TECHNISCHE UNIVERSITÄT MÜNCHEN

Fakultät für Informatik
Computer Aided Medical Procedures & Augmented Reality / I16

Human Pose Estimation in Complex Environments

Vasileios Belagiannis

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. Th. Huckle
Prüfer der Dissertation:

1. Univ.-Prof. Dr. N. Navab
2. Univ.-Prof. Dr. H. Bischof, Technische Universität Graz, Österreich
3. Priv.-Doz. Dr. S. Ilic

Die Dissertation wurde am 27.05.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 16.08.2015 angenommen.

Abstract

Estimating human body poses from images is a demanding task that has attracted great interest from the computer vision community. Determining automatically the body pose promotes many applications such as human tracking, motion capture, activity recognition, surveillance and surgical workflow analysis. This work addresses the problem of human pose estimation from different perspectives. At first, we tackle the problem of single human pose estimation from a single view. Then, we move to multi-view camera systems, where we work on both single and multiple human pose estimation. In this thesis, we propose novel discriminative and generative methods to address all these problems of human pose estimation.

The primary contributions of this work are threefold. At the beginning, we propose two discriminative methods for 2D human pose estimation from a single view. The first contribution exploits Random Forests (RF), while the second builds on Deep Learning. In both cases, we formulate the problem of body pose estimation as a regression task, where the body pose is defined by a set of body joints. To build a regressor that predicts the 2D body poses, we learn a model either using a Regression Forest or Convolutional Neural Network (ConvNet). For the convolutional neural network (ConvNet), we propose a regression model that achieves robustness to outliers by minimizing Tukey's biweight function, an M-estimator robust to outliers, as the loss function of the network. In addition to the robust loss, we introduce a coarse-to-fine model, which processes input images of progressively higher resolutions for improving the accuracy of the regressed values. Our third contribution is a generative model for multi-view human pose estimation. In particular, we address the problem of 3D pose estimation of multiple humans from multiple views. This is a more challenging problem than single human pose estimation due to the much larger search space of body poses, different type of occlusions as well as across view ambiguities when not knowing the identity of the individuals in advance. To address these problems, we first create a reduced search space by triangulation of corresponding body parts obtained from 2D detectors in pairs of camera views. In order to resolve ambiguities of wrong and mixed body parts hypotheses of multiple humans after triangulation, we introduce a 3D pictorial structures (3DPS) model. Our model builds on multi-view unary potentials, while a prior model is integrated into pairwise and ternary potential functions. The model is generic and applicable to both single and multiple human pose estimation. Finally, we apply the 3D pictorial structures (3DPS) model on estimating the body pose of multiple individuals from multiple cameras in the operating room (OR). Therefore, this work considers several aspects of the human pose estimation problem, starting from single view scenarios and completing with human pose estimation multiple views in complex environments.

Zusammenfassung

Das Bestimmen der menschlichen Körperhaltung aus Fotos ist eine anspruchsvolle Aufgabe die von Guppen des Fachgebiets Computer Vision große Aufmerksamkeit bekommt. Das automatische Schätzen der Körperkonfiguration nützt vielen Anwendungen wie beispielsweise das Nachverfolgen von Menschen in Videos, Bewegungs-Erfassung (motion capture), Handlungserkennung, Überwachungsaufgaben und der Analyse chirurgischer Abläufe. Diese Dissertation befasst sich mit der Problemstellung des Bestimmens der menschlichen Körperhaltung aus einem oder mehreren Blickwinkeln. Zu Beginn wird die Aufgabe der Haltung einer einzelnen Person aus einem einzelnen Bild abzuschätzen betrachtet. Anschließend werden Mehrkamerasystemen benutzt um sowohl eine wie auch mehrere Körperkonfigurationen gleichzeitig abschätzen zu können. Hierfür stellt diese Arbeit neue diskriminative sowie generative Methoden vor.

