

#### 259 4.1.1. Accuracy Evaluation

260 We evaluate the validation accuracy of SGM, SGM-ForestS, and our SGM-ForestM by comparing  
 261 the generated disparity map with ground truth. Only the non-occluded pixels observed by both scenes  
 262 are considered. The percentage of pixels with an estimation error less than 0.5, 1, 2, and 4 pixels,  
 263 respectively, are calculated as indicated by Table 4 and 5. It should be noticed that, in Table 4, a suffix  
 264 of '-5dirs' or '-8dirs' is appended at the end of each algorithm to differentiate SGM, SGM-ForestS, and  
 265 SGM-ForestM implemented using 5 or 8 scanlines, respectively. For the follow-up in this paper, unless  
 266 mentioned explicitly, all the SGM related terms without a suffix represent the implementation based  
 267 on 8 scanlines.

**Table 4.** The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: Census; '-5dirs' for 5 scanlines version, '-8dirs' for 8 scanlines version).

	Middlebury				ETH3D			
	0.5pix	1pix	2pix	4pix	0.5pix	1pix	2pix	4pix
SGM-5dirs	55.89%	67.60%	73.34%	77.48%	67.60%	79.18%	85.80%	90.33%
SGM-ForestS-5dirs	55.97%	68.71%	74.44%	78.37%	70.87%	82.97%	89.93%	95.03%
SGM-ForestM-5dirs	56.88%	70.30%	76.44%	80.37%	71.83%	85.00%	91.69%	95.96%
SGM-8dirs	58.92%	69.47%	74.87%	78.84%	70.14%	80.88%	87.02%	91.27%
SGM-ForestS-8dirs	59.38%	70.71%	76.33%	80.41%	72.87%	83.91%	90.55%	95.44%
SGM-ForestM-8dirs	<b>60.38%</b>	<b>72.16%</b>	<b>78.00%</b>	<b>82.19%</b>	<b>74.04%</b>	<b>86.20%</b>	<b>92.48%</b>	<b>96.37%</b>

**Table 5.** The validation accuracy of SGM, SGM-ForestS, and SGM-ForestM on Middlebury and ETH3D datasets, respectively (Matching cost: MC-CNN-acrt).

	Middlebury				ETH3D			
	0.5pix	1pix	2pix	4pix	0.5pix	1pix	2pix	4pix
SGM	69.35%	79.35%	83.37%	86.07%	72.39%	83.29%	89.48%	94.18%
SGM-ForestS	<b>70.01%</b>	<b>81.34%</b>	<b>85.71%</b>	<b>88.64%</b>	74.25%	86.03%	92.04%	96.30%
SGM-ForestM	69.92%	81.32%	85.56%	88.28%	<b>74.61%</b>	<b>86.47%</b>	<b>92.36%</b>	<b>96.44%</b>

268 As for 8 scanlines implementation, it is found that the two SGM-Forest implementations perform  
 269 steadily better than the standard SGM, in both benchmarks considering different estimation errors  
 270 as the upper limit. With MC-CNN-acrt as matching cost, the results on Middlebury datasets report  
 271 slightly worse performance of SGM-ForestM (about 0.1% difference) than SGM-ForestS. However, a  
 272 stable improvement is achieved by SGM-ForestM in all the other cases (the results on Middlebury  
 273 and ETH3D using Census as matching cost, on ETH3D using MC-CNN-acrt as matching cost), which  
 274 indicates the significance of applying the multi-label classification strategy to train the random forest.

275 For 5 scanlines version, the performance of all the algorithms decreases as expected due to the  
 276 information loss from fewer scanlines. Nevertheless, SGM-ForestM is still better than SGM-ForestS,  
 277 and both of them are superior to the standard SGM. It is worth to mention that, SGM-ForestS-5dirs  
 278 and SGM-ForestM-5dirs achieve even better results than SGM-8dirs on ETH3D datasets, which  
 279 indicates the potential to embed SGM-Forest into efficient stereo systems. On Middlebury datasets,  
 280 SGM-ForestS-5dirs is not able to keep its superiority to SGM-8dirs. However, it's good to find that  
 281 SGM-ForestM-5dirs remains to be better than the standard SGM using 8 scanlines (except for 0.5 pixel  
 282 error) and proves its robustness.

283 MC-CNN is a "data-hungry" method, which requires a large amount of training data to achieve  
 284 high performance [22]. The training of the random forest in SGM-Forest, nevertheless, relies on much  
 285 less data (500K pixels used in this paper and [10]). With Census as matching cost, SGM-ForestM  
 286 consistently outperforms SGM and SGM-ForestS in all settings, which further indicates the potential



287 of the algorithm, especially when the amount of data is too limited for training a well-performing  
 288 MC-CNN.

289 In order to apply an unbiased demonstration for our multi-label classification strategy, below in  
 290 Table 6, we exhibit the official results of the ETH3D benchmark by evaluating our SGM-ForestM on  
 291 the test datasets. As the proposed method focuses on the refinement of SGM itself, we simply use  
 292 Census for a quick test. The random forest is also trained on 500K pixels, with 8 scanlines for disparity  
 293 proposals.

**Table 6.** The benchmark results of SGM-ForestM on ETH3D datasets (Matching cost: Census).

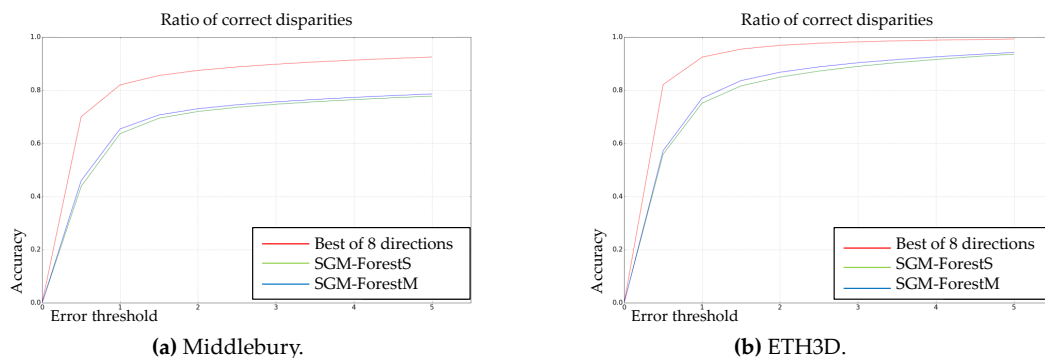
	SGM-ForestM			
	0.5pix	1pix	2pix	4pix
non-occluded	76.28%	83.01%	87.44%	91.11%
all	74.79%	81.39%	85.75%	89.42%

294 The accuracy for ‘non-occluded pixels’ is consistent with the numbers obtained in Table 4  
 295 (SGM-ForestM-8dirs), however, compared with other algorithms, our result is not competitive. The  
 296 reason includes that, we execute no post-processing, e.g. left-right consistency check, interpolation,  
 297 etc., and Census is used for calculating matching cost instead of a well-trained MC-CNN. It should  
 298 be noted that the main goal of this paper is to improve SGM and SGM-ForestS further, therefore, the  
 299 whole processing pipeline is not fully considered.

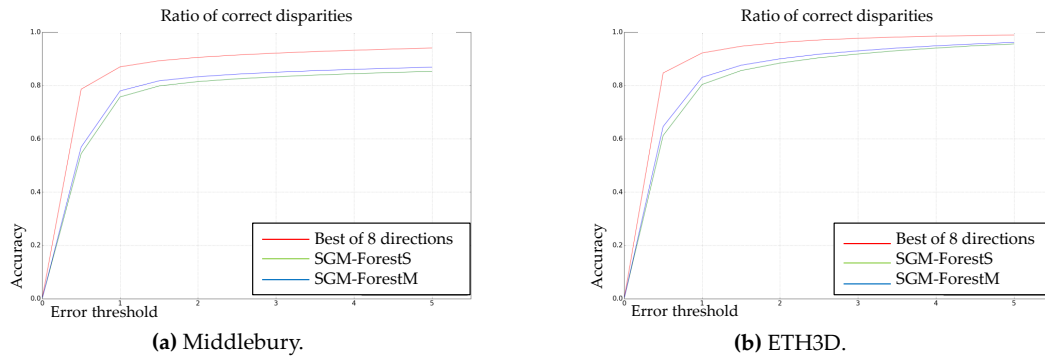
#### 300 4.1.2. Random Forest Prediction

301 In addition, we analyze the quality of  $r^*$  (see Section 3), which is the direct prediction of the  
 302 random forest and the reference for further confidence based processing. Adaptive scanline selection  
 303 based on a classification strategy is the core concept of SGM-Forest that is superior to the scanline  
 304 average of the standard SGM. Hence,  $r^*$  and the corresponding  $d_p^*$  are necessary for further comparison  
 305 between SGM-ForestS and SGM-ForestM.

306 In Figure 3 and 4, the error plots are displayed for SGM-ForestS, SGM-ForestM, and the upper  
 307 bound of SO if the best scanline can always be selected from 8 alternatives. At here, it should be  
 308 noted that the disparity prediction of the random forest ( $d_p^*$ ) is directly compared to the ground  
 309 truth for calculating the ratio of correct disparity estimation (y-axis), considering different estimation  
 310 errors allowed (x-axis). We still test two matching cost algorithms (Census and MC-CNN-acrt) on two  
 311 benchmark datasets (Middlebury and ETH3D).



**Figure 3.** Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: Census).

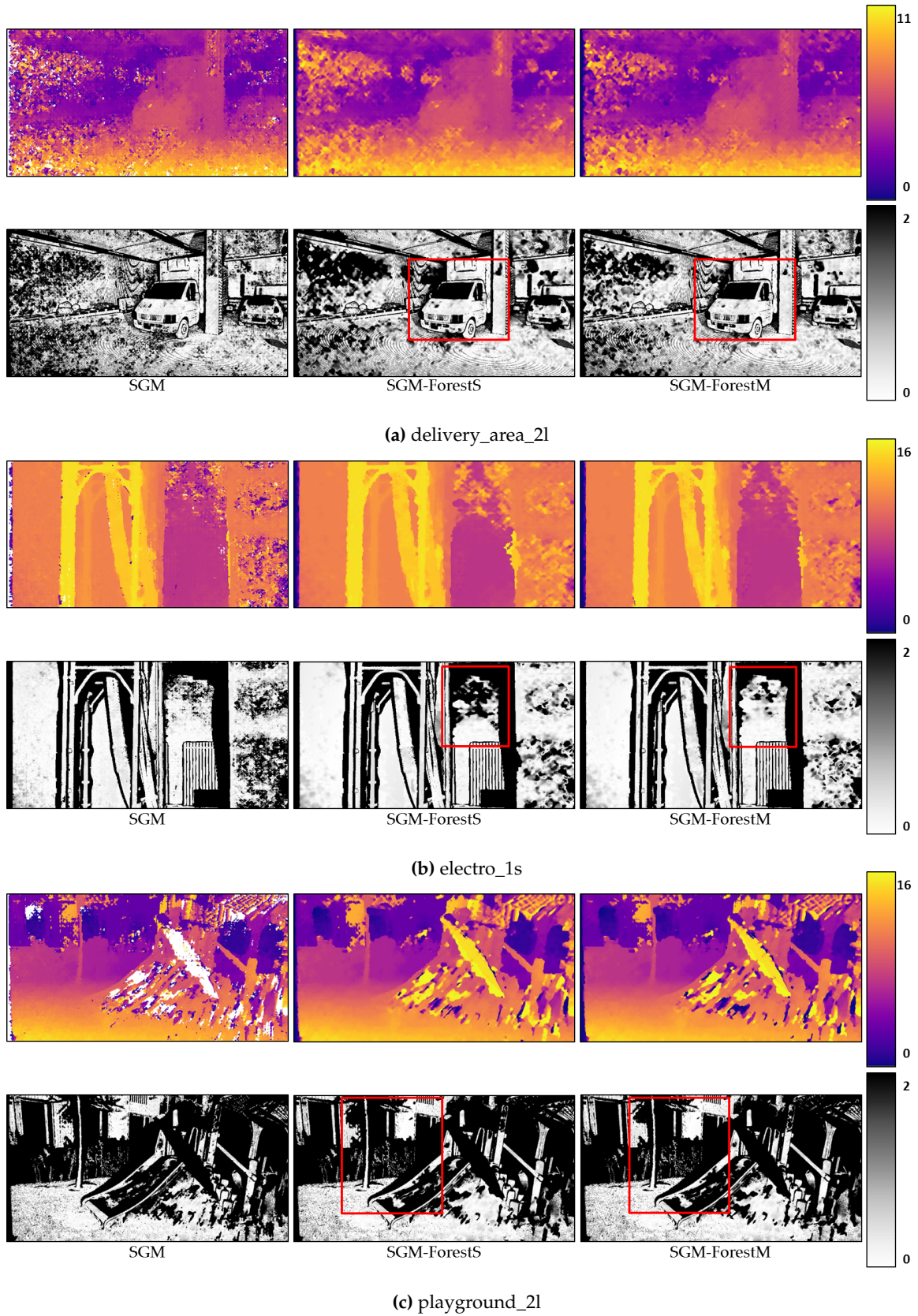


**Figure 4.** Error plots for SGM-ForestS, SGM-ForestM, and the upper bound of SO (Matching Cost: MC-CNN-acrt).

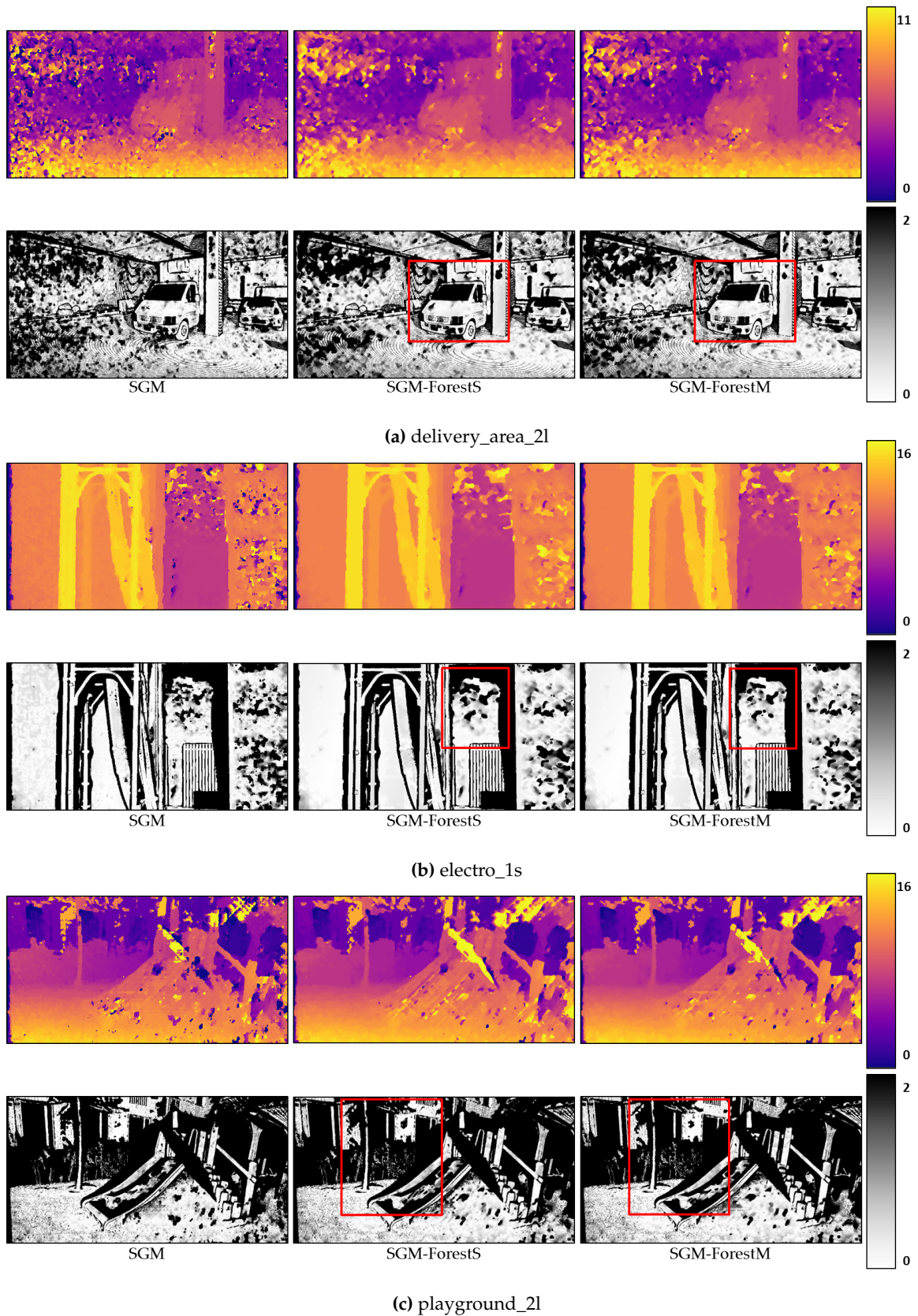
312 The figures above show that both SGM-Forest implementations achieve good performance to  
 313 approach the best SO, which demonstrates the feasibility of scanline selection based on a classification  
 314 framework. In addition, SGM-ForestM is superior to SGM-ForestS in all cases. The results indicate  
 315 that SGM-ForestM is essentially better at scanline prediction and capable of deriving preferable initial  
 316 disparity values for further processing.

#### 317 4.1.3. Qualitative Results

318 In this section, we select several stereo pairs from ETH3D to show the disparity maps generated  
 319 based on SGM, SGM-ForestS, and SGM-ForestM, respectively. The corresponding error maps are  
 320 displayed below. Regarding '2 pixel' as the upper bound, all the pixels with an error above the bound  
 321 are colored black, while the rest are colored uniformly according to the error as indicated by the color  
 322 bar. We apply Census and MC-CNN-acrt to calculate the matching cost, respectively, and the results  
 323 are displayed in Figure 5 and 6.



**Figure 5.** The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: Census).



**Figure 6.** The disparity maps and the corresponding error maps. From left to right, the results of SGM, SGM-ForestS, and SGM-ForestM are displayed, respectively (Matching cost: MC-CNN-acrt).