Der Hauptbeitrag dieser Dissertation gliedert sich in drei Teile. Anfangs werden zwei diskriminative Methoden zur 2D Körperhaltungserkennung aus einem einzigen Bild vorgestellt. Der erste Beitrag verwendet Random Forests - ein Klassifikationsverfahren aus unkorrelierten Entscheidungsbäumen - wo hingegen der zweite Ansatz Deep Learning Verfahren benutzt. In beiden Fällen wird das Problem der Köperhaltungsbestimmung als Regressionsaufgabe definiert. Dazu wird die menschliche Pose über die Menge der Gelenkpositionen parametrisiert. Das Modell des Regressors der aus Bildern 2D Körperposen vorhersagt wird mittels eines Regression Forests (bzw. eines Convolutional Neural Networks - ConvNet) gelernt. Für das ConvNet schlagen wir ein Regressionmodell vor, das mittels Tukeys Biwight-Schätzer (ein robuster M-Schätzer) als Fehlermaß robuster gegen Ausreißer gemacht wird. Zusätzlich zum robusten Fehlermaß führen wir ein grob-zu-fein Modell ein, das, um die Genauigkeit der Regressionsergebnisse zu verbessern, Eingabebilder in zunehmender Auflösung verarbeitet. Der dritte Beitrag ist ein generativer Ansatz die Körperhaltung aus mehreren Perspektiven zu bestimmen. Insbesondere betrachten wir hier das Problem der Erkennung der 3D Körperpose mehrerer Menschen mit Hilfe mehrerer Blickwinkel. Dieses Problem zeichnet sich durch deutlich erhöhte Komplexität aus, da der Suchraum über Haltungskonfigurationen, verschiedenste - auch gegenseitige - Verdeckungen und blickwinkelübergreifende Mehrdeutigkeiten da die Identitäten der Personen in den einzelnen Perspektiven im Voraus nicht bekannt ist, deutlich größer ist. Zum Umgang mit diesen Problemen wird ein verringerter Suchraum konstruiert, für den korrespondierende Körperteile, bestimmt durch 2D Detektoren, in Blickwinkelpaaren trianguliert werden. Um Mehrdeutigkeiten Aufgrund falscher oder falsch zugeordneter Körperteile auflösen zu können führen wir ein 3D prictorial structures (3DPS) Modell ein. Dieses Modell verwendet unäre Potentiale aus den Blickwinkeln zusätzlich zu a-priori paarweisen sowie tertiären Potentialen. Damit ist das Modell sowohl generisch als auch anwendbar für einzelne sowie mehrere Personen im Bild. Abschließend wird das 3DPS Modell benutzt um die Körperhaltung mehrerer Personen aus mehreren Perspektiven im Operationssaal bestimmen zu können. So beinhaltet diese Arbeit verschiedenste Aspekte des Körperhaltungsbestimmungsproblems - von Szenarios mit einer

Person und einem Blickwinkel bis hin zu mehreren Personen in komplexen Umgebungen und vielen Blickwinkeln.

Acknowledgments

For the last three years, I had the pleasure to be part of the Computer Aided Medical Procedures (CAMP) group. The first person that I would like to cordially thank is Prof. Dr. Nassir Navab, the leader of the group and my Doktorvater. He has taught me how to change perspectives in order to find solutions to my problems. I am very grateful to him for his motivational words and our inspirational discussions.

I have interacted with many CAMPers which I would like to thank. Dr. Slobodan Ilic has guided me on the topic of human pose estimation and helped to start my research. I would like to thank him for his supervision and assistance. Of course, I would like to thank the vision and machine learning teams for the infinite hours of discussions, brainstorming and paper corrections. I am also very thankful to the newly created deep learning team for the close collaboration. There are many CAMP members with whom I had not a scientific collaboration, but I feel very happy to have met them. So, I would like to thank the famous Hilla family (including Ryu), Tobias Lasser, Maximilian Baust, Amin Katouzian, Ralf Stauder as well as all the other CAMP members with we usually meet in the kitchen.

I would like to thank my external partners for the great collaborations. I had a great time wokring with my colleagues from MPI in Saarbrücken. I would like to thank my partners from CVLAB in EPFL for working together in human pose estimation in the operating room. Of course, I am very thankful to Gustavo Carneiro for our great collaboration.

The people around me are not all related to my research environment. I would like to thank my friends for their patience all these years. Friends in Germany (Ünal, Rafael, Douglas, Debora, Jessy), friends in Greece (Antonis, Dimitris, Giannis, Kostas, Zoi, Nikos, Stefanos), friends abroad (Matthaiou) and many more. Moreover, I would like to thank Lena, Melanie, Spackolina and Melrow for being always by my side.

Last, and most important, I am extremely thankful to my parents Ilia and Alexandra, my brother Nikiforo and my whole family. They have supported all of my efforts, from the beginning of my studies until today.

-Vasilis

Φτὰσε ὀπου δεν μπορεὶς!
Νὶκος Καζαντζὰκης.

Reach what you cannot!
Nikos Kazantzakis.

Contents

List of Figures

List of Tables

1 Introduction

Computer vision contributes actively to the evolution of the modern societies by advancing the automation of processes in different areas. A well-studied subject within the field of computer vision are humans. Human detection, tracking and pose estimation from images are fundamental high-level problems that the community has addressed several times, during the past. In detection and tracking, humans are usually localized, by means of a bounding box, within a single view or multiple views. In pose estimation, the objective is to recover the position of the human body in the 2D or 3D space from image data. The human body position is commonly described by a skeleton and the image data comes from a single or multiple views. While reliable algorithms have been proposed for human detection and tracking in real-world environments, pose estimation still remains an open problem. In this thesis, we address the problem of human pose estimation in complex environments. We study the problem from a single view, as well as from multiple views. In multiple views, we investigate the problem both for single and multiple human pose estimation. Our ultimate goal is to perform multiple human pose estimation from multiple views. We select several applications to demonstrate our methodology, including the very challenging human pose estimation in the operating room (OR).

1.1 Motivation

The problem of human pose estimation has been addressed from different perspectives, based on the input modalities and number of camera views. Initially, the task had been defined within the domain of a single image. There is a vast amount of literature on human 2D pose estimation, where the goal is to parse the 2D body pose, in terms of skeleton, of a single human [8, 52, 60, 141, 149, 174]. In most cases, the application of single 2D human pose estimation has been oriented to sport activities [91, 186]. Later on, the computer vision community has considered the problem of multiple 2D human pose estimation from a single image [53]. In this paradigm, the 2D pose of multiple

individuals has been recovered from group activities, such as dancing. At the same time, the problem of 3D human pose estimation has attracted a lot of interest. Several methods have been proposed that recover the body pose in the 3D space, again in terms of skeleton, from a single image [2, 37, 68, 158]. However, it has been quickly understood that 3D human pose estimation requires more than a single camera view in order to robustly model the lack of the third dimension of image data. As a result, many methods have relied on multi-view camera systems for estimating the pose of a single individual. Due to the complexity of the task, most of the related work has been applied in studio environments, mainly focused on motion capture [5, 122, 161]. In addition to multi-view camera systems, the Kinect sensor, an RGB and depth camera, produces depth images which have been proven very useful for the task of 3D human pose estimation [157]. However, this technology is mainly applicable to indoor applications and the number of employed sensors is limited due to interference problems. More recently, estimating the 3D body pose in real-world scenarios has enjoyed substantial attention in the community. For example, estimating the 3D pose of players in sport events [40] or in public crowed places [81] is an active research topic. Important steps in this direction have been made by estimating the 3D pose of single human from multiple views using learning-based methods. Nevertheless, the problem of multiple human 3D pose estimation from multiple views remains open (Figure 1.1). While researchers have showed willingness to tackle the problem [121], there has been not much progress.

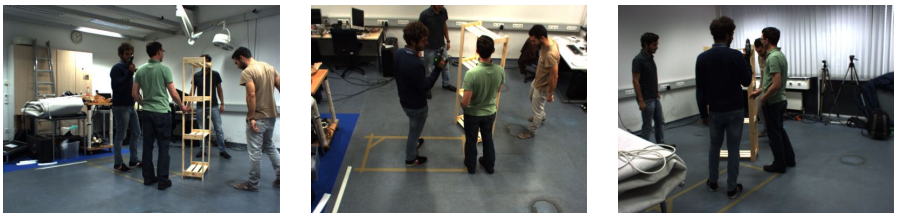


Figure 1.1: **Multi-View Human Pose Estimation**: Estimating the human 3D body pose from multiple views and for multiple individuals is an open problem that we address in this thesis.