324 In each subfigure, the disparity map and the error map for SGM, SGM-ForestS, and SGM-ForestM,  
 325 respectively, are displayed from left to right, with a color bar at the end. The red rectangles marked in

326 the error maps represent the main difference of the result between SGM-ForestS and SGM-ForestM.  
 327 It is found that the disparity maps generated by the two SGM-Forest implementations are smoother  
 328 than SGM. Moreover, according to the error map, SGM-ForestM suffers fewer errors compared with  
 329 SGM-ForestS. Especially for the ill-posed regions (e.g. textureless areas, reflective surfaces, etc.),  
 330 SGM-ForestM performs better as highlighted by the red rectangles.

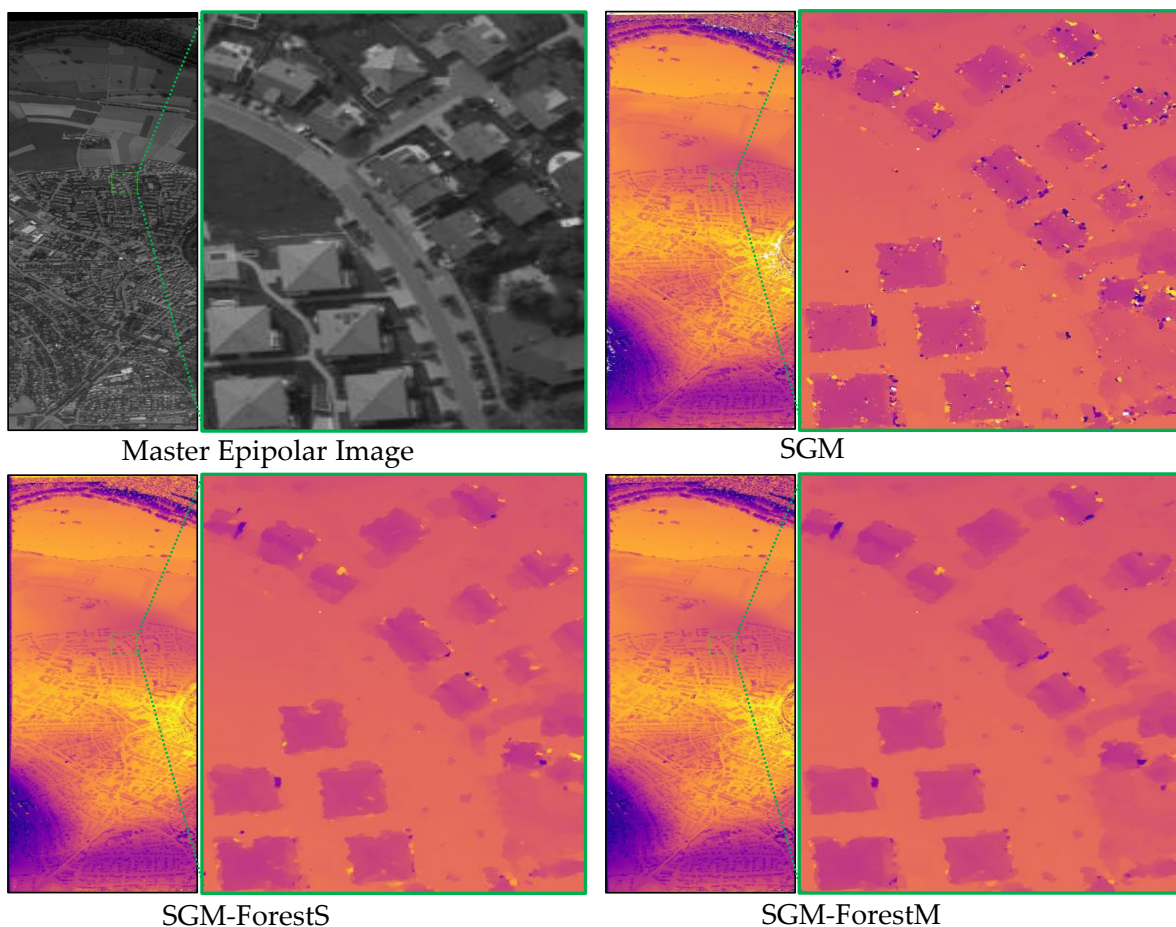
#### 331 4.2. Airborne and Satellite Datasets

332 Besides the close-range images, we also test the proposed algorithm on airborne data, the aerial  
 333 image matching benchmark from EuroSDR, and on satellite data, from the pairwise semantic stereo  
 334 challenge (Track 2) in the 2019 IEEE GRSS data fusion contest [19].

##### 335 4.2.1. Airborne Dataset Experiment

336 The aerial image matching benchmark project is motivated by the development of matching  
 337 algorithms and the improved quality of the elevation data obtained by advanced airborne cameras.  
 338 Based on the benchmark datasets and the corresponding evaluation platform, the potential of the  
 339 ongoing photogrammetric software is assessed by comparing their generated 3D products, including  
 340 point clouds, digital surface models (DSM), etc.

341 The nadir airborne datasets, Vaihingen/Enz with moderate ground sampling distance (20 cm)  
 342 and overlap (63% in flight and 62% cross flight), are used in this paper. We randomly select a stereo  
 343 pair and apply SGM, SGM-ForestS, and SGM-ForestM to generate a disparity map, respectively. The  
 344 master epipolar image and the corresponding result of each algorithm are displayed in Figure 7, with  
 345 an area highlighted by a green rectangle to compare details.



**Figure 7.** Stereo matching results on EuroSDR benchmark datasets (Vaihingen/Enz).

346 According to the results above, it is still found that the two SGM-Forest implementations generate  
347 a smoother disparity map than the standard SGM. Within the highlighted region, SGM-ForestM suffers  
348 less noise than SGM-ForestS, which further demonstrates the superiority of the former.

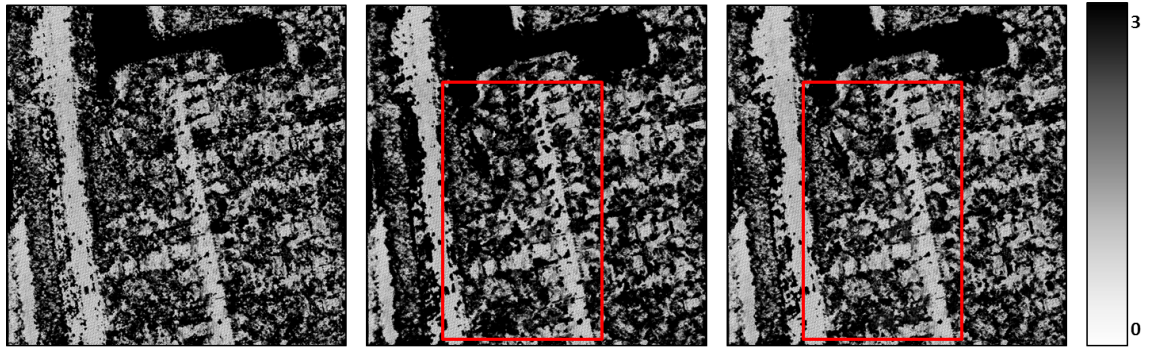
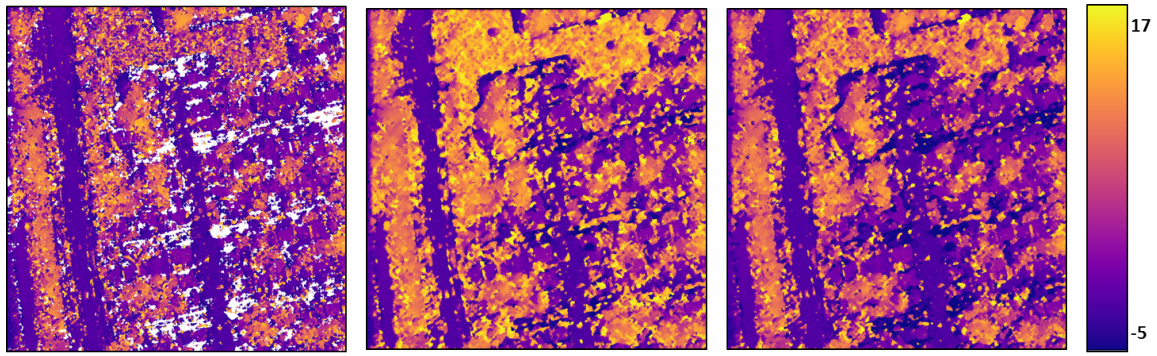
#### 349 4.2.2. Satellite Dataset Experiment

350 The 2019 IEEE GRSS data fusion contest provides the `grss_dfc_2019` dataset [42], a subset of the  
351 Urban Semantic 3D (US3D) [18] data, including multi-view, multi-band satellite images and ground  
352 truth geometric and semantic labels. Several tasks are designed to reconstruct both a 3D geometric  
353 model and a segmentation of semantic classes for urban scenes, aiming at further supporting the  
354 research in stereo and semantic 3D reconstruction using machine intelligence and deep learning.

355 The contest data are captured by WorldView-3 satellite including RGB and 8-band visible and  
356 near infrared (VNIR) multi-spectral images, with ground sampling distance as approximately 35 cm.  
357 26 images are collected between 2014 and 2016 over Jacksonville, Florida, and 43 images are collected  
358 between 2014 and 2015 over Omaha, Nebraska, United States. In our experiment, epipolar rectified  
359 stereo pairs from challenge track 2 are used, with pairwise ground truth disparity images generated  
360 using airborne LiDAR data. For evaluation, we only consider the reconstructed stereo geometry,  
361 ignoring the semantics information.

362 We apply SGM, SGM-ForestS, and SGM-ForestM, on 150 stereo pairs randomly selected from  
363 Jacksonville data. Due to the data inconsistency between the stereo images and LiDAR point clouds,  
364 the random forest is trained on ETH3D datasets for SGM-ForestS and SGM-ForestM. Thus, the  
365 robustness of the proposed algorithm is also tested when different data sources are used for training  
366 and validation.

367 When using 3 pixels as the upper limit of the allowed error, the validation accuracy for SGM,  
368 SGM-ForestS, and SGM-ForestM are 66.06%, 61.36%, and 67.18%, respectively. With different datasets  
369 to train the random forest, the performance of SGM-ForestS is limited and even surpassed by original  
370 SGM. The reason is the poor inference of the random forest when data different from the training sets  
371 are fed as input. However, SGM-ForestM is capable of providing more reliable scanline prediction,  
372 which is consistent with our demonstration in Figure 3 and 4. Therefore, it performs the best. Some  
373 visualization results are displayed in Figure 8.

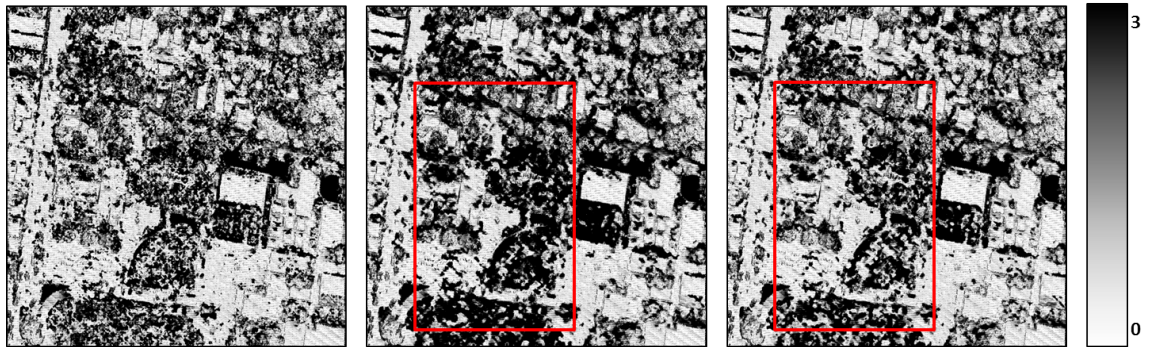
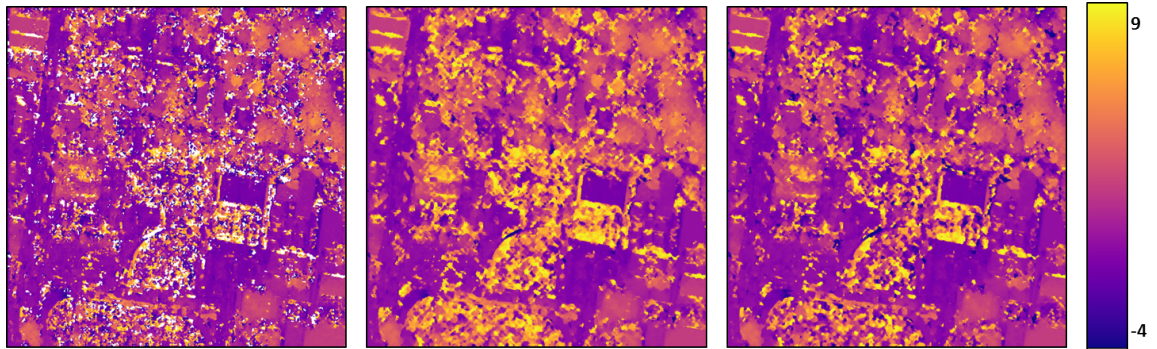


SGM

SGM-ForestS

SGM-ForestM

(1) JAX\_004\_009\_007

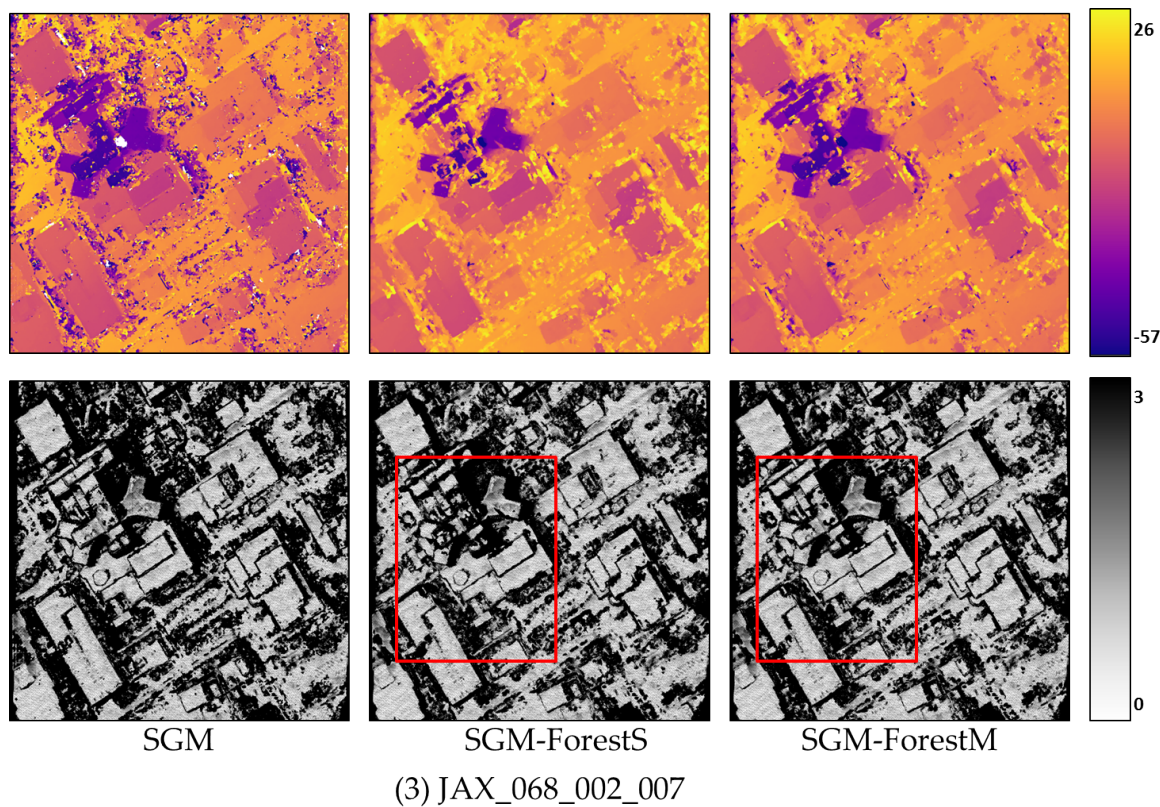


SGM

SGM-ForestS

SGM-ForestM

(2) JAX\_018\_012\_001



**Figure 8.** Results on stereo datasets from the 2019 IEEE GRSS data fusion contest (Track 2, pairwise semantic stereo challenge).

374        The reference LiDAR data were collected several years before the satellite images. Therefore, the  
 375 images containing stable objects, e.g. buildings, are selected for visualization and evaluation. It is  
 376 found that SGM-ForestM is capable of better recovering the roads and buildings (as highlighted by the  
 377 red rectangles).

## 378 5. Conclusions

379        In this paper, we propose SGM-ForestM as an extension of SGM-ForestS based on a multi-label  
 380 classification strategy. Compared with the single scanline selection scheme of the latter using random  
 381 forest, we collect all the promising scanlines, given that normally more than one scanline is capable of  
 382 predicting the correct disparity. We test the method on several datasets from close-range imagery, to  
 383 airborne and satellite data. The results indicate that SGM-ForestM performs better almost in all cases,  
 384 since it reconstructs the ill-posed regions more reasonably, e.g. textureless areas, reflective surfaces, etc.  
 385 It is found that the inference of the random forest is improved when using the proposed multi-label  
 386 scheme, leading to improvements between 0.5% to 2.3%, depending on the benchmark used.

387        In future work, the idea of adaptive scanline selection can be embedded to other stereo matching  
 388 systems as a further optimization step, such as the Sgm-nets [24], or an end-to-end network.  
 389 Furthermore, self-supervision is promising as the random forest has low demand on the number  
 390 of training samples. A rigid standard can be set to exclude outliers for a reliable self-training.

391 **Author Contributions:** conceptualization, Y.X. and P.A.; methodology, Y.X.; software, Y.X. and P.A.; validation,  
 392 Y.X.; investigation, Y.X.; resources, P.R.; data curation, Y.X.; writing—original draft preparation, Y.X.; writing—review  
 393 and editing, P.A., J.T., F.F. and P.R.; supervision, P.A., J.T. and F.F.; funding acquisition, J.T.

394 **Funding:** This research was funded by the “ForDroughtDet” project (FKZ: 22WB410602), from the  
 395 Waldklimafonds, under joint leadership of Bundeslandwirtschafts (BMEL) and Bundesumweltministerium  
 396 (BMU).