1.2 Problem Definition and Challenges

In this thesis, we address the problem of multiple human pose estimation from multiple views. We regard this problem as an important missing piece of the task of human pose estimation. Our goal is to automatically estimate the pose of multiple individuals in 3D space, given a set of images from a calibrated multi-view camera system. However, the transition from 2D to 3D space and from single to multiple human pose estimation is a challenging task.

First of all, modelling the human body from image data is quite demanding due to its articulation and the big deformations that it can go through. Firstly, we propose 2D human models for addressing the problem of 2D human pose

estimation. Since the 2D models are built from images, we have to compensate for the missing dimension. Secondly, we introduce a 3D human model for multi-view human pose estimation. While a 3D model can capture a more complete body representation, modelling the body pose is not straight forward task due to the high dimensional body pose space.

Figure 1.2: **Different Human Body Poses**: Performing human pose estimation from images can be a difficult task due to the body pose and appearance high variation.

In a multi-view setup, the 3D space can be discretized into a volume in which the human body is defined as a meaningful configuration of body parts. Estimating the 3D body pose can be an expensive task due to the six degrees of freedom (6 DoF) of each body part and the level of discretization, as it has been analyzed by Burenius et al. [40]. In order to reduce the complexity of the 3D space, many approaches rely on background subtraction [161] or assume a simplified human model with fixed limb lengths and uniformly distributed rotations of body parts [40].

Another common problem, which has been particularly addressed in single human approaches (i.e. [5, 40]), is the separation between left-right and front-back of the body anatomy because of the different camera views. This problem becomes more complicated in multiple human 3D pose estimation when the identity of individuals is unknown. Thus, an association between the individuals across all views is required to avoid mixing the body parts of different individuals. For example, a left hand of one person in one view will have multiple left hand candidates in other camera views coming not only from the same person, but also from other individuals and potential false positive detections. In practice, this will create incorrect body part hypotheses that can lead to fake body poses in the 3D space. Similar to other computer vision tasks, we have to deal with common problems such as appearance variation, self-occlusion, occlusion between humans and different motion types (e.g. running or walking), which should be described by a single model (Figure 1.2). When considering also dynamic environments, where background subtraction is not feasible, the problem becomes more complicated in comparison to human pose estimation in a studio setup [121, 161].

To overcome these problems, we need robust models that can cope with the aforementioned constraints and sustain their robustness. For that purpose, we introduce a number of models for 2D and 3D human pose estimation which are applied in different single view and multi-view scenarios.

1.2.1 General Applications

A framework for multiple human pose estimation from multiple views can be applied in motion capture, surveillance and sport capturing systems (Figure 1.3). We provide some examples in which our framework could be applied. Motion capture systems haven been beneficial in film industry, especially for animating cartoon characters. The current technology is based on marker-based solutions which work only in a studio environment. Our framework is marker-less and can be directly applied in unconstrained environments. Another entertainment activity, where human pose estimation is very useful, are sport games. For example, estimating the pose of football or volleyball players, captured from different views, supports the analysis of a game. Furthermore, body pose estimation in sport activities helps for studying the tactics of the team and its opponents. A further application of our framework would be on surveillance. Public or crowed places are usually monitored by multiple view camera systems. Automatic human pose estimation could facilitate the recognition of unusual human actions and activities. Our framework could be combined with activity recognition solutions for automatizing the surveillance of public areas. Therefore, we see a number of different possible applications for a system that estimates the human body pose. In the following chapters, we adapt some of the above scenarios for evaluating our models and also demonstrate the applicability of our algorithms.

Figure 1.3: **Different applications of body pose estimation**: A framework for automatic human pose estimation can be applied in motion capture (top row), sport activities (middle row) and surveillance (bottom row).

1.2.2 Application in the Operating Room (OR)

A particular environment for human pose estimation is the operating room (OR). We choose this application among others to evaluate our methods in order to show the robustness of our algorithms in such a complex scenario. Our aim is to estimate the body pose of the surgeons and staff in OR. However, the plausible question of why we need to perform human pose estimation in OR can arise. Behind this interesting application, there is another motivation which is related to the surgical workflow modelling.

The task of surgical workflow refers to the phase recovery and analysis of a medical operation [135]. To that end, a number of available signals inside the OR are employed. The signals come from different instruments, monitoring and medical devices. Within this environment, the role of pose estimation from a multi-view camera system is to serve as an additional input modality to the surgical workflow analysis and modelling. For instance, the 3D body poses can be distinctive features for identifying human activities and thus can contribute to the phase recognition of the medical operation (Figure 1.4). In the thesis, we apply our models on this particular application and show that 3D human pose estimation can be used as an additional modality for the surgical workflow modelling.

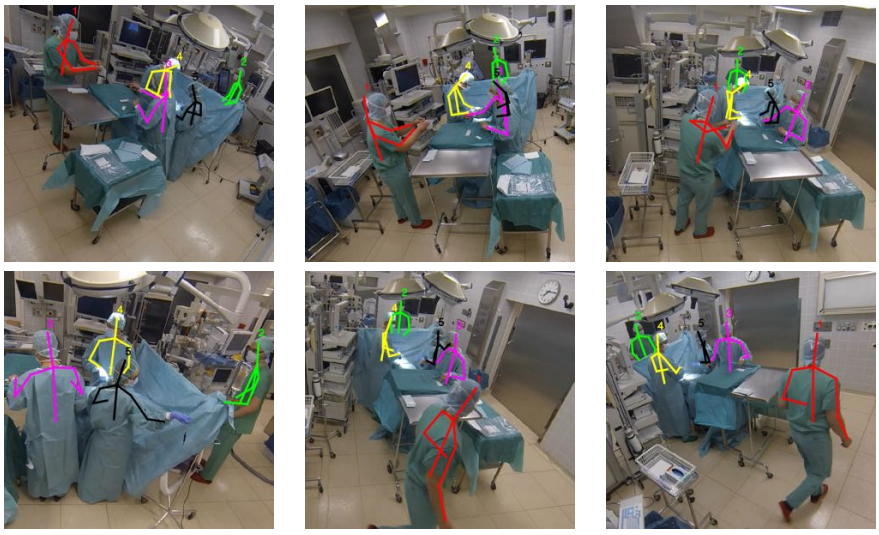


Figure 1.4: **Human Pose estimation in OR**: We address the problem of human pose estimation in the operating room (OR). We find this environment significantly complex and thus appropriate for evaluating our methods. The presented samples come from a unique dataset that we introduce for applying our human models.

1.3 Contributions

To achieve our objectives, we introduce a number of novel algorithms for human pose estimation in complex environments. The primary contributions

of the thesis are summarized as follows:

- We investigate the problem of human pose estimation by starting with the 2D space and single human. We propose two discriminative methods for estimating the 2D body pose of single human from an image. The first method is based on random forests, while the second builds on convolutional neural networks. In both methods, we regress the body joints of single human.
- We focus on the problem of pose estimation in the 3D space. In this case, we also consider the multiple human scenario and introduce an algorithm to perform multiple human pose estimation from multiple views. To that end, we build on 3D Pictorial Structures (3DPS) model, that relies on 2D body part observations across all camera views and a geometric 3D body prior.
- We introduce a number of datasets for multiple human pose estimation from multiple views. Among all datasets, we introduce the OR dataset for human pose estimation in OR. This is a unique dataset from a complex real-world environment.
- We show that object tracking is more effective without the bounding box target localisation. Instead, we propose a tracking by segmentation algorithm that applies on pixel level. We demonstrate that our tracker is generic, which means that it could be eventually applied to the of task human pose estimation.

The aforementioned contributions are presented of the following chapters. Next, we provide the outline of the thesis.

1.4 Thesis Outline

We provide an overview for each chapter of the thesis. Most of the methods and material of this thesis are published or are under submission for a major conference or journal. Therefore, we additionally provide the related work for each chapter.

Chapter 2. We present the theoretical background of the thesis. In particular, we go through the probabilistic graphical models and conditional random fields (CRFs) which form the base of our 3D human model.