397 **Acknowledgments:** We are indebted to the Middlebury College and the Swiss Federal Institute of Technology in  
 398 Zurich (ETH Zürich) for providing the benchmark datasets. The authors would like to thank EuroSDR and the  
 399 Johns Hopkins University Applied Physics Laboratory and IARPA for providing the data used in this study, and  
 400 the IEEE GRSS Image Analysis and Data Fusion Technical Committee for organizing the Data Fusion Contest.

401 **Conflicts of Interest:** The authors declare no conflicts of interest.

## 402 Abbreviations

403 The following abbreviations are used in this manuscript:

404 MC-CNN	Matching Cost based on Convolutional Neural Networks
SGM	Semi-Global Matching
SGM-ForestS	SGM-Forest based on single-label classification strategy
405 SGM-ForestM	SGM-Forest based on multi-label classification strategy
SO	Scanline Optimization
WTA	winner-take-all

## 406 References

- 407 1. Bleyer, M.; Breiteneder, C., Stereo matching—state-of-the-art and research challenges. In *Advanced Topics in*  
 408 *Computer Vision*; Farinella, G.M.; Battiato, S.; Cipolla, R., Eds.; Springer London: London, 2013; pp. 143–179.  
 409 doi:10.1007/978-1-4471-5520-1\_6.
- 410 2. Hirschmüller, H. Accurate and efficient stereo processing by semi-global matching and mutual information.  
 411 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005, Vol. 2,  
 412 pp. 807–814 vol. 2. doi:10.1109/CVPR.2005.56.
- 413 3. d’Angelo, P.; Reinartz, P. Semiglobal matching results on the ISPRS stereo matching benchmark.  
 414 *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **2011**,  
 415 XXXVIII-4/W19, 79–84. doi:10.5194/isprsarchives-XXXVIII-4-W19-79-2011.
- 416 4. d’Angelo, P. Improving semi-global matching: Cost aggregation and confidence measure. *ISPRS*  
 417 *- International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **2016**,  
 418 XLI-B1, 299–304. doi:10.5194/isprs-archives-XLI-B1-299-2016.
- 419 5. Hirschmüller, H. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern*  
 420 *Anal. Mach. Intell.* **2008**, *30*, 328–341. doi:10.1109/TPAMI.2007.1166.
- 421 6. Scharstein, D.; Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence  
 422 algorithms. *Int. J. Comput. Vis.* **2002**, *47*, 7–42. doi:10.1023/A:1014573219977.
- 423 7. Hirschmüller, H. Semi-global matching - motivation, developments and applications. Photogrammetric  
 424 Week. Wichmann Verlag Heidelberg, Germany, 2011, Vol. 11, pp. 173–184.
- 425 8. Kusch, G.; d’Angelo, P.; Qin, R.; Poli, D.; Reinartz, P.; Cremers, D. DSM accuracy evaluation for the ISPRS  
 426 commission I image matching benchmark. *ISPRS - International Archives of the Photogrammetry, Remote*  
 427 *Sensing and Spatial Information Sciences* **2014**, XL-1, 195–200. doi:10.5194/isprsarchives-XL-1-195-2014.
- 428 9. Qin, R.; Huang, X.; Gruen, A.; Schmitt, G. Object-based 3-D building change detection on  
 429 multitemporal stereo images. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2015**, *8*, 2125–2137.  
 430 doi:10.1109/JSTARS.2015.2424275.
- 431 10. Schönberger, J.L.; Sinha, S.N.; Pollefeys, M. Learning to fuse proposals from multiple scanline optimizations  
 432 in semi-global matching. *Computer Vision – ECCV 2018*; Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss,  
 433 Y., Eds.; Springer International Publishing: Cham, 2018; pp. 758–775.
- 434 11. Scharstein, D.; Hirschmüller, H.; Kitajima, Y.; Krathwohl, G.; Nešić, N.; Wang, X.; Westling, P.  
 435 High-resolution stereo datasets with subpixel-accurate ground truth. In *Lecture Notes in Computer Science*;  
 436 Springer International Publishing, 2014; pp. 31–42. doi:10.1007/978-3-319-11752-2\_3.
- 437 12. Menze, M.; Geiger, A. Object scene flow for autonomous vehicles. 2015 IEEE Conference on Computer  
 438 Vision and Pattern Recognition (CVPR), 2015, pp. 3061–3070. doi:10.1109/CVPR.2015.7298925.
- 439 13. Schöps, T.; Schönberger, J.L.; Galliani, S.; Sattler, T.; Schindler, K.; Pollefeys, M.; Geiger, A. A multi-view  
 440 stereo benchmark with high-resolution images and multi-camera videos. 2017 IEEE Conference on  
 441 Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2538–2547. doi:10.1109/CVPR.2017.272.

- 442 14. Scharstein, D.; Pal, C. Learning conditional random fields for stereo. 2007 IEEE Conference on Computer  
443 Vision and Pattern Recognition (CVPR), 2007, pp. 1–8. doi:10.1109/CVPR.2007.383191.
- 444 15. Hirschmüller, H.; Scharstein, D. Evaluation of cost functions for stereo matching. 2007 IEEE Conference  
445 on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–8. doi:10.1109/CVPR.2007.383248.
- 446 16. Scharstein, D.; Hirschmüller, H.; Kitajima, Y.; Krathwohl, G.; Nešić, N.; Wang, X.; Westling, P.  
447 High-resolution stereo datasets with subpixel-accurate ground truth. German Conference on Pattern  
448 Recognition (GCPR 2014), Münster, Germany. Springer, 2014, pp. 31–42.
- 449 17. Haala, N. Dense image matching final report. *EuroSDR Publication Series, Official Publication* **2014**,  
450 *64*, 115–145.
- 451 18. Bosch, M.; Foster, K.; Christie, G.A.; Wang, S.; Hager, G.D.; Brown, M.Z. Semantic Stereo for Incidental  
452 Satellite Images. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* **2019**, pp. 1524–1532.
- 453 19. Le Saux, B.; Yokoya, N.; Hansch, R.; Brown, M.; Hager, G.; Kim, H. 2019 IEEE GRSS Data Fusion Contest:  
454 Semantic 3D Reconstruction [Technical Committees]. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 103–105.  
455 doi:10.1109/MGRS.2019.2893783.
- 456 20. Birchfield, S.; Tomasi, C. Depth discontinuities by pixel-to-pixel stereo. Sixth International Conference on  
457 Computer Vision (ICCV), 1998, pp. 1073–1080. doi:10.1109/ICCV.1998.710850.
- 458 21. Ni, J.; Li, Q.; Liu, Y.; Zhou, Y. Second-Order Semi-Global Stereo Matching Algorithm Based on Slanted  
459 Plane Iterative Optimization. *IEEE Access* **2018**, *6*, 61735–61747. doi:10.1109/ACCESS.2018.2876420.
- 460 22. Zbontar, J.; LeCun, Y. Stereo matching by training a convolutional neural network to compare image  
461 patches. *J. Mach. Learn. Res.* **2016**, *17*, 1–32.
- 462 23. Luo, W.; Schwing, A.G.; Urtasun, R. Efficient Deep Learning for Stereo Matching. 2016 IEEE Conference  
463 on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 5695–5703. doi:10.1109/CVPR.2016.614.
- 464 24. Seki, A.; Pollefeys, M. Sgm-nets: semi-global matching with neural networks. 2017 IEEE Conference on  
465 Computer Vision and Pattern Recognition (CVPR), 2017, pp. 6640–6649. doi:10.1109/CVPR.2017.703.
- 466 25. Scharstein, D.; Tani, T.; Sinha, S.N. Semi-global stereo matching with surface orientation priors. 2017  
467 International Conference on 3D Vision (3DV), 2017, pp. 215–224. doi:10.1109/3DV.2017.00033.
- 468 26. Michael, M.; Salmen, J.; Stallkamp, J.; Schlipsing, M. Real-time stereo vision: Optimizing  
469 semi-global matching. 2013 IEEE Intelligent Vehicles Symposium (IV), 2013, pp. 1197–1202.  
470 doi:10.1109/IVS.2013.6629629.
- 471 27. Poggi, M.; Mattocchia, S. Learning a general-purpose confidence measure based on O(1) features and a  
472 smarter aggregation strategy for semi global matching. 2016 Fourth International Conference on 3D Vision  
473 (3DV), 2016, pp. 509–518. doi:10.1109/3DV.2016.61.
- 474 28. Zhang, F.; Prisacariu, V.; Yang, R.; Torr, P.H. GA-Net: Guided Aggregation Net for End-to-end Stereo  
475 Matching. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019,  
476 pp. 185–194.
- 477 29. Bromley, J.; Bentz, J.; Bottou, L.; Guyon, I.; LeCun, Y.; Moore, C.; Sackinger, E.; Shah, R. Signature  
478 Verification using a “Siamese” Time Delay Neural Network. *Int. J. Pattern Recognit. Artif. Intell.* **1993**, *7*.
- 479 30. Zabih, R.; Woodfill, J. Non-parametric local transforms for computing visual correspondence. Computer  
480 Vision — ECCV '94; Eklundh, J.O., Ed.; Springer Berlin Heidelberg: Berlin, Heidelberg, 1994; pp. 151–158.
- 481 31. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2 ed.; Wiley: New York, 2001.
- 482 32. Boutell, M.R.; Luo, J.; Shen, X.; Brown, C.M. Learning multi-label scene classification. *Pattern Recognit.*  
483 **2004**, *37*, 1757 – 1771. doi:https://doi.org/10.1016/j.patcog.2004.03.009.
- 484 33. McCallum, A. Multi-label text classification with a mixture model trained by EM. AAAI workshop on Text  
485 Learning, 1999, pp. 1–7.
- 486 34. Schapire, R.E.; Singer, Y. BoosTexter: A boosting-based system for text categorization. *Mach. Learn.* **2000**,  
487 *39*, 135–168. doi:10.1023/A:1007649029923.
- 488 35. Clare, A.; King, R.D. Knowledge discovery in multi-label phenotype data. Principles of Data Mining and  
489 Knowledge Discovery; De Raedt, L.; Siebes, A., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2001;  
490 pp. 42–53.
- 491 36. Tsoumakas, G.; Katakis, I. Multi-label classification: An overview. *Int. J. Data Warehous. Min. (IJDWM)*  
492 **2007**, *3*, 1–13.
- 493 37. Schumacher, F.; Greiner, T. Matching cost computation algorithm and high speed FPGA  
494 architecture for high quality real-time semi global matching stereo vision for road scenes. 17th

- 495 International IEEE Conference on Intelligent Transportation Systems (ITSC), 2014, pp. 3064–3069.  
496 doi:10.1109/ITSC.2014.6958182.
- 497 38. Gehrig, S.K.; Rabe, C. Real-time semi-global matching on the CPU. 2010 IEEE Computer  
498 Society Conference on Computer Vision and Pattern Recognition - Workshops, 2010, pp. 85–92.  
499 doi:10.1109/CVPRW.2010.5543779.
- 500 39. Arndt, O.J.; Becker, D.; Banz, C.; Blume, H. Parallel implementation of real-time semi-global matching  
501 on embedded multi-core architectures. 2013 International Conference on Embedded Computer Systems:  
502 Architectures, Modeling, and Simulation (SAMOS), 2013, pp. 56–63. doi:10.1109/SAMOS.2013.6621106.
- 503 40. Gehrig, S.K.; Eberli, F.; Meyer, T. A real-time low-power stereo vision engine using semi-global matching.  
504 Computer Vision Systems; Fritz, M.; Schiele, B.; Piater, J.H., Eds.; Springer Berlin Heidelberg: Berlin,  
505 Heidelberg, 2009; pp. 134–143.
- 506 41. Banz, C.; Hesselbarth, S.; Flatt, H.; Blume, H.; Pirsch, P. Real-time stereo vision system using  
507 semi-global matching disparity estimation: Architecture and FPGA-implementation. 2010 International  
508 Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, 2010, pp. 93–101.  
509 doi:10.1109/ICSAMOS.2010.5642077.
- 510 42. 2019 IEEE GRSS Data Fusion Contest. [http://www.grss-ieee.org/community/technical-committees/data-](http://www.grss-ieee.org/community/technical-committees/data-fusion)  
511 [fusion](http://www.grss-ieee.org/community/technical-committees/data-fusion). Accessed: 2019-11-26.



## Appendices






- C Xia, Y., d'Angelo, P., Fraundorfer, F., Tian, J., Fuentes Reyes, M. and Reinartz, P., 2022. GA-Net-Pyramid: an efficient end-to-end network for dense matching. Remote Sensing, 14(8), p.1942.**

<https://www.mdpi.com/2072-4292/14/8/1942>



Article

# GA-Net-Pyramid: An Efficient End-to-End Network for Dense Matching

Yuanxin Xia <sup>1,\*</sup>, Pablo d'Angelo <sup>1</sup>, Friedrich Fraundorfer <sup>1,2</sup>, Jiaojiao Tian <sup>1</sup>, Mario Fuentes Reyes <sup>1</sup>  
and Peter Reinartz <sup>1</sup>

<sup>1</sup> Department of Photogrammetry and Image Analysis, Remote Sensing Technology Institute, German Aerospace Center (DLR), 82234 Wessling, Germany; pablo.angelo@dlr.de (P.d.); fraundorfer@icg.tugraz.at (F.F.); jiaojiao.tian@dlr.de (J.T.); mario.fuentesReyes@dlr.de (M.F.R.); peter.reinartz@dlr.de (P.R.)

<sup>2</sup> Institute of Computer Graphics and Vision, Graz University of Technology (TU Graz), Graz 8010, Austria

\* Correspondence: yuanxin.xia@dlr.de; Tel.: +49-8153-2816-37

**Abstract:** Dense matching plays a crucial role in computer vision and remote sensing, to rapidly provide stereo products using inexpensive hardware. Along with the development of deep learning, the Guided Aggregation Network (GA-Net) achieves state-of-the-art performance via the proposed Semi-Global Guided Aggregation layers and reduces the use of costly 3D convolutional layers. To solve the problem of GA-Net requiring large GPU memory consumption, we design a pyramid architecture to modify the model. Starting from a downsampled stereo input, the disparity is estimated and continuously refined through the pyramid levels. Thus, the disparity search is only applied for a small size of stereo pair and then confined within a short residual range for minor correction, leading to highly reduced memory usage and runtime. Tests on close-range, aerial, and satellite data demonstrate that the proposed algorithm achieves significantly higher efficiency (around eight times faster consuming only 20–40% GPU memory) and comparable results with GA-Net on remote sensing data. Thanks to this coarse-to-fine estimation, we successfully process remote sensing datasets with very large disparity ranges, which could not be processed with GA-Net due to GPU memory limitations.

**Keywords:** dense matching; deep learning; convolutional neural networks; end-to-end; pyramid architecture



**Citation:** Xia, Y.; d'Angelo, P.; Fraundorfer, F.; Tian, J.; Fuentes Reyes, M.; Reinartz, P. GA-Net-Pyramid: An Efficient End-to-End Network for Dense Matching. *Remote Sens.* **2022**, *1*, 0. <https://doi.org/>

Academic Editor: Sander Oude Elberink

Received: 8 March 2022  
Accepted: 13 April 2022  
Published:

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Over the years, dense stereo matching has been studied persistently in the field of computer vision, remote sensing, and photogrammetry, as the corresponding applications keep promoting the development of self-driving, urban digitization, topographic survey, forest management, etc. [1–4]. Given a pair of images with the camera parameters and the relative distance (baseline) in between, the object depth is computed which extends 2D image information to 3D knowledge of the scene [5]. In stereo matching, the depth is obtained in the form of disparity which presents the (horizontal) displacement of two corresponding pixels from each of the (rectified) stereo pair, respectively. A disparity map allows each pixel to be triangulated to its location in the 3D space. Stereo vision methods define two terms for locating correspondences, the data term and smoothness term. The former searches pixels with similar intensity as potential matches, while the latter requires close disparity predictions between neighboring points for spatial smoothness. Semi-Global Matching (SGM) is a representative method in stereo matching [6]. The algorithm acquires dense correspondences via a simple pixel-wise cost comparison under a disparity searching range, and guarantees the (piece-wise) smoothness of the reconstructed surface simultaneously. For each target pixel, the previous point along a certain path is also considered to avoid neighboring disparity inconsistency. By repeatedly applying

the strategy through multiple (normally 8 or 16) symmetric paths, 2D regularization is performed while keeping the algorithm computationally feasible.

As more high-quality, high-resolution data become available, the computational cost of dense matching rises exponentially, especially in the field of remote sensing. To limit the memory usage and runtime, Rothermel [7] proposed tSGM. Images are firstly down-sampled to several scales constituting a pyramid structure, in which the dense matching is applied from the lowest resolution to the highest, level by level. On the pyramid top, the disparity range is downscaled accordingly together with the image size, leading to reduced workload. The matching result is then passed to the next higher resolution level as an initial prediction, from which a small disparity buffer is set as a new search range to locally refine the estimation. The coarse-to-fine scheme thus greatly reduces the demand for memory and runtime. Moreover, the influence of ambiguous disparity candidates is limited. Additionally, this strategy enables the use of deep learning-based algorithms, which typically only support small search ranges due to memory limits on datasets with large disparity ranges of sometime several thousand pixels, as typically occurring in extreme mountainous regions, such as the Himalayas.

Recently, Zhang et al. [8] introduced their GA-Net, which approximates SGM as a differentiable Semi-Global Guided Aggregation (SGA) layer, to construct an end-to-end neural network for stereo matching. All the user-defined parameters in SGM can be learned; thus, the smoothness requirement is satisfied in a smarter way depending on the specific scene situation. With SGA and only a few 3D convolutional layers to regularize the cost volume, GA-Net is more efficient than other networks, e.g., GC-Net [9], PSMNet [10], etc., and achieves state-of-the-art performance. For processing high-resolution remote sensing data, however, the training and prediction are still memory- and time-consuming (days are needed for training on patches of  $384 \times 576$ , with  $[0, 192]$  as the disparity search range, consuming around 15 GB GPU memory for each batch).