Chapter 3. In this chapter, a discriminative model for 2D human pose estimation from a single view is introduced. We build our method based on a random forest that regresses the 2D body pose from an input image. Related work:

- Belagiannis, V., Amann, C., Navab, N., Ilic, S.: Holistic human pose estimation with regression forests. In: Articulated Motion and Deformable Objects, pp. 20–30. Springer (2014)

Chapter 4. We continue on single human 2D pose estimation from a single view and introduce another novel discriminative method. Our algorithm employs the convolutional neural networks (ConvNets) and a robust loss function for learning a map between the 2D body pose and input image. Related work:

- Belagiannis, V., Rupprecht, C., Carneiro, G., Navab, N.: Robust optimization for deep regression. In: Computer Vision (ICCV), 2015 IEEE International Conference on. IEEE (2015)

Chapter 5. In this chapter, we work on multiple human pose estimation from multiple views. We propose the 3D Pictorial Structures (3DPS) model as a generative model. Using the 3DPS model and 2D body part detectors, we parse the 3D body pose of multiple individuals. Related work:

- Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures for multiple human pose estimation. In: CVPR 2014-IEEE International Conference on Computer Vision and Pattern Recognition (2014)
- Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures revisited: Multiple human pose estimation. Pattern Analysis and Machine Intelligence, IEEE Transactions on (revised)

Chapter 6. We employ the 2D model from Chapter 4 and 3D model from Chapter 5 for human pose estimation in the operating room (OR). To that end, we introduce the OR dataset that simulates a medical operation in a real operating room (OR). Related work:

- Belagiannis, V., Wang, X., Beny Ben Shitrit, H., Hashimoto, K., Stauder, R., Aoki, Y., Kranzfelder, M., Schneider, A., Fua, P., Ilic, S., Feussner, H., Navab, N.: Parsing human skeletons in the operating room. Machine Vision and Applications (submitted)

Chapter 7. We step back from the problem of human pose estimation and we concentrate on generic object tracking. We argue that object localizations with a bounding box can be inaccurate and, for that reason, we propose a segmentation by tracking algorithm. Related work:

- Belagiannis, V., Schubert, F., Navab, N., Ilic, S.: Segmentation based particle filtering for real-time 2d object tracking. In: Computer Vision–ECCV 2012, pp. 842–855. Springer (2012)

Chapter 8. We conclude our work by presenting our findings, the limitations of the proposed methods and our directions for future work.

2 Background

The theoretical background of the thesis stems from Conditional Random Fields (CRFs), a type of probabilistic graphical model. For that reason, the principles of probabilistic graphical models and CRFs are presented in this chapter. To illustrate the theory behind a CRF, we rely on the paradigm of 2D human pose estimation from a single image. Moreover, we present the pictorial structures model, the most well-known graphical model for human pose estimation, which we later use in order to propose a graphical model for 3D human pose estimation.

2.1 Conditional Random Fields

Defining the body pose as a constellation of N parts, each part would correspond to a random variable y_i. The goal of pose estimation is to predict the vector $\mathbf{y} = (y_0, \ldots, y_N)$ of random output variables from the observation $\mathbf{x}$. The *observation* $\mathbf{x}$, an observed random variable, is a feature vector which can be computed from the input data (e.g. an image). The random variables, within the context of human pose estimation, can correspond to body parts and therefore there will have strong dependencies with each other, as well as with the observation $\mathbf{x}$ from the image data. *Graphical models* are a natural way to encode dependencies between random variables. In particular, a probabilistic graphical model (PGM) encodes the relation between the dependent and/or independent variables in one model. Moreover, the output of a PGM can be label predictions or marginals probability distributions. More specific to our problem, the task of the PGM is to infer a particular body pose $\mathbf{y}$, represented by a set of image coordinates, from the observation $\mathbf{x}$.

In general, a graphical model represents a family of distributions over the random variables, based on the type of graph. In this thesis, we focus on discriminative undirected graphical models. This type of graphical model is a Conditional Random Field (CRF). A CRF models the conditional distribution $p(\mathbf{y} \mid \mathbf{x})$ of the random variables $\mathbf{y}$, where the observation $\mathbf{x}$ is available. There are different types of graphical models [98], but we rely on a CRF for the

following reasons: First of all, the dependencies between the random variables and observation can be defined based on the problem. This means that, unlike generative models, the joint distribution $p(\mathbf{y}, \mathbf{x})$, which can be complex, does not have to be estimated. Consequently, the prior $p(\mathbf{x})$ is not part of the model as well. Modelling the prior $p(\mathbf{x})$ can be a difficult task due to the highly dependent features or large dimensions of the observation. As a result, the main advantage of a CRF among the graphical models is the computational efficient and simple structure. Designing and using CRFs includes three main steps: modelling, inference and parameter estimation of the CRF. Next, we describe these steps and connect them to the human pose estimation task. Note that a undirected graphical model has been often referred as Markov Random Field (MRF) in the literature. We also adapt to this terminology.

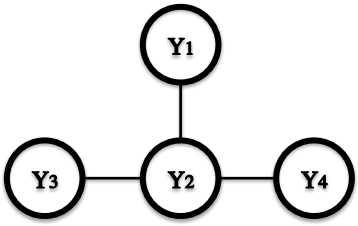


Figure 2.1: **Graphical model**: An undirected graphical model that would correspond to the head and shoulders of the human body.

2.1.1 Graphical Modelling

A graphical model is a framework to represent multivariate joint or conditional probability distributions and thus it is also called probabilistic graphical model (PGM). It can model the dependencies between different random variables and the relation between the random variables and observation. There are different types of graphical models such as Bayesian networks (directed graphical model) or Markov Random Fields (undirected graphical model). Moreover, the relation between the observation and random variables defines the type of the distribution that is joint or conditional. In our problem, we will consider an undirected graphical model for representing the conditional probability distribution over a set of random variables. Based on the type of the graphical model, a family of distributions is defined my means of a graph $G = (\mathbf{V}, \mathcal{E})$. In the graph, the nodes $\mathbf{V}$ correspond to random variables and the edges $\mathcal{E}$ denote dependencies between the variables, as depicted by Figure 2.1.

We follow the same notation as [170] and define the probability distribution p over sets of random variables $\mathbf{V} = \mathbf{X} \cup \mathbf{Y}$. The input variables or observation is $\mathbf{X}$, while the output variables is $\mathbf{Y}$. Moreover, we assume an output domain $\mathcal{V}$ of discrete states from which each variable can take its values. For example, the output domain of the human pose estimation problem would be pixel coordinates within the image plane. We consider discrete output, although continuous would also be possible. An assignment to the observation $\mathbf{X}$ can be defined by the vector $\mathbf{x}$ and x_s denotes the assigned value to the s variable,

where $s \in \mathbf{X}$. Moreover an assignment to a subset of the observation $a \subset \mathbf{X}$ can be denoted as x_a. A probability distribution p can be represented by a product of factors, where each *factor* is represented by $\Psi_a(\mathbf{x}_a, \mathbf{y}_a)$ and has scope $a \subseteq V$ (i.e. subset of variables). In general, a factor defines an interaction between a single or multiple random variables.

An *undirected graphical model* or MRF defines a family of probability distributions and factorizes according to a set of scopes $\mathcal{F} = a \subset \mathbf{V}$. The factorization can be written as:

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z(\mathbf{x})} \prod_{a \in \mathcal{F}} \Psi_a(\mathbf{x}_a, \mathbf{y}_a), \tag{2.1}$$

where for every factor (also called potential or local function) $F = \{\Psi_a\} : \mathcal{V}^{|a|} \rightarrow \mathbf{R}^+$, where the output is non-negative. The normalization constant $Z(\mathbf{x})$ transforms into probabilities the values of the factors by summing up all possible scopes. It is defined by:

$$Z(\mathbf{x}) = \sum_{\mathbf{x},\mathbf{y}} \prod_{a \in F} \Psi_a(\mathbf{x}_a, \mathbf{y}_a). \tag{2.2}$$