Inspired by tSGM and some corresponding pyramid networks [11–13], we adjust GA-Net to a pyramid architecture, and propose our GA-Net-Pyramid. The disparity is initially estimated for the full depth range at the coarsest resolution, then refined through the pyramid. Thus, we enhance the efficiency of the algorithm significantly, with moderately decreased accuracy especially for remote sensing data. To summarize our contributions:

- Firstly, we propose a hierarchical strategy for GA-Net stereo matching to estimate the depth from coarse to fine, for which two pyramid models are introduced with explicit or implicit image downsampling, respectively. A trainable Spatial Propagation Network (SPN) [14] is tested as a post-processing step to sharpen the depth boundaries. It is shown that the effect from SPN varies depending on the target data domain.
- Secondly, the proposed methods are tested on cross-domain datasets, from close-range benchmarks, Scene Flow [15] and KITTI-2012 [16], to large-scale aerial/satellite stereo data. We prove that our algorithm is robust and consistently more efficient in all cases. We also build a stereo dataset, consisting of simultaneously acquired 30-cm satellite and 6-cm aerial imagery which are co-registered to sub-pixel precision. This is particularly important for remote sensing scenarios, considering that the currently published data, such as [17], cannot provide reliable ground truth disparity maps, due to different sensing modalities or scene changes caused by temporal inconsistency.
- At last, we successfully solve a satellite stereo task on stereo pairs with very large disparity ranges, which cannot be handled by the baseline model GA-Net.

The rest of the paper is organized as follows: In Section 2, traditional stereo methods, SGM and its variants are recapped, which enlighten the main idea of GA-Net and our GA-Net-Pyramid. We also describe representative learning-based algorithms, from hybrid approaches replacing certain traditional components with deep learning-based ones, to full end-to-end stereo networks. Afterwards, we state the principle of our method, GA-Net-Pyramid, with a review of its prototype GA-Net in Section 3. In Section 4, we present a detailed comparison between GA-Net and our GA-Net-Pyramid on various datasets. At



last, we discuss the strengths and limitations of the method in Section 5, and conclude the paper in Section 6.

## 2. Related Work

### 2.1. Traditional Stereo Methods

Conventional stereo matching algorithms define two terms to find dense correspondences from a stereo pair, data term and smoothness term [5]. The data term measures the photo consistency between potentially matched pixels through a pre-defined disparity range. The smoothness term guarantees a smooth reconstructed surface by limiting neighboring points' disparity differences. SGM well balanced the two terms via a scanline optimization strategy, which was widely applied thanks to the good compromise between accuracy and efficiency [6,18,19]. The strategy was further improved with a dynamic searching range for correspondences through a pyramid structure, leading to tSGM which consumed less memory and runtime [7]. As More Global Matching (MGM) was proposed, the support from neighboring pixels was increased without extra overhead, by additionally considering the previous scanline visited already [19,20]. Compared with other traditional stereo methods [21–25], which may solely rely on the cost function and winner-takes-all (WTA) strategy resulting in limited accuracy, or struggle to find the minimum global energy under certain runtime or memory budget, the SGM variants achieve robust stereo estimation consuming reasonable computational resource.

### 2.2. Learning-Assisted Stereo Methods

#### 2.2.1. Integration of Conventional Stereo Methods and Machine Learning

Recent advances in machine/deep learning and convolutional neural networks (CNNs) enable the learning of data representation [26], and promote the development of stereo matching with a series of state-of-the-art algorithms. Deep learning could be exploited to extract features from images, in order to better measure the similarity for matching cost calculation. Zbontar and LeCun [27] used a Siamese network [28] to extract features from two patches symmetrically, after which a cost volume was constructed and regularized by SGM. The idea was adjusted by Luo et al. [29] based on multi-class classification, achieving faster estimation. Regarding the cost aggregation and disparity computation, Seki and Pollefeys [30] proposed their SGM-Net to learn the penalty terms for conflicting disparity predictions from neighboring points. Michael et al. [31] considered a specific weight for each scanline in SGM to achieve a weighted 2D scanline optimization, since varying performance could be obtained via each scanline depending on the scene structure. Poggi and Mattoccia [32] constructed a feature vector for each pixel according to the disparity estimation via a single scanline. The feature represented the statistical dispersion of surrounding disparities, which could be analyzed by a random forest to predict a confidence measure of the scanline for a weighted scanline summation. Similar work was accomplished in [33,34]. The disparity predicted by each scanline and the corresponding costs were fed to a random forest, so that the better performed scanlines were adaptively selected. The corresponding disparity estimation could serve as a reference to guide the further stereo prediction.

#### 2.2.2. End-to-End Stereo Networks

The above methods mainly integrated deep learning with traditional stereo matching techniques for better performance, which were then followed by encoder-decoder structures for depth prediction as an end-to-end system. Dosovitskiy et al. [35] firstly presented a network, FlowNet, to estimate optical flow directly from a stereo pair. They used a correlation layer to measure the similarity between corresponding patches. Mayer et al. [15] then designed a large synthetic dataset, Scene Flow, allowing an initial training of deep neural networks before adjusting to specific scenarios. They also proposed DispNet and DispNet-Corr, as one of the first end-to-end stereo matching networks. Kendall et al. [9] proposed GC-Net, which applied 3D convolutions to regularize the cost volume, with both geometry and context information incorporated. Chang and Chen [10] introduced a pyramid pooling

module in their PSMNet to aggregate multi-scale features. Thus, the global context and local details were simultaneously contained within the cost volume. Guo et al. [36] improved PSMNet by proposing the group-wise correlation stereo network (GwcNet). They constructed a group-wise correlation-based cost volume which required less parameters for the cost aggregation, achieving similar performance as PSMNet. Zhu et al. [37] proposed a multi-scale pyramid aggregation module to handle the cost volume, leading to MPANet with significantly better disparity estimation for foreground objects. Xu and Zhang [38] proposed AANet, utilizing intra- and cross-scale cost aggregation, which delivered better results for depth discontinuities and large textureless area. Wang et al. [39] applied a recurrent unit to iteratively refine the stereo estimation, and designed a pyramid voting module to produce a semi-dense disparity map for self-supervision. Confident disparity prediction was achieved via seeking consistent estimation across scales. Inspired by SGM, Zhang et al. [8] proposed the GA-Net using so-called SGA layer for cost aggregation, to replace 3D convolution which was computationally expensive. They achieved great performance on multiple benchmark datasets, which coincided with the idea from [40] that classical stereo matching methods could serve as a robust guideline to develop deep learning-based algorithms, rather than designing a pure learning architecture. Semantic information could also be involved for stereo matching problems [41,42] as the object boundaries mostly corresponded to the depth discontinuities. The two tasks supported each other leading to a win-win situation. Other works included cost distribution study, disparity refinement, cross-domain prediction, stereo neural architecture search, etc., which boosted the state-of-the-art constantly [40,43–47].

Recently, the pyramid architecture was tested in a learning-based stereo framework, since the efficiency could be largely enhanced via a coarse-to-fine estimation [11–13,48,49]. Regarding the architecture in [11–13] as a baseline model, the stereo correspondences were firstly located on the pyramid top using downsampled features. Then, the disparity was iteratively refined through the network towards the pyramid bottom in full resolution, which considerably reduced the computational effort and GPU memory consumption. Chang et al. [48] benefited from the architecture to achieve real-time performance, with an attention-aware feature aggregation module for better representative ability of the feature. Compared with these methods, our contributions are different. At first, we additionally test our model on airborne and spaceborne images. We fill the application gap of the previous research, considering the very limited test cases applying newly proposed computer vision algorithms in the field of remote sensing. The proposed model is proven effective to process stereo imagery with large disparity range (thousand pixels) over mountain areas. It should be noted that our model acquires no supervision in training phase on stereo data with large baselines, with no need to normalize/denormalize the disparity measurement in test phase as [50]. This is, to the best of our knowledge, a novel showcase of adapting well-performed computer vision models to deliver high-quality geographical products in extreme regions. In addition, our baseline is the up-to-date model GA-Net-deep from [8], rather than the shallower and less accurate version GA-Net-11 used in [49].

### 3. Methodology

In this section, we recap GA-Net by presenting the proposed SGA and LGA (Local Guided Aggregation) layers, which approximate SGM for cost regularization and protect thin structures, respectively. SGM applies the scanline optimization strategy to efficiently locate stereo correspondences and avoids the streaking problem. For a detailed description of SGM, we encourage readers to follow the papers [6,51]. Afterwards, we describe our pyramidal extension of GA-Net, GA-Net-Pyramid. Two architectures are proposed. The first model explicitly downsamples the input stereo pair according to the pyramid level, and simply applies GA-Net on each level to regress disparity. The second model applies a different feature extraction strategy via a U-Net [52] structure to generate multi-scale features implicitly.

### 3.1. GA-Net

In traditional SGM, the scanline optimization technique [53] is applied to satisfy the spatial smoothness, by limiting the depth difference between neighboring pixels. To avoid the streaking problem, a pixel is accessed through multiple scanlines simultaneously along several canonical directions, typically 8 or 16, to consider the disparity estimation from its neighbor. Along a certain scanline traversing in direction  $r$ , the cost for a pixel located at the image position  $p$  assuming  $d$  as the disparity, is calculated as:

$$L_r(p, d) = C(p, d) + \min(L_r(p - r, d), L_r(p - r, d - 1) + P_1, L_r(p - r, d + 1) + P_1, \min_i L_r(p - r, i) + P_2). \quad (1)$$

In the above equation, the photo inconsistency is measured by  $C(p, d)$ , while  $P_1$  and  $P_2$  are defined for penalizing the prediction when the previous neighboring point  $p - r$  prefers a different disparity value. In practice, however, two problems exist. Firstly, the users need expertise to determine appropriate  $P_1$  and  $P_2$  to punish neighboring disparity inconsistency. Tuning of  $P_1$  and  $P_2$  additionally depends on scene structure and the used similarity measure. Moreover, the values of  $P_1$  and  $P_2$  are fixed throughout the stereo processing or simply adapted according to, e.g., pixel gradients, which are not optimal for all the pixels within the image, especially under a varied scene structure, e.g., from plains to mountains.

GA-Net addresses these issues by introducing the SGA layer, a differentiable approximation of Equation (1) that is suitable for an end-to-end trainable network. Specifically, the master epipolar image provides guiding information through a sub-network to better penalize depth discontinuity, and enable a self-adaptive parameter setting. Thus, the penalty terms for conflicting neighboring disparities are determined according to the pixel location and scanline direction, which is more reasonable for smoothness regularization. Via the guidance sub-network, a weight is supplied for each term in Equation (1) to simulate the scanline optimization in SGM, leading to the following equation:

$$L_r(p, d) = C(p, d) + \text{sum}(w_1(p, r) \cdot L_r(p - r, d), w_2(p, r) \cdot L_r(p - r, d - 1), w_3(p, r) \cdot L_r(p - r, d + 1), w_4(p, r) \cdot \max_i L_r(p - r, i)). \quad (2)$$

Compared with Equation (1), the punishment from  $P_1$  and  $P_2$  is replaced by the relative importance (weight)  $w_i$  of each term, which is predicted separately for each pixel along a directed scanline. Moreover, there are two differences with SGM, one of which is that the first/external minimum operation is substituted by a weighted sum. This can be regarded as a replacement from a max-pooling layer to a convolution with strides, which is proven effective without accuracy loss [54]. In addition, the second/internal minimum search is changed to a maximum, which embodies the learning target to maximize the probability at the ground truth disparity rather than minimizing the cost. To avoid the exploding accumulation of  $L_r(p, d)$  along the scanline,  $C(p, d)$  is also included within the weighted summation, with the sum of all the weights equal to 1. Thus, SGA is finally formulated as:

$$L_r(p, d) = \text{sum}(w_0(p, r) \cdot C(p, d), w_1(p, r) \cdot L_r(p - r, d), w_2(p, r) \cdot L_r(p - r, d - 1), w_3(p, r) \cdot L_r(p - r, d + 1), w_4(p, r) \cdot \max_i L_r(p - r, i)), \quad (3)$$

$$\sum_{i=0,1,2,3,4} w_i(p, r) = 1.$$

In SGM, the cost  $L_r(p, d)$  from each scanline is simply summed up to approximate 2D smoothness, which was demonstrated to be not reasonable for incurring inferior scanlines [33,34]. Accordingly, GA-Net takes the maximum as  $L(p, d) = \max_r L_r(p, d)$  to keep the best information.

The guidance sub-network also provides weights for another layer, LGA, to further filter the cost volume as below:

$$L_*(p, d) = \text{sum} \left( \begin{array}{l} \sum_{q \in N_p} w_0(p, q) \cdot L(q, d), \\ \sum_{q \in N_p} w_1(p, q) \cdot L(q, d - 1), \\ \sum_{q \in N_p} w_2(p, q) \cdot L(q, d + 1) \end{array} \right), \quad (4)$$

$$\sum_{q \in N_p} w_0(p, q) + w_1(p, q) + w_2(p, q) = 1,$$

from which a 3D neighborhood (in both spatial and disparity dimensions) centered around each pixel within the cost volume is utilized for a weighted average to protect thin structures. Afterwards as suggested by [9], a softmax operation  $\sigma(\cdot)$  is applied to the filtered cost volume in order to acquire a normalized probability for each disparity candidate (from  $[0, D_{max}]$ ) and regress the final disparity value  $\hat{d}$  as:

$$\hat{d} = \sum_{d=0}^{D_{max}} d \times \sigma(-L_{*d}). \quad (5)$$

### 3.2. GA-Net-Pyramid with Explicit Downsampling

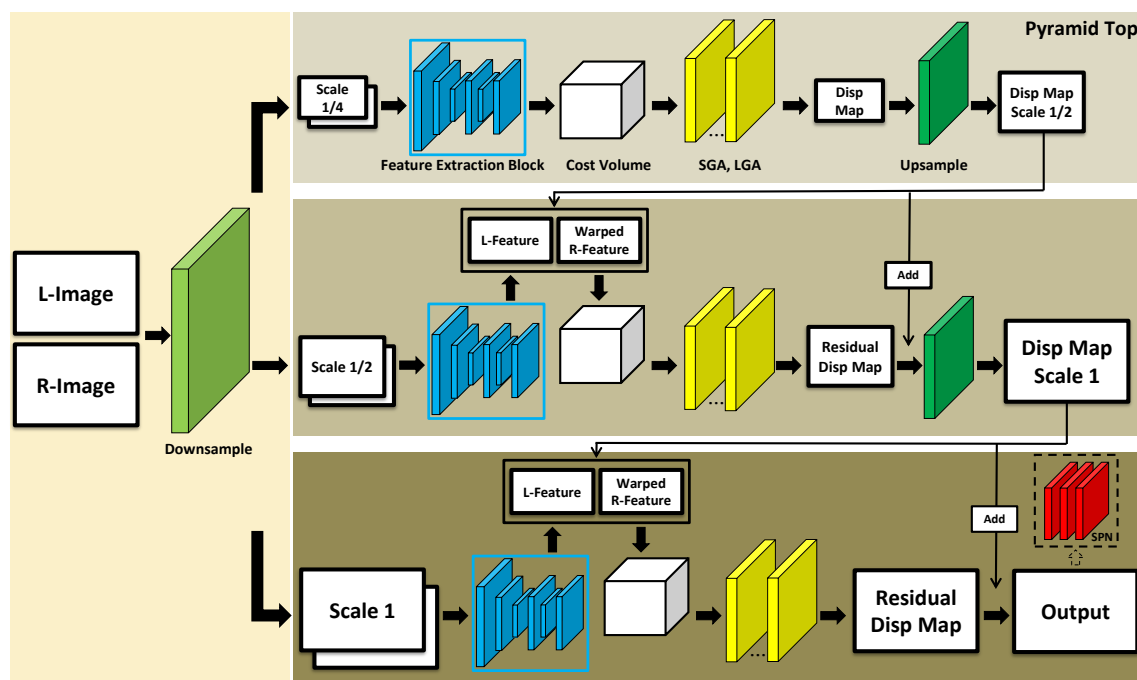
GA-Net adapts the scanline optimization scheme to an end-to-end stereo matching system. Inspired by SGM, the disparity of each pixel can be estimated with the support from all the previous neighbors along multiple paths, instead of a pure convolution-based encoder-decoder to regularize the cost volume. Furthermore, the proposed SGA and LGA layers are computationally more efficient than 3D convolutions, which are used by most state-of-the-art methods [9,10]. However it can still take days to train a well performing model, when the computational power is limited. In our case, for example, the training on the Scene Flow dataset (patch size  $384 \times 576$ ), which is normally used by most stereo matching networks for the initial learning phase, takes around 12 days to finish 8 epochs on two Quadro P6000 GPU cards. Hence, the employment of the network is hampered. In the field of remote sensing, it can be imagined that GA-Net would struggle to process high-resolution aerial or satellite stereo data, especially for wide baseline stereo pairs requiring larger disparity search ranges.

Rothermel [7] proposed an improved SGM, tSGM, which constructed a pyramid architecture to search correspondences between the stereo pair from coarse to fine. Based on this strategy, comparable quality was achieved with far less memory and runtime consumed. This inspires us to restructure GA-Net with a pyramid architecture as well, to regress the depth from coarse to fine. Figure 1 presents the schematic overview of our GA-Net-Pyramid. Three pyramid levels are depicted which could be extended. We use the same stacked hourglass module (a double U-Net structure) as GA-Net, which is essentially a Siamese network [28] for symmetric feature extraction from the left and right image, respectively. The input of the feature extraction module, however, is a stereo pair downsampled in accordance with the pyramid level. Afterwards, the cost volume is generated and then processed by SGA and LGA for disparity regression, in order to guide the subsequent level for the disparity refinement until the original resolution is recovered.

#### 3.2.1. Pyramid Top

We start from the pyramid top with the original image downsampled by a factor of 4 along both row and column directions in our implementation (termed as ‘Scale 1/4’ in Figure 1). Then, the feature is extracted to construct a 4D cost volume by concatenating the left and right feature maps along the channel dimension, with a horizontal shift indicated by a disparity candidate within the search range. Assuming the cost volume on the original full-resolution image is of size  $H \times W \times D_{max} \times 2C$ , for the image height, width, the maximum disparity, and twice the channel number of the generated feature maps, respectively, our cost volume on the pyramid top reaches a highly reduced dimen-

sion as  $H/4 \times W/4 \times D_{max}/4 \times 2C$ . Thus, the memory consumption and computational complexity are decreased by a factor of  $1/64$ .



**Figure 1.** GA-Net-Pyramid with explicit downsampling. The input stereo pair is downsampled explicitly according to the resolution required by each pyramid level. At the pyramid top, the stereo correspondences are located within an absolute disparity range in low resolution. The following pyramid levels perform disparity refinement within a pre-defined residual disparity range until the original resolution is recovered at the pyramid bottom. SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement, as described in Section 3.3.

Afterwards, the cost volume enters the cost aggregation block containing SGA and LGA layers, for which the guiding information is obtained from the downscaled master epipolar image. At last, the filtered cost is used for the following disparity regression as GA-Net. Thus, a disparity map of the downsampled image ‘Scale 1/4’ is obtained for the pyramid top. From here, the depth of the scene is already roughly estimated and the large-scale context is perceived, which provides a good guidance for the following processing.

### 3.2.2. The Other Pyramid Levels

Based on the prediction of the pyramid top, the other levels thus only need to locally refine the disparity values. Therefore, the disparity map from ‘Scale 1/4’ level is upsampled by a factor of 2 via bilinear interpolation, to match the resolution of ‘Scale 1/2’ level as an initial estimation  $d_{ini}$ . Feature maps are computed for the left and right image of ‘Scale 1/2’ level as  $F_l$  and  $F_r$ , respectively. Assuming  $d_{ini}$  is accurate enough, we can warp  $F_r$  according to  $d_{ini}$  which would perfectly match  $F_l$ . However, considering the details lost through downsampling on the pyramid top and the corresponding matching error, a small shift would exist between the left and the warped right feature, which is named disparity residual and should be additionally considered for a perfect match. Accordingly, a cost volume  $CV$  is built in size of  $H/2 \times W/2 \times (2disp\_resi + 1) \times 2C$  for ‘Scale 1/2’ level. Here,  $disp\_resi$  is a pre-defined threshold, leading to a range  $[d_{ini} - disp\_resi, d_{ini} + disp\_resi]$

around the initial disparity estimation  $d_{ini}$  for refinement. The cost volume is thus formed by concatenation of  $F_l$  and  $F_r$  as:

$$CV(x, y, d) = F_l(x, y) \oplus F_r(x + (d_{ini}(x, y) + d), y), \quad d \in [-disp\_resi, +disp\_resi]. \quad (6)$$

In Equation (6),  $x$  and  $y$  are the indices of a pixel along the width and height dimension.  $\oplus$  represents the concatenation. Then, the cost volume is regularized by SGA and LGA, and a residual disparity map  $d_{resi}$  is calculated via multiplying each residual candidate to the corresponding probability and summing them up. The disparity estimation for the current level is obtained by adding the residual and the previously upscaled disparity map as:  $d_{resi} + d_{ini}$ .

The stereo pair on 'Scale 1/2' level is twice larger in height and width; however, the search for correspondences is restricted within a narrow range. Hence, only a small overhead is accumulated. We apply the same procedure for the remaining pyramid level, to continuously improve the disparity estimation until the original resolution is reached.

Each pyramid level only requires the input epipolar imagery at its level and the disparity image of the previous level. For an efficient and memory saving implementation during disparity estimation, computation of the levels could be decoupled to significantly lower the memory footprint while allowing large input image sizes. Compared to GA-Net, it is thus feasible to significantly increase both image size and disparity range, as only the pyramid top needs to process the full disparity search range, for example, processing of images with a four times larger width, height, and disparity range is possible without additional GPU memory requirements in this case. Note that the evaluation in Section 4 is recorded without adding these optimizations.

### 3.2.3. Loss

We train the model using the same smooth  $L_1$  loss function as GA-Net in [8]. However, our pyramid architecture predicts more than one disparity map, which should all be considered to allow for intermediate supervision. Hence, a weight is assigned to each pyramid level for a weighted loss summation as:

$$L = \sum_{i=1}^N l(|\hat{d}_i - \bar{d}|) \cdot \omega_i, \quad (7)$$

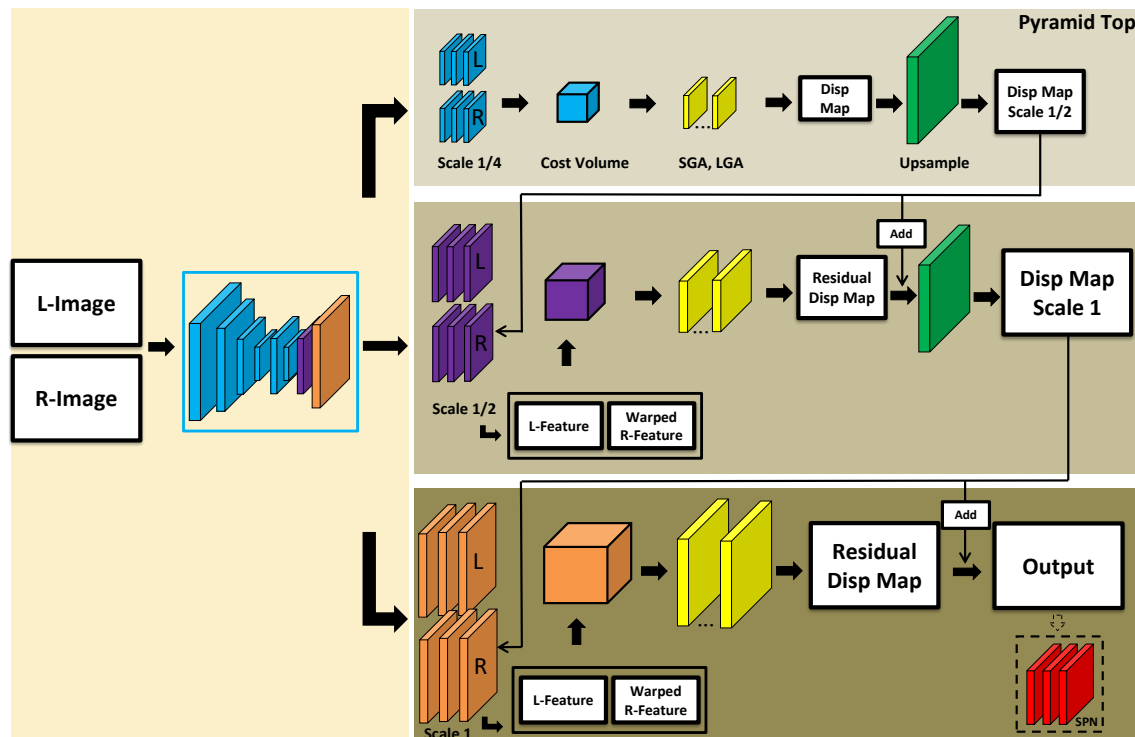
in which  $\hat{d}_i$  denotes the disparity predicted by the pyramid level  $i$  (starting from 1 at the pyramid top), and  $\bar{d}$  is the corresponding ground truth.  $l$  computes the smooth  $L_1$  loss from the disparity difference. A weight  $\omega_i$  is assigned to the level  $i$  for a weighted summation through all  $N$  pyramid levels. The disparity map from each level is upscaled to the original full resolution before computing the loss. As the estimation is improved from the pyramid top to the bottom, the corresponding weight is also increased (details for parameter setting are in Section 4).

### 3.3. GA-Net-Pyramid with Implicit Downsampling

The paper focuses on presenting a more efficient model based on the structure of GA-Net, in order to achieve robust estimation on datasets from multiple domains. Thus, we design different feature extractors and observe the corresponding performance, so that an appropriate model could be used to handle specific data types. The architecture in Figure 1 simply applies GA-Net in a pyramidal manner, which takes the linearly downsampled stereo pair as input to extract features for further processing. Therefore, we propose another architecture to implicitly learn the downsampled feature, as displayed Figure 2, such that both explicit and implicit image downsampling strategies are tested.

Instead of downsampling the input stereo pair level by level, we only use the stacked hourglass once to extract the feature from the original (full-resolution) images for feeding all the pyramid levels. The input images are firstly downsampled via convolutions with stride two, and then deconvolved to gradually recover the resolution, in which a skip

connection is exerted between corresponding feature maps of the encoder and decoder at the same resolution. Before reaching the original size, we directly extract the intermediate feature maps from the decoder to feed each level, as long as the expected resolution is acquired. To differentiate the GA-Net-Pyramid with explicit and implicit downsampling, in the following sections we name the two variants as GA-Net-PyramidED and GA-Net-PyramidID, respectively.



**Figure 2.** GA-Net-Pyramid with implicit downsampling. The feature extractor is applied on the stereo pair in original resolution, with the intermediate feature maps from its decoder to feed each pyramid level according to the expected resolution. SPN indicates the Spatial Propagation Network which is an optional module for depth boundary enhancement, as described in Section 3.3.

As the disparity is estimated and refined through the pyramid, we add a Spatial Propagation Network (SPN) as a post-processing step to explore its influence on the matching results. SPN is capable of sharpening the object boundaries, by learning from the source image (in our case, the master epipolar image) in a data-driven mode, which is appropriate as a further refinement in our pyramid architecture, especially for close-range data with rich details. Hence, four models are finally proposed including GA-Net-PyramidED and GA-Net-PyramidID, respectively, with or without SPN added at the end of the pyramid bottom.

#### 4. Experiments

In this section, we compare our GA-Net-Pyramid with GA-Net through a series of experiments on close-range, including Scene Flow and KITTI-2012, aerial, and satellite stereo datasets. For a fair comparison, the implementation details are rigidly controlled between the two algorithms. Regarding the training, we use the same patch size with a pre-defined disparity search range, to train the networks for certain epochs, based on Adam optimization strategy [55]. Each stereo pair is normalized, according to the mean and standard deviation of the pixel values from each channel, before feeding to the network. SGA is applied along four directions (horizontally and vertically) for both GA-Net-Pyramid and GA-Net.

For GA-Net-Pyramid specifically, the number of pyramid levels is 3 and the search range for the disparity residual after the pyramid top is set as  $[-6, +6]$  to refine the matching results. Details about the pyramid setting are discussed in Section 4.2.3. We apply 3 SGA and 2 LGA layers to regularize the cost volume on our pyramid top, which is the same as GA-Net. With regard to the other pyramid levels, only 1 SGA layer (with 2 LGA layers) is utilized due to the small disparity search range. The weight is set as 0.25, 0.5, and 1, to the pyramid level 1 (top), 2 and 3 (bottom), respectively, to calculate the final loss in Equation (7). The implementation of the methods is based on Python and Pytorch.

#### 4.1. Experiments on Close-Range Stereo Data

We firstly test the networks on Scene Flow and KITTI-2012 datasets, in which the scene structure is relatively complicated with rich details. Referring to most learning-based dense matching algorithms, we train the models on Scene Flow data from scratch, and utilize real data, KITTI-2012 in our case, for finetuning. Both the pre-trained and finetuned models are tested on the corresponding dataset. Regarding the former, the whole Scene Flow training dataset is used for training (8 epochs), while only 1000 stereo pairs from its validation set are selected for test to save time. On the other hand, 170 images from KITTI-2012's training data are exploited to finetune the models for 800 epochs, with the remaining 24 images for test. All the data selection is random, so that a fair evaluation is achieved. In training, we use the same patch size ( $384 \times 576$ ) with the maximum disparity set to 192. The networks are trained with a batch size of two on two Quadro P6000 GPU cards.

##### 4.1.1. Close-Range Stereo Data

Scene Flow is a synthetic dataset via randomly combining human-made objects with backgrounds from real images, which is used by most stereo networks for initial training. Afterwards, only a small dataset from a specific field is sufficient to adjust the model into practical scenarios. The dataset contains three subsets, namely FlyingThings3D, Monkaa and Driving, including around 35,000 images for training and 4370 images for validation. KITTI-2012 is a stereo dataset with a focus on outdoor street views, which is normally applied in the field of autonomous driving. The dataset includes 194 training and 195 test stereo pairs, with ground truth disparity maps based on LiDAR measurements provided or withheld.

##### 4.1.2. Visualization and Evaluation on Close-Range Stereo Data

The pre-trained networks are firstly tested on the Scene Flow dataset. The quantitative and visual comparison between our pyramid models and GA-Net is shown in Table 1 and Figure 3. As indicated by the table, we calculate the percentage of pixels, for which the estimation error is smaller than 1, 2, and 3 pixels, respectively, and the end point error (EPE) for accuracy evaluation. Regarding the efficiency, the runtime and GPU memory consumption are reported. For all the experiments in this paper, the runtime in the test period is counted for processing the whole test dataset. Specifically, we generate a binary file to save the disparity value of each correspondence, and a png (Portable Network Graphics) file to visualize the result. In the tables, M denotes megabytes for the GPU memory consumed by each network, while the time spent in training and test is expressed in hours (h) or seconds (s). Better performance is highlighted in bold.

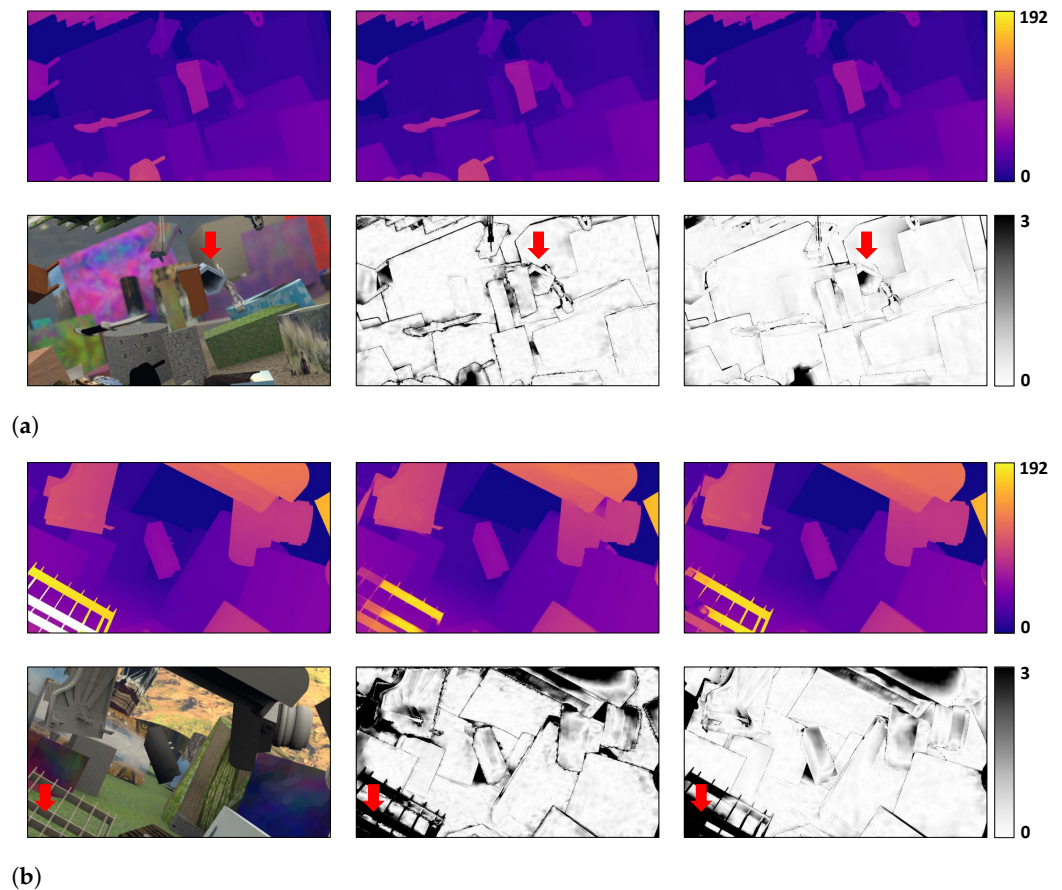


**Table 1.** Accuracy and efficiency comparison between GA-Net-Pyramid, including GA-Net-PyramidED and GA-Net-PyramidID, and GA-Net on Scene Flow data.

	Accuracy				Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	EPE	Memory	Runtime	Memory	Runtime
GA-Net-PyramidED	81.77%	88.59%	91.42%	1.61	<b>7052 M</b>	38.25h	<b>2761 M</b>	<b>0.39 h</b>
GA-Net-PyramidED+SPN	83.04%	89.97%	92.67%	1.44	7140 M	40.62 h	<b>2761 M</b>	<b>0.39 h</b>
GA-Net-PyramidID	81.26%	89.10%	92.05%	1.49	7264 M	<b>30.07 h</b>	3501 M	0.40 h
GA-Net-PyramidID+SPN	84.27%	91.09%	93.64%	1.23	7422 M	31.69 h	3501 M	<b>0.39 h</b>
GA-Net	<b>91.41%</b>	<b>95.35%</b>	<b>96.60%</b>	<b>0.86</b>	30,464 M	280.53 h	6983 M	2.10 h

<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

From the results, it is found that GA-Net outperforms the two pyramid models in accuracy; however, the latter consume much less memory and runtime in both training and test periods. In case of the close-range data, the objects are captured under an ideal viewing condition, thus very high resolution is achieved with plenty of details and texture information contained. Moreover, as Scene Flow is a synthetic dataset, the random arrangement of man-made objects makes the scene non-natural, non-logical, and highly complicated with many occlusions. Hence, our GA-Net-Pyramid is surpassed by GA-Net, considering the information loss due to a sequence of downsampling-upsampling through the pyramid levels. On the other hand, our hierarchical strategy highly simplifies the problem complexity, consuming far less computational source but at a much higher speed. Between the two pyramid models, GA-Net-PyramidED and GA-Net-PyramidID, similar accuracy is obtained. Regarding the SPN processing, a positive effect is achieved for both pyramid structures, while GA-Net-PyramidID could be improved by a larger extent. The experiments of this paper are implemented on a server open to multiple users; therefore, the runtime of each model could be slightly influenced by unknown processes. We recommend referring to the training time to evaluate the speed of the algorithms, especially for each pyramid model with similar efficiency, considering the relatively long training process compared with the test period. GA-Net-PyramidID is faster than GA-Net-PyramidED, since the feature extraction in the former case is applied only once on the full-resolution stereo pair, rather than repeatedly learning from the corresponding downsampled images level by level. In case of the GPU memory consumption, GA-Net-PyramidED performs better.



**Figure 3.** Visual comparison on Scene Flow data. Two test cases are displayed in subfigure (a) and (b). In each subfigure, the disparity maps from the ground truth, GA-Net-PyramidID+SPN and GA-Net are displayed from left to right in the first row. The second row provides the master epipolar image and the corresponding error map of each model. Regions where the proposed algorithm outperforms GA-Net are marked with red arrows.

As for the figures in the paper, only the best performed pyramid model is visually compared with GA-Net, e.g., GA-Net-PyramidID+SPN on Scene Flow dataset. Accordingly, we display the master epipolar image, where the guidance information is acquired for SGA and LGA, the ground truth, and the corresponding results from each algorithm. The color bar at the end shows the disparity and error changes. In Figure 3, it is found that GA-Net obtains a generally better disparity result than GA-Net-PyramidID+SPN, with clear edges and more details included. However, our pyramid model still produces a disparity map in good quality, even including superior depth results in certain regions. We discover that GA-Net-PyramidID+SPN is capable of better reconstructing hollow-shaped objects, e.g., the barrel and the shelf as indicated by the red arrows. The finding is also supported by the following experiments on the KITTI dataset.

The pre-trained networks are finetuned on part of KITTI-2012's training data and tested on the remaining stereo pairs. In Table 2 and Figure 4, the corresponding quantitative and qualitative results are provided. Regarding the training efficiency, only the time spent for finetuning is recorded. Similar to the previous experiment, GA-Net acquires the best accuracy, however, the pyramid models are faster and more memory friendly. SPN still improves the results of all the pyramid models, among which GA-Net-PyramidID+SPN achieves the highest accuracy. It should be noted that our GA-Net-Pyramid performs better for real data, leading to a further reduced accuracy gap compared with GA-Net. From the visual inspection, the depth result of each algorithm is barely distinguishable. Moreover as mentioned before, we obtain a better depth prediction for hollow-shaped structures (see the regions indicated by the red arrows). KITTI-2012 does not provide ground truth for the

whole scene; nevertheless, according to the image content, it is obvious that our pyramid architecture gives a clean and more reasonable depth estimation.

**Table 2.** Accuracy and efficiency comparison between GA-Net-Pyramid, including GA-Net-PyramidED and GA-Net-PyramidID, and GA-Net on KITTI-2012 data.

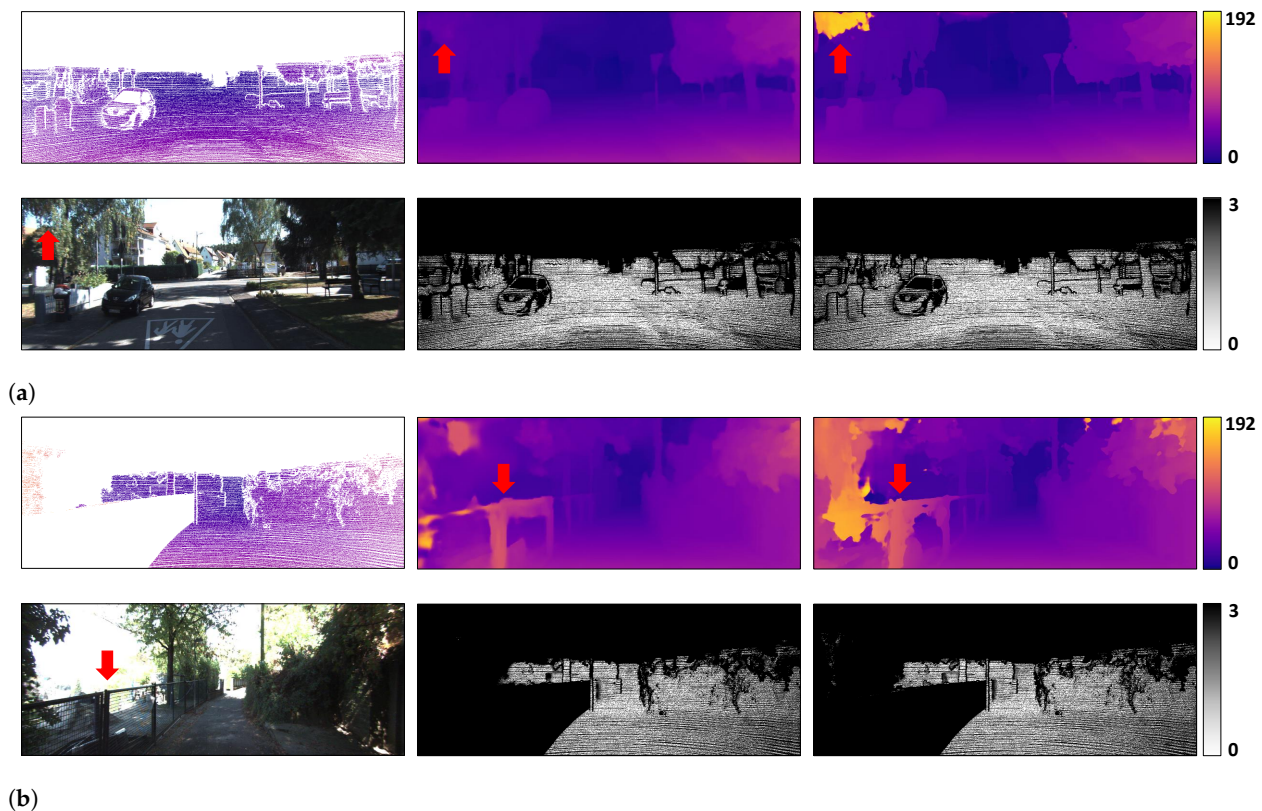
	Accuracy				Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	EPE	Memory	Runtime	Memory	Runtime
GA-Net-PyramidED	86.54%	93.57%	95.76%	0.89	<b>7140 M</b>	17.81 h	<b>2641 M</b>	28.07 s
GA-Net-PyramidED+SPN	86.56%	93.53%	95.66%	0.88	7242 M	18.49 h	<b>2641 M</b>	29.29 s
GA-Net-PyramidID	83.20%	92.68%	95.12%	1.10	7546 M	<b>13.77 h</b>	3379 M	<b>27.02 s</b>
GA-Net-PyramidID+SPN	86.88%	94.13%	96.18%	0.83	7680 M	15.02 h	3379 M	29.89 s
GA-Net	<b>91.55%</b>	<b>96.64%</b>	<b>97.65%</b>	<b>0.60</b>	30,514 M	135.47h	6565 M	165.72 s

<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

#### 4.2. Experiments on Aerial Stereo Data

In this section, the networks are tested using our aerial data. The airborne and satellite (discussed in the following section) stereo processing is the target domain of this research, since the corresponding data are usually large in size and own a much wider stereo baseline, which presents a higher demand on the algorithm's efficiency. The networks are trained on synthetic remote sensing data (854 stereo pairs) from scratch for 200 epochs, then finetuned on a subset (200 stereo pairs) of our aerial data for 100 epochs (data details are in Section 4.2.1). We randomly select another 20 aerial stereo pairs, possessing no overlap with the finetuning data, to test the trained models. Image patches in size of  $384 \times 576$  are randomly cropped for training, and the test images are  $1152 \times 1152$ . The data may contain negative or very large disparity values; hence, we exclude the stereo pairs with large baselines in order to keep the disparity range processible by both GA-Net-Pyramid and GA-Net. Accordingly, the disparity range is also set as  $[0, 192]$ . The models are trained with a batch size of two on two Quadro P6000 GPU cards.

In addition, SGM is utilized as a baseline model in our aerial and satellite experiments, since the algorithm is widely used in the field of remote sensing for dense reconstruction. We exploit Census [56] to calculate the matching cost with a  $7 \times 7$  window. The penalty terms  $P_1$  and  $P_2$  (see Equation (1)) are set to 19 and 33, respectively. The cost from 8 symmetric scanlines along horizontal, vertical, and diagonal directions are accumulated to compute the disparity based on the WTA strategy, which is then further refined using a left-right consistency check.



**Figure 4.** Visual comparison on KITTI-2012 data. Two test cases are displayed in subfigure (a) and (b). In each subfigure, the disparity maps from the ground truth, GA-Net-PyramidID+SPN and GA-Net are displayed from left to right in the first row. The second row provides the master epipolar image and the corresponding error map of each model. Regions where the proposed algorithm outperforms GA-Net are marked with red arrows.

#### 4.2.1. Aerial Stereo Data

Nowadays, most state-of-the-art dense matching algorithms are data-driven deep neural networks [8–10,12,41–43]. The high performance usually originates from a thorough training, for which a synthetic dataset is preferred for an initial learning phase, to avoid time-consuming data collection and annotation. In the field of remote sensing, nevertheless, a well-annotated stereo dataset is scarce. For example, the aerial image matching benchmark [57,58] provides reference data using LiDAR measurement. However, each algorithm is finally evaluated by the median of the DSM estimation from all the evaluated approaches, due to the limited accuracy of the reference data. Therefore, we propose a synthetic dataset, which is designed specifically for airborne and satellite stereo tasks. The dataset focuses on urban regions via referring to six city models provided by the software CityEngine: Paris, Venice, New York, Philadelphia, and two small development scenes. The models were exported and processed in Blender to preserve the textures and relevant information. Afterwards, we used BlenderProc [59] to render the dataset according to the geometry of the model which included RGB images and the corresponding disparity maps. Considering both aerial and satellite platforms, the simulated camera for rendering was located at 200 m and 500 km above the cities, respectively. A total of 854 stereo pairs in size of  $1024 \times 1024$  pixels were generated, with the ground sampling distance (GSD) ranging from 5 cm to 50 cm.

Regarding our real aerial data, we use the 4K sensor system mounted on a helicopter for the data collection [60]. Three off-the-shelf Canon EOS cameras (one 1D-C and two 1D-X) constitute the imaging unit. The data contain geo-referenced images with a size of 17.9 megapixels, acquired over Gilching in the southwest of Munich, Germany. Equipped with 50-mm lenses looking in varying view directions, a field of view (FOV) up

to 104° is reached. The flight height was 500 m above ground, enabling 6.9-cm nadir GSD. A multi-view stereo matching based on SGM was applied, in which the calculated heights (depths) from multiple highly overlapped images were fused to achieve a high-quality digital surface model (DSM). The DSM was used to compute disparity maps for each stereo pair, which were utilized as reference data for finetuning and evaluation.

#### 4.2.2. Visualization and Evaluation on Aerial Stereo Data

In Table 3, the performance of each algorithm is recorded. We firstly find that all the GA-Net models outperform the baseline SGM by a certain margin. Moreover, our pyramidal revision leads to a very small accuracy decrease compared with the original structure, but highly improves the efficiency. Our GA-Net-PyramidED (without SPN added) is the best performing pyramid model, which is only around 1% worse than GA-Net in accuracy. Nevertheless, the pyramid models are about 8 and 7 times faster than GA-Net, by only expending around 25% and 40% memory usage for training and prediction, respectively. It should be noted that for airborne data, SPN cannot improve the performance for either of the pyramid models, which is different from the close-range experiments. A visual comparison among the methods is provided in Figure 5.

**Table 3.** Accuracy and efficiency comparison between GA-Net-Pyramid, including GA-Net-PyramidED and GA-Net-PyramidID, and GA-Net on aerial data (baseline model: SGM).

	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
GA-Net-PyramidED	77.28%	86.19%	89.70%	<b>7124 M</b>	25.18 h	<b>5623 M</b>	<b>83.60 s</b>
GA-Net-PyramidED+SPN	74.06%	86.08%	89.69%	7238 M	26.19 h	<b>5623 M</b>	89.08 s
GA-Net-PyramidID	76.35%	85.46%	89.14%	7544 M	<b>20.59 h</b>	6979 M	84.02 s
GA-Net-PyramidID+SPN	76.14%	84.82%	88.21%	7676 M	21.54 h	6979 M	86.19 s
GA-Net	<b>78.75%</b>	<b>86.99%</b>	<b>90.13%</b>	30,512 M	187.59 h	15,685 M	616.74 s
SGM	72.14%	75.89%	77.15%	—	—	—	—

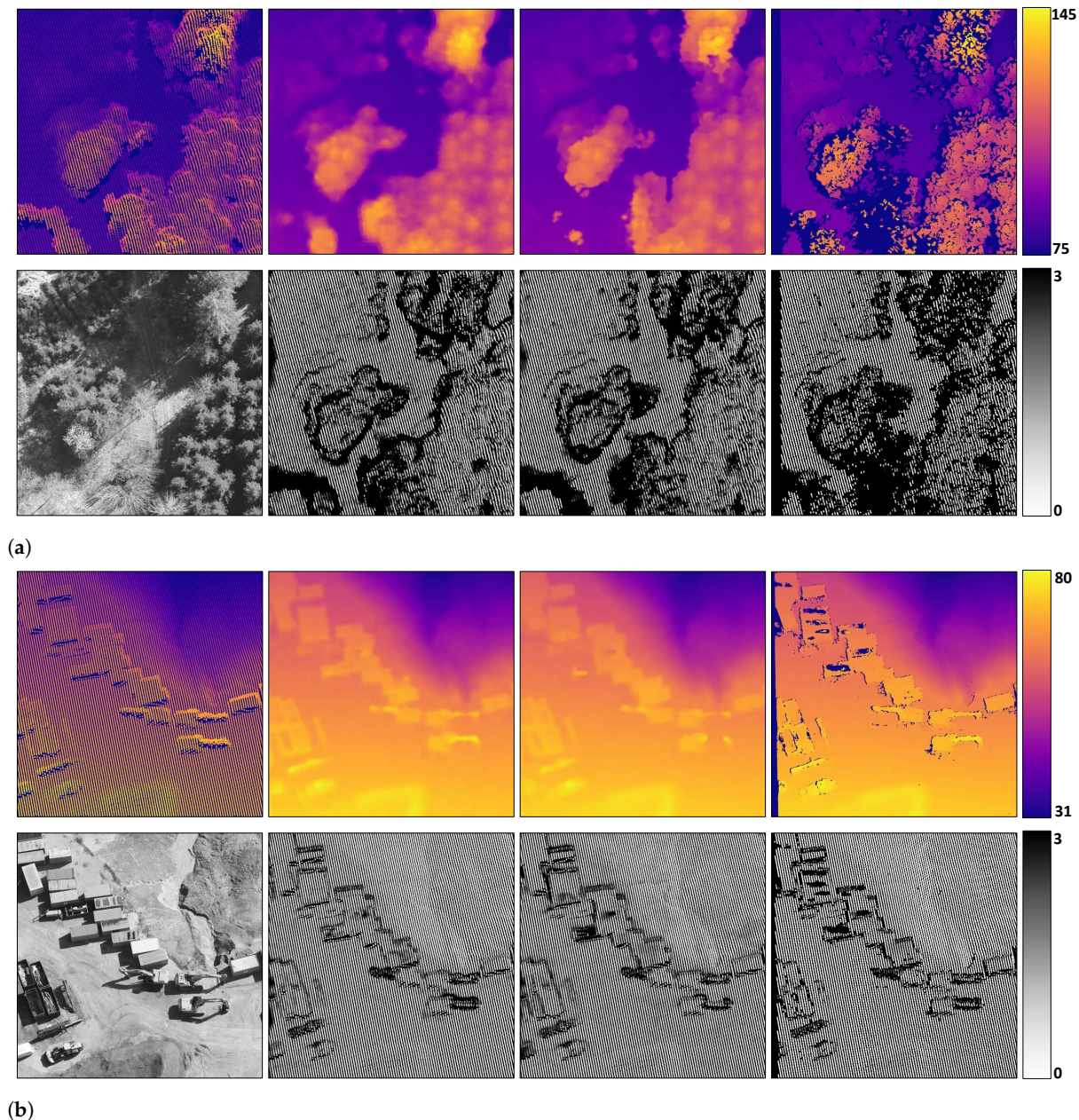
<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

We select two regions, one vegetation and one building area from the test data for the visualization. It is shown that GA-Net-PyramidED archives good performance in airborne stereo matching. When the scene is relatively simple, containing fewer depth discontinuities and a smooth depth change, the hierarchical estimation and refinement of disparity is capable of highly enhancing the efficiency, without a noteworthy sacrifice of the result's quality.

#### 4.2.3. Pyramid Setting

To further understand our GA-Net-Pyramid when applied in the field of remote sensing, we explore the impact of different pyramid architectures using our aerial data. Regarding the pyramid structure, two variants are the most important factors, the number of pyramid levels and the residual search range for disparity refinement. The main difference between GA-Net-PyramidED and GA-Net-PyramidID is the strategy to extract features, which is not directly related to the above two factors. In addition, our two pyramid models achieve similar accuracy. Therefore, we select GA-Net-PyramidED without SPN for post-processing to study the pyramid setting, since it is the more intuitive pyramidal modification of GA-Net. As for the number of pyramid levels, we start from 2, since a 1-level GA-Net-Pyramid will degenerate to GA-Net, to 4 levels, with a fixed residual range  $[-6, +6]$ . The model is trained on our synthetic dataset from scratch and evaluated on the same test data. We use the same hyperparameter setting as before, except that the size of the training patches changes to  $384 \times 768$  to facilitate the downsampling when more levels are applied. We train the model on one GPU card due to the less memory requirement of GA-Net-Pyramid. The results are in Table 4.

According to the table, it is found that the architecture with 4 pyramid levels acquires the best efficiency. However, with slightly increased memory and runtime, the model with 3 pyramid levels achieves better results. Along with GA-Net-PyramidED regresses towards GA-Net (from 3 to 2 levels), the efficiency drastically deteriorates as expected, nevertheless, without a noticeable improvement of the accuracy. Therefore, we determine to use the number of pyramid levels as 3. Then, we adjust the residual search range to  $[-3, +3]$ ,  $[-6, +6]$  and  $[-12, +12]$ , respectively. The model is also trained from scratch on our synthetic dataset using one GPU card, and tested on the same 20 aerial images. We keep the training setting unchanged, except that the patch size is set back to  $384 \times 576$ . In Table 5, the performance for different residual search ranges is recorded.



**Figure 5.** Visual comparison on aerial data. Two test cases regarding vegetation and building area are displayed in subfigure (a) and (b), respectively. In each subfigure, the reference disparity map and the stereo results from GA-Net-PyramidED, GA-Net and SGM are displayed from left to right in the first row. The second row provides the master epipolar image and the corresponding error map of each model.

**Table 4.** Accuracy and efficiency comparison for GA-Net-PyramidED with different pyramid levels.

Pyramid Levels	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
2	<b>72.38%</b>	80.89%	85.14%	11521M	70.25h	5813M	120.28s
3	72.17%	<b>81.22%</b>	<b>85.69%</b>	8121M	29.13h	5623M	82.11s
4	72.08%	81.19%	85.57%	<b>7647M</b>	<b>27.80h</b>	<b>5589M</b>	<b>63.92s</b>

<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

**Table 5.** Accuracy and efficiency comparison for GA-Net-PyramidED with different residual search ranges.

Residual Range	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
[−3, +3]	73.38%	81.95%	86.04%	<b>5941 M</b>	<b>23.49 h</b>	<b>5467 M</b>	<b>55.23 s</b>
[−6, +6]	<b>73.76%</b>	<b>82.21%</b>	<b>86.40%</b>	6283 M	26.35 h	5623 M	84.50 s
[−12, +12]	73.38%	82.11%	86.37%	7033 M	34.96 h	6489 M	123.09 s

<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

Table 5 indicates that as the residual range becomes larger, the efficiency naturally decreases. Moreover, when the residual buffer expands over [−6, +6], the accuracy cannot be further enhanced. Hence, the structure of our pyramid is determined as 3 levels, with the maximum/minimum residual set as 6/−6. To keep the experiments consistent, the pyramid structure is used for both GA-Net-PyramidED and GA-Net-PyramidID in this paper.

#### 4.3. Experiments on Satellite Stereo Data

The flight campaign regarding our aerial 4K images was performed during a WorldView-3 stereo acquisition of the same area [61]. Due to the minimal time difference of less than 1 hour of each aerial image from the satellite images, the higher resolution airborne data are well suited as reference data for the satellite stereo matching to finetune the models and evaluate the results. This is a notable improvement over other satellite stereo datasets [17,62], which do not provide sub-pixel disparity accuracy due to different sensing modalities and scene changes due to time difference between the image and ground truth acquisition. In contrast, the data used in this article allow reliable evaluation for 1- and 2-pixel accuracy metrics. This is especially important for photogrammetry and remote sensing, as many applications require highly precise elevation measurements.

Similar to Section 4.2, the networks are pre-trained on our synthetic remote sensing data for 200 epochs, and finetuned on the generated satellite training data for 150 epochs. The training conditions stay the same, including the patch size (384 × 576), disparity range ([0, 192]), batch size (2), GPU usage (2 Quadro P6000 cards), etc. SGM is also tested for reference.

##### 4.3.1. Satellite Stereo Data

WorldView-3 is a very-high-resolution imaging satellite currently offering the most detailed publicly available spaceborne imagery, at a resolution of 30 cm. After bundle-adjustment of the data with the 4K aerial imagery and DSM as reference, we generated an epipolar rectified stereo pair using the algorithm implemented by the CARS stereo pipeline [63]. Similar to the aerial imagery, a reference disparity map was calculated by projecting each point of the 4K DSM into the epipolar satellite stereo pair. The stereo pair has a dimension of 20,815 × 28,264 pixels, which was cut into 98 tiles (in size of 1152 × 1152) owning an overlap larger than 25% with the 4K data coverage. From them, 78 tiles were randomly selected for finetuning the pre-trained GA-Net models, with the other 20 image pairs as the test data.

As the airborne data were geo-referenced in two separate blocks using differential GPS and only few ground control points (GCPs), a slight height offset was found between the aerial and satellite data, yielding disparity differences between the aerial reference and the satellite stereo pair in the pixel range, but rising up to 4 pixels at the corner of one aerial block. Since these systematic differences strongly affected training and evaluation of the networks, a second-order offset surface was fitted to the difference of the airborne reference disparity map and the satellite disparity map estimated by SGM, on each of the 98 tiles. The offset was added to the reference disparity map to alleviate the systematic bias which was reduced from 0.97 to 0.51 pixels.

#### 4.3.2. Visualization and Evaluation on Satellite Stereo Data

In Table 6, we record the performance of GA-Net-Pyramid, GA-Net and SGM. Similar to the results of airborne data, GA-Net achieves the highest accuracy, after which GA-Net-PyramidED still acquires the best performance among all the other models. The 1-pixel accuracy of our GA-Net-PyramidED, without SPN added for post-processing, is only surpassed by GA-Net by 0.08%. However, the former is around 8 and 13 times faster than the latter, consuming only 23% and 36% GPU memory in training and test, respectively. In addition, GA-Net-PyramidED performs better than GA-Net\_PyramidID, with less GPU memory consumption but longer training time. SPN also impairs the performance of the pyramid models which is consistent with our experiments on aerial data. The visual comparison is in Figure 6, including a vegetation and a building area as well. It is found that both networks predict a smoother disparity map than SGM, with less erroneous estimation. Moreover, similar results are obtained between our GA-Net-PyramidED and GA-Net, considering the reconstruction density and quality.

**Table 6.** Accuracy and efficiency comparison between GA-Net-Pyramid, including GA-Net-PyramidED and GA-Net-PyramidID, and GA-Net on satellite data (baseline model: SGM).

	Accuracy			Training Efficiency		Test Efficiency	
	1 pix	2 pix	3 pix	Memory	Runtime	Memory	Runtime
GA-Net-PyramidED	83.76%	90.70%	93.00%	<b>7144 M</b>	23.77 h	<b>5623 M</b>	<b>31.53 s</b>
GA-Net-PyramidED+SPN	82.99%	91.05%	93.34%	7250 M	24.56 h	<b>5623 M</b>	35.93 s
GA-Net-PyramidID	81.45%	89.58%	92.40%	7558 M	<b>19.11 h</b>	6979 M	33.11 s
GA-Net-PyramidID+SPN	80.66%	89.10%	92.00%	7700 M	20.27 h	6979 M	32.87 s
GA-Net	<b>83.84%</b>	<b>91.42%</b>	<b>93.74%</b>	30,514 M	179.19 h	15,685 M	401.91 s
SGM	79.98%	82.74%	83.32%	—	—	—	—

<sup>1</sup> Bold font means the best accuracy/efficiency in each group.

#### 4.3.3. Stereo Processing over Mountain Area

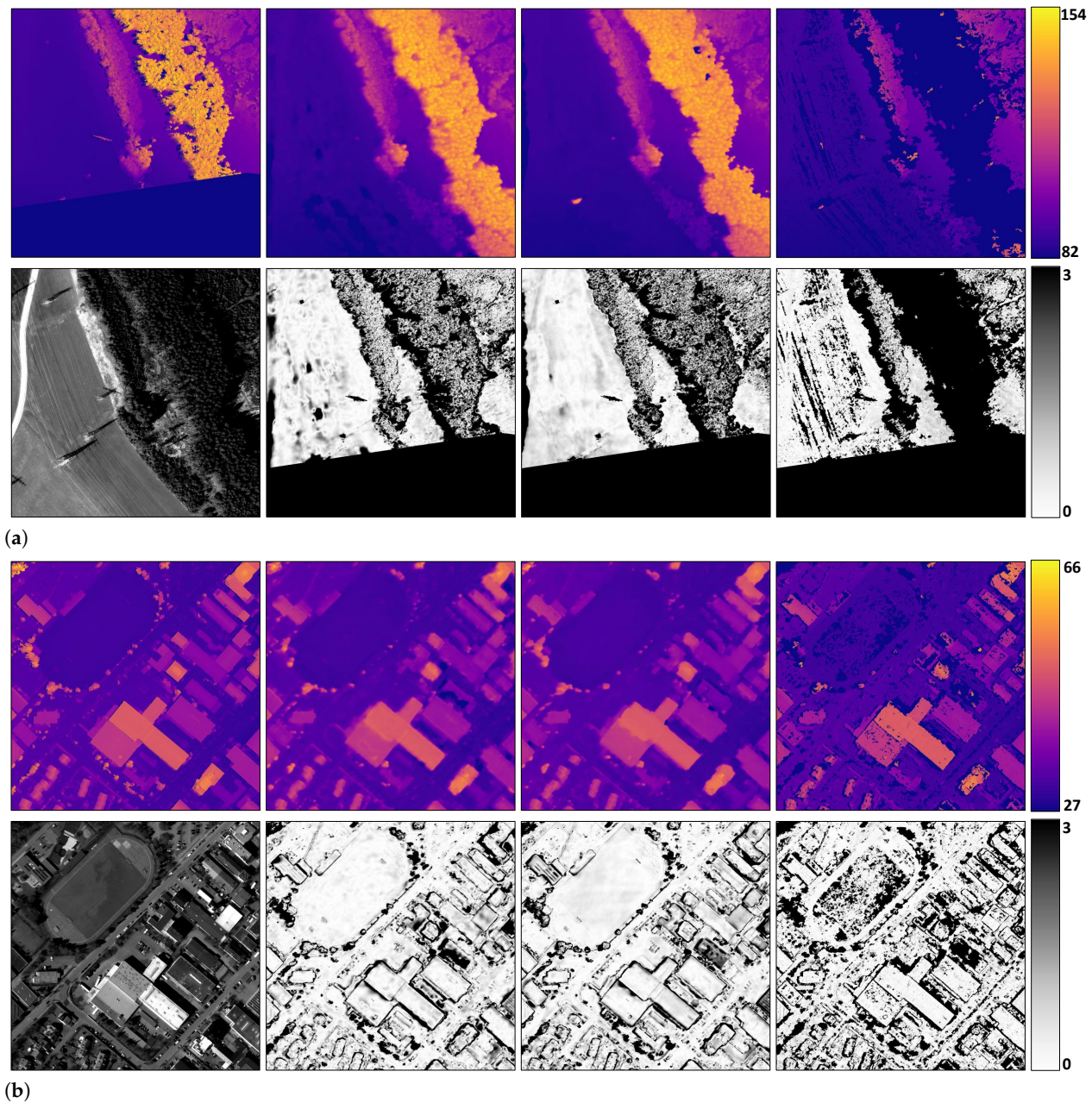
In this section, we apply our pyramid network on a stereo pair with a large disparity range, in order to indicate the model's ability to process large-scale remote sensing data. The imagery is from WorldView-2 [64] at a resolution of 50 cm, covering the Matterhorn mountain, Switzerland. We select a stereo pair with 14° conversion angle for which the disparity varies in range of thousand pixels, due to the very large ground height difference from 1800 m to 4478 m. The best performing model finetuned in our previous satellite experiments, GA-Net-PyramidED, is directly used for disparity prediction in this test. Regarding the evaluation, we follow our processing chain in Section 4.3.1, using an aerial dataset with good stereo geometry to the same area to generate reference data. The test region, the reference disparity map, and our stereo results are displayed in Figure 7.

The mountain peak is located at the center of the image with a disparity up to around 1250 pixels; thus, we set the disparity range as [0, 1248]. Note that the model we use receives no supervision and knowledge regarding the mountain area with that large disparity difference. However, we achieve a 3-pixel accuracy of 87.34%. There are temporal inconsistencies between the satellite and reference data, leading to varying snow cover. Therefore, we use 3-pixel as the threshold. The visual comparison shows very similar results

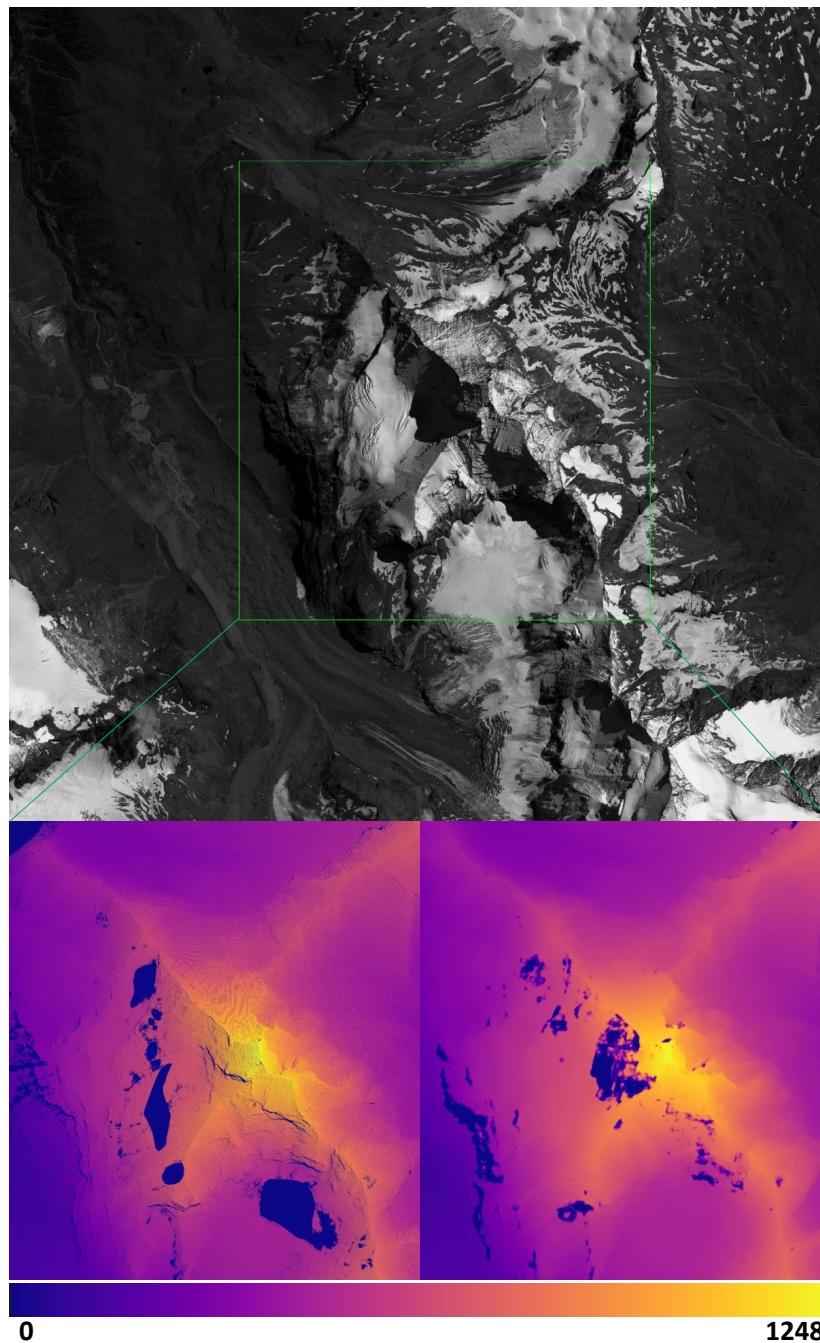


between our disparity prediction and the reference, considering the reconstruction density, smoothness, etc. Disparity holes are found from certain regions in our results. According to the image content, the regions are in shadow with limited texture information, where the network suffers from collecting enough information to locate the correspondences.

In the test period, the patch in size of  $768 \times 6912$  is fed to the network for disparity prediction. Considering the disparity range  $[0, 1248]$ , GA-Net will theoretically need more than 200 GB GPU memory to process the same data. Our GA-Net-PyramidED, however, consumes only around 20 GB.



**Figure 6.** Visual comparison on satellite data. Two test cases regarding vegetation and building area are displayed in subfigure (a) and (b), respectively. In each subfigure, the reference disparity map and the stereo results from GA-Net-PyramidED, GA-Net and SGM are displayed from left to right in the first row. The second row provides the master epipolar image and the corresponding error map of each model.



**Figure 7.** A showcase to indicate the ability of our pyramid network in processing remote sensing stereo pair with large baseline. The test image and the corresponding stereo reconstruction from the reference disparity map (lower left) and our pyramid model (lower right) are shown. The reconstructed region is highlighted by the green rectangle with a size of  $19,791 \times 15,639$  pixels. Test region: Matterhorn mountain, Switzerland. Test model: GA-Net-PyramidED.

## 5. Discussion

Based on a pyramid architecture, our GA-Net-Pyramid is able to roughly estimate the depth from a downsampled feature, and then refine the prediction level by level until the original resolution is recovered. Thus, the efficiency is significantly enhanced with the accuracy maintained to be comparable with GA-Net on remote sensing datasets. Some technical details are found below.

We firstly propose GA-Net-PyramidED which applies the GA-Net model hierarchically. In our experiments on airborne and satellite data, it is demonstrated that GA-Net-

PyramidED is able to achieve similar results as GA-Net, nevertheless, consuming much less GPU memory and runtime for both training and prediction. Considering that only the pyramid top exploits the absolute disparity range in low resolution to locate the stereo correspondence, GA-Net-PyramidED is capable of processing stereo pairs with wider baselines if the same GPU memory for GA-Net is available. This is particularly suitable to process large stereo pairs with high-disparity search ranges in the field of remote sensing, which usually triggers the bottleneck of most memory-hungry deep neural networks. On the other hand, the aerial/satellite images mainly focus on large-scale landscapes such as city areas, for which the local object heights/depths are generally smoother and regular with fewer occlusions, depth discontinuities, fine structures, etc., compared with the close-range datasets. Thus, the results from the previous pyramid level can better guide the disparity estimation on the current level. When a large height variance exists within the scene, e.g., in mountain areas, a rough depth prediction from lower resolution pyramid level is effective to limit the search range and avoid influence from ambiguous disparity candidates for higher resolution level.

Another architecture is designed as GA-Net-PyramidID, which implicitly downsamples the input stereo pair via a U-Net feature extractor to feed each pyramid level using the intermediate feature map of its decoder. Concerning the close-range datasets, especially for Scene Flow that contains very complex and non-logical scene structures, both GA-Net-PyramidED and GA-Net-PyramidID are not competitive with GA-Net (GA-Net-PyramidID+SPN performs the best among all the pyramid models). The accuracy could be influenced when details are possibly omitted by the low-resolution level. Moreover, the residual search range may not support refinement for regions with rapid depth changes and discontinuities. Although GA-Net outperforms the proposed pyramid approaches on both close-range datasets, Scene Flow and KITTI, the performance difference is smaller for the real-world KITTI 2012 data.

SPN is applied on image segmentation to refine the object boundaries. In our experiments on close-range data, better depth estimation is achieved by our pyramid networks with SPN added, especially for GA-Net-PyramidID. However, it is found that negative influence from SPN occurs on airborne and satellite data, for both GA-Net-PyramidED and GA-Net-PyramidID. The reason is that the resolution of aerial/satellite data is relatively low, with fewer details and depth discontinuities included; thus, the strength of SPN is not embodied. More importantly, the training of SPN cannot be well supervised, considering that the number of valid training patches from airborne (987 millions) and satellite (934 millions) datasets is far less than the close-range datasets (18 billions). The condition to collect reference data is not as ideal as close-range scenarios using precise LiDAR scanning, structured light or synthetic labeling. In addition, SPN essentially refers to the input to improve the output, which are the master epipolar image and the disparity result in our case, respectively. The natural land texture and shadows, which are not necessarily related to ground height variation, may confuse SPN to locate the correct depth borders. The slightly changing and rolling ground height, e.g., in natural regions, could confuse the disparity post-processing as indicated by the lower 1-pixel accuracy.

## 6. Conclusions

Nowadays, the rapid development of deep learning and CNNs has made the technique dominate in the field of dense matching, leading to a sequence of high-rank algorithms in different close-range benchmarks. Compared to conventional approaches, the depth estimation for ill-posed areas, e.g., textureless regions, occlusions, etc., is better accomplished resulting in a considerable improvement. However, a large amount of well-annotated data and a time-consuming training are usually required before a network reaches high performance. In the field of remote sensing, a huge amount of high-definition data is supplied by unmanned aerial vehicles, helicopters, airplanes or satellites at all times. The data cover large areas with varying stereo baselines and image sizes of up to multiple gigapixels. Hence, a well-performed deep network from the field of computer vision would

struggle to process the remote sensing data, under a certain time and memory budget. Since that stereo datasets with reliable ground truth are not available in remote sensing, we build a dataset consisting of simultaneously acquired 30-cm satellite and 6-cm aerial imagery which are co-registered to sub-pixel disparity precision. The experimental results demonstrate that our proposed model can largely enhance the efficiency in training and test, while maintaining a comparable accuracy. The test on a satellite stereo pair over Matterhorn specifically highlights the significance of our method for processing large baseline stereo data.

We suggest to use GA-Net-PyramidED for remote sensing stereo processing. With slightly increased runtime, GA-Net-PyramidED produces better depth results than GA-Net-PyramidID, while consuming less GPU memory. As for the close-range dataset, GA-Net-PyramidID with an SPN module to enhance the depth borders is preferred. Regarding the effect of SPN, it is demonstrated that a minor improvement is obtained on close-range data; nevertheless, the depth estimation could be impaired using SPN in case of remote sensing data, especially when the reference data own limited quantity or quality for training.

In future research, more reference data should be collected for urban, rural and mountainous scenarios for remote sensing, in order to better supervise a learning-based model in stereo prediction. Thus, we can better handle the ill-posed regions in shadows, depth boundaries, etc., and obtain high-quality geographical measurements for earth observation.

**Author Contributions:** Conceptualization, Y.X.; data curation, Y.X., P.d. and M.F.R.; funding acquisition, J.T. and P.R.; investigation, Y.X., P.d., F.F., J.T., M.F.R. and P.R.; methodology, Y.X.; supervision, P.d., F.F., J.T. and P.R.; validation, Y.X.; visualization, Y.X.; writing—original draft, Y.X.; writing—review and editing, P.d., F.F., J.T., M.F.R. and P.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by “ForDroughtDet” project (FKZ: 22WB410602), from the Waldklimafonds, under joint leadership of Bundeslandwirtschafts (BMEL) and Bundesumweltministerium (BMU). Yuanxin Xia is funded by a DLR-DAAD Research Fellowship (No. 57265855).

**Data Availability Statement:** The Scene Flow dataset can be accessed in <https://lmb.informatik.uni-freiburg.de/resources/datasets/SceneFlowDatasets.en.html/>. The KITTI-2012 dataset can be accessed in [http://www.cvlibs.net/datasets/kitti/eval\\_stereo\\_flow.php?benchmark=stereo/](http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo/).

**Acknowledgments:** We are indebted to University of Freiburg, Karlsruhe Institute of Technology, and Toyota Technological Institute at Chicago for providing the close-range benchmark datasets. We would like to thank Franz Kurz from the German Aerospace Center (DLR) for providing the aerial data, and DigitalGlobe and European Space Imaging (EUSI) for providing the satellite data used in the research.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

GA-Net	Guided Aggregation Network
GA-Net-Pyramid	GA-Net based on a pyramid architecture
GA-Net-PyramidED	GA-Net-Pyramid with Explicit Downsampling
GA-Net-PyramidID	GA-Net-Pyramid with Implicit Downsampling
SGM	Semi-Global Matching

## References

1. Hirschmüller, H. Semi-global Matching—Motivation, Developments and Applications. In *Photogrammetric Week*; Wichmann Verlag: Heidelberg, Germany, 2011; Volume 11, pp. 173–184.
2. Kuschik, G.; d’Angelo, P.; Qin, R.; Poli, D.; Reinartz, P.; Cremers, D. DSM Accuracy Evaluation for the ISPRS Commission I Image Matching Benchmark. In *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; 2014; Volume XL-1, pp. 95–200. <http://doi.org/10.5194/isprsarchives-XL-1-195-2014>.
3. Qin, R.; Huang, X.; Gruen, A.; Schmitt, G. Object-based 3-D building change detection on multitemporal stereo images. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2015**, *8*, 2125–2137. <http://doi.org/10.1109/JSTARS.2015.2424275>.

4. Xia, Y.; d'Angelo, P.; Tian, J.; Fraundorfer, F.; Reinartz, P. Self-supervised convolutional neural networks for plant reconstruction using stereo imagery. *Photogramm. Eng. Remote. Sens.* **2019**, *85*, 389–399.
5. Bleyer, M.; Breiteneder, C. Stereo matching—State-of-the-art and research challenges. In *Advanced Topics in Computer Vision*; Farinella, G.M., Battiato, S., Cipolla, R., Eds.; Springer: London, UK, 2013; pp. 143–179. [http://doi.org/10.1007/978-1-4471-5520-1\\_6](http://doi.org/10.1007/978-1-4471-5520-1_6).
6. Hirschmüller, H. Accurate and Efficient Stereo Processing by Semi-global Matching and Mutual Information. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–26 June 2005; Volume 2, pp. 807–814. <http://doi.org/10.1109/CVPR.2005.56>.
7. Rothermel, M. Development of a SGM-Based Multi-View Reconstruction Framework for Aerial Imagery. Ph.D. Thesis, University of Stuttgart, Stuttgart, Germany, 2017.
8. Zhang, F.; Prisacariu, V.; Yang, R.; Torr, P.H. GA-Net: Guided Aggregation Net for End-to-End Stereo Matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 185–194.
9. Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, P.; Kennedy, R.; Bachrach, A.; Bry, A. End-to-End Learning of Geometry and Context for Deep Stereo Regression. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 66–75. <http://doi.org/10.1109/ICCV.2017.17>.
10. Chang, J.; Chen, Y. Pyramid Stereo Matching Network. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 5410–5418. <http://doi.org/10.1109/CVPR.2018.00567>.
11. Tonioni, A.; Tosi, F.; Poggi, M.; Mattoccia, S.; Stefano, L.D. Real-Time Self-Adaptive Deep Stereo. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 195–204. <http://doi.org/10.1109/CVPR.2019.00028>.
12. Wang, Y.; Lai, Z.; Huang, G.; Wang, B.H.; van der Maaten, L.; Campbell, M.; Weinberger, K.Q. Anytime Stereo Image Depth Estimation on Mobile Devices. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 5893–5900.
13. Yang, G.; Manela, J.; Happold, M.; Ramanan, D. Hierarchical Deep Stereo Matching on High-Resolution Images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5510–5519. <http://doi.org/10.1109/CVPR.2019.00566>.
14. Liu, S.; De Mello, S.; Gu, J.; Zhong, G.; Yang, M.H.; Kautz, J. Learning Affinity via Spatial Propagation Networks. In *Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30, pp. 1520–1530.
15. Mayer, N.; Ilg, E.; Häusser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; Brox, T. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4040–4048.
16. Geiger, A.; Lenz, P.; Urtasun, R. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 3354–3361. <http://doi.org/10.1109/CVPR.2012.6248074>.
17. Le Saux, B.; Yokoya, N.; Hansch, R.; Brown, M.; Hager, G.; Kim, H. 2019 IEEE GRSS data fusion contest: Semantic 3D reconstruction [Technical Committees]. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 103–105. <http://doi.org/10.1109/MGRS.2019.2893783>.
18. d'Angelo, P.; Reinartz, P. Semiglobal Matching Results on the ISPRS Stereo Matching Benchmark. In *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; ISPRS: Hanover, Germany, 2011; Volume XXXVIII-4/W19, pp. 79–84. <http://doi.org/10.5194/isprsarchives-XXXVIII-4-W19-79-2011>.
19. d'Angelo, P. Improving Semi-global Matching: Cost Aggregation and Confidence Measure. In *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; ISPRS: Prague, Czech Republic, 2016; Volume XLI-B1, pp. 299–304. <http://doi.org/10.5194/isprs-archives-XLI-B1-299-2016>.
20. Facciolo, G.; de Franchis, C.; Meinhardt, E. MGM: A Significantly More Global Matching for Stereovision. In *Proceedings of the British Machine Vision Conference (BMVC)*; Xie, X., Tam, G.K.L., Eds.; BMVA Press: Swansea, UK, 2015; pp. 90.1–90.12. <http://doi.org/10.5244/C.29.90>.
21. Geman, S.; Geman, D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *PAMI-6*, 721–741.
22. Pollard, S.B.; Mayhew, J.E.W.; Frisby, J.P. PMF: A stereo correspondence algorithm using a disparity gradient limit. *Perception* **1985**, *14*, 449–470, [<https://doi.org/10.1068/p140449>]. PMID: 3834387, <http://doi.org/10.1068/p140449>.
23. Barnard, S. Stochastic stereo matching over scale. *Int. J. Comput. Vis.* **1989**, *3*, 17–32.
24. Kolmogorov, V.; Zabih, R. Computing Visual Correspondence with Occlusions using Graph Cuts. In Proceedings of the Proceedings Eighth IEEE International Conference on Computer Vision (ICCV), Vancouver, BC, Canada, 7–14 July 2001; Volume 2, pp. 508–515. <http://doi.org/10.1109/ICCV.2001.937668>.
25. Boykov, Y.; Kolmogorov, V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 1124–1137.
26. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. <http://doi.org/10.1109/5.726791>.
27. Zbontar, J.; LeCun, Y. Stereo matching by training a convolutional neural network to compare image patches. *J. Mach. Learn. Res.* **2016**, *17*, 1–32.

28. Bromley, J.; Bentz, J.; Bottou, L.; Guyon, I.; LeCun, Y.; Moore, C.; Sackinger, E.; Shah, R. Signature verification using a “siamese” time delay neural network. *Int. J. Pattern Recognit. Artif. Intell.* **1993**, *7*, 25.
29. Luo, W.; Schwing, A.G.; Urtasun, R. Efficient Deep Learning for Stereo Matching. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 5695–5703. <http://doi.org/10.1109/CVPR.2016.614>.
30. Seki, A.; Pollefeys, M. Sgm-nets: Semi-global Matching with Neural Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6640–6649. <http://doi.org/10.1109/CVPR.2017.703>.
31. Michael, M.; Salmen, J.; Stallkamp, J.; Schlipsing, M. Real-time Stereo Vision: Optimizing Semi-global Matching. In Proceedings of the 2013 IEEE Intelligent Vehicles Symposium (IV), Gold Coast City, Australia, 23 June 2013; pp. 1197–1202. <http://doi.org/10.1109/IVS.2013.6629629>.
32. Poggi, M.; Mattoccia, S. Learning a General-purpose Confidence Measure based on O(1) Features and a Smarter Aggregation Strategy for Semi Global Matching. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 509–518. <http://doi.org/10.1109/3DV.2016.61>.
33. Schönberger, J.L.; Sinha, S.N.; Pollefeys, M. Learning to Fuse Proposals from Multiple Scanline Optimizations in Semi-global Matching. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 758–775.
34. Xia, Y.; d’Angelo, P.; Tian, J.; Fraundorfer, F.; Reinartz, P. Multi-label learning based semi-global matching forest. *Remote Sens.* **2020**, *12*, 1069. <http://doi.org/10.3390/rs12071069>.
35. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Häusser, P.; Hazirbas, C.; Golkov, V.; v. d. Smagt, P.; Cremers, D.; Brox, T. FlowNet: Learning Optical Flow with Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 2758–2766.
36. Guo, X.; Yang, K.; Yang, W.; Wang, X.; Li, H. Group-Wise Correlation Stereo Network. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3268–3277. <http://doi.org/10.1109/CVPR.2019.00339>.
37. Zhu, Z.; Guo, W.; Chen, W.; Li, Q.; Zhao, Y. MPANet: Multi-Scale Pyramid Aggregation Network For Stereo Matching. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 2773–2777. <http://doi.org/10.1109/ICIP42928.2021.9506705>.
38. Xu, H.; Zhang, J. AANet: Adaptive Aggregation Network for Efficient Stereo Matching. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1956–1965. <http://doi.org/10.1109/CVPR42600.2020.00203>.
39. Wang, H.; Fan, R.; Cai, P.; Liu, M. PVStereo: Pyramid voting module for end-to-end self-supervised stereo matching. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4353–4360. <http://doi.org/10.1109/LRA.2021.3068108>.
40. Stucker, C.; Schindler, K. ResDepth: Learned Residual Stereo Reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 13–19 June 2020.
41. Wu, Z.; Wu, X.; Zhang, X.; Wang, S.; Ju, L. Semantic Stereo Matching with Pyramid Cost Volumes. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October to 2 November 2019; pp. 7483–7492. <http://doi.org/10.1109/ICCV.2019.00758>.
42. Song, X.; Zhao, X.; Fang, L.; Hu, H.; Zhou Yu, Y. EdgeStereo: An effective multi-task learning network for stereo matching and edge detection. *Int. J. Comput. Vis.* **2020**, *128*, 910–930.
43. Cheng, X.; Wang, P.; Yang, R. Learning depth with convolutional spatial propagation network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 2361–2379.
44. Zhang, F.; Qi, X.; Yang, R.; Prisacariu, V.; Wah, B.; Torr, P. Domain-invariant Stereo Matching Networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2019.
45. Zhang, Y.; Chen, Y.; Bai, X.; Yu, S.; Yu, K.; Li, Z.; Yang, K. Adaptive Unimodal Cost Volume Filtering for Deep Stereo Matching. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 7 February–12 February 2020.
46. Cheng, X.; Zhong, Y.; Harandi, M.; Dai, Y.; Chang, X.; Li, H.; Drummond, T.; Ge, Z. Hierarchical Neural Architecture Search for Deep Stereo Matching. In Proceedings of the Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H., Vancouver, Canada, 6–12 December 2020.
47. Song, X.; Yang, G.; Zhu, X.; Zhou, H.; Wang, Z.; Shi, J. AdaStereo: A Simple and Efficient Approach for Adaptive Stereo Matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021. Nashville, TN, USA, 19–25 June 2021.
48. Chang, J.R.; Chang, P.C.; Chen, Y.S. Attention-Aware Feature Aggregation for Real-time Stereo Matching on Edge Devices. In Proceedings of the Asian Conference on Computer Vision (ACCV), Kyoto, Japan, 30 November–December 4 2020.
49. Gu, X.; Fan, Z.; Zhu, S.; Dai, Z.; Tan, F.; Tan, P. Cascade Cost Volume for High-Resolution Multi-View Stereo and Stereo Matching. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 2492–2501. <http://doi.org/10.1109/CVPR42600.2020.00257>.

50. Hu, Y.; Wang, W.; Yu, H.; Zhen, W.; Scherer, S. ORStereo: Occlusion-Aware Recurrent Stereo Matching for 4K-Resolution Images. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021.
51. Hirschmüller, H. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 328–341. <http://doi.org/10.1109/TPAMI.2007.1166>.
52. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 234–241.
53. Scharstein, D.; Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vis.* **2002**, *47*, 7–42. <http://doi.org/10.1023/A:1014573219977>.
54. Springenberg, J.; Dosovitskiy, A.; Brox, T.; Riedmiller, M. Striving for Simplicity: The All Convolutional Net. International Conference on Learning Representations (ICLR, workshop track). 2016. arXiv:1606.04038.
55. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. 2014. arXiv:1412.6980.
56. Zabih, R.; Woodfill, J. Non-parametric Local Transforms for Computing Visual Correspondence. In *Computer Vision—ECCV’94*; Eklundh, J.O., Ed.; Springer Berlin Heidelberg: Berlin, Heidelberg, 1994; pp. 151–158.
57. Haala, N. *The Landscape of Dense Image Matching Algorithms*; Wichmann/VDE: Belin/Offenbach, Germany, 2013.
58. Haala, N. Dense image matching final report. *Eurosdrr Publ. Ser. Off. Publ.* **2014**, *64*, 115–145.
59. Denninger, M.; Sundermeyer, M.; Winkelbauer, D.; Zidan, Y.; Olefir, D.; Elbadrawy, M.; Lodhi, A.; Katam, H. BlenderProc. **2019** arXiv preprint. arXiv:1911.01911 .
60. Kurz, F.; Rosenbaum, D.; Meynberg, O.; Mattyus, G.; Reinartz, P. Performance of a Real-Time Sensor and Processing System on a Helicopter. In *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; ISPRS: Denver, Colorado, USA, 2014; Volume XL-1, pp. 189–193. <http://doi.org/10.5194/isprsarchives-XL-1-189-2014>.
61. Hu, F.; Gao, X.; Li, G.; Li, M. DEM Extraction from WorldView-3 Stereo-images and Accuracy Evaluation. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Prague, Czech Republic, 12–19 July 2016; Volume 41.
62. Bosch, M.; Foster, K.; Christie, G.A.; Wang, S.; Hager, G.D.; Brown, M.Z. Semantic Stereo for Incidental Satellite Images. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1524–1532.
63. Michel, J.; Sarrazin, E.; Youssefi, D.; Cournet, M.; Buffe, F.; Delvit, J.M.; Emilien, A.; Bosman, J.; Melet, O.; L’Helguen, C. A New Satellite Imagery Stereo Pipeline Designed for Scalability, Robustness and Performance. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; ISPRS: Nice, France, 2020; Volume V-2-2020, pp. 171–178. <http://doi.org/10.5194/isprs-annals-V-2-2020-171-2020>.
64. Aguilar, M.A.; Bianconi, F.; Aguilar, F.J.; Fernández, I. Object-based greenhouse classification from GeoEye-1 and WorldView-2 stereo imagery. *Remote Sens.* **2014**, *6*, 3554–3582. <http://doi.org/10.3390/rs6053554>.