The normalization constant $Z(\mathbf{x})$ is also referred as *partition* function and its computation is in general intractable [98]. There has been proposed many algorithms for approximating it. The factor (or potential function) $\Psi_a(\mathbf{x}_a, \mathbf{y}_a)$ is assumed to have the form:

$$\Psi_a(x_a, y_a) = exp\left\{\sum_k w_{a,k} f_{a,k}(x_a, y_a)\right\}, \tag{2.3}$$

where $w_{a,k}$ are real valued parameters and $f_{a,k}(x_a, y_a)$ is a set of feature functions (also called sufficient statistics). The parameter $w_{a,k}$ defines a specific distribution over V from an exponential family. The feature function $f_{a,k}(x_a, y_a)$ models the interaction between variables and/or observation and it can be computed using an indicator function for discrete variables. A specific distribution out of the defined family is called *random field* . Finally, the factorization of the probability distribution in (2.1) can be represented by means of a factor graph.

A *factor graph* is a bipartite graph of the tuple $(\mathbf{V}, \mathbf{F}, \mathcal{E})$ that represents the factorization of the probability distribution p. Each node u_s in the graph corresponds to a random variable (Figure 2.2). The variables that define a factor Ψ_a are connected in the factor graph. In a factor graph, the variables nodes are denoted by circles and the boxes are factor nodes. Note that a factor/potential function is not bounded only to pairs of observed and unobserved variables. It can also model triplets or higher order potentials with multiple variables. In the next chapters, we make use of three type of potentials: unary, pairwise and ternary.

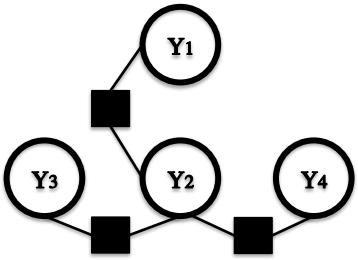


Figure 2.2: **Factor graph**: The factor graph of the undirected graphical model from Figure 2.1. The circles are random variables and the shaded boxes represent factors (also called potential functions). This factor graph encodes dependencies only between the output variables.

Conditional Random Field (CRF) For a factor graph G over the random variables $\mathbf{Y}$ and observation $\mathbf{X}$, the conditional distribution $p(\mathbf{y} \mid \mathbf{x})$ is a conditional random field iff for any assignment $\mathbf{x}$, the distribution $p(\mathbf{y} \mid \mathbf{x})$ factorizes according to G [104].

The conditional distribution is defined by a set of factors $F = \{\Psi_a\}$ that belong to G. Based on the factor exponential from Eq. (2.3) and the parameters $w_{a,k} \in \mathbb{R}^{K(A)}$ of each factor, the conditional distribution can be written as:

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{\Psi_A \in G} exp \left\{ \sum_{k=1}^{K(A)} w_{a,k} f_{a,k}(\mathbf{x}_a, \mathbf{y}_a) \right\} \tag{2.4}$$

where $K(A)$ is the number of feature functions for each factor Ψ_a. To define the parameters of the factors, we partition them to cliques. A *clique* in the graph G is a complete subgraph. That is a set of nodes which is connected with each other. The maximal clique of a factor graph includes largest possible node number [33]. We use the maximal clique rule to partition the factor graph G and consequently the factor into $C = \{C_1, \ldots, C_T\}$ cliques where each clique has fixed parameters. Theses parameters have to be estimated in the stage of parameter learning. At the end, each clique C_t corresponds to set of factors with parameters $w_{t,k} \in \mathbb{R}^{K(t)}$ and feature functions $\{f_{t,k}(\mathbf{x}_t, \mathbf{y}_t)\}$. Note that we use the index t instead of a to indicate the cliques. We can write the conditional random field (CRF) with the clique factorization as:

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{C_t \in C} \prod_{\Psi_c \in C_t} \Psi_c(\mathbf{x}_c, \mathbf{y}_c; w_t) \tag{2.5}$$

Moreover, each factor Ψ_c can be written as

$$\Psi_c(\mathbf{x}_c, \mathbf{y}_c; w_t) = exp \left\{ \sum_{k=1}^{K(t)} w_{t,k} f_{t,k}(\mathbf{x}_c, \mathbf{y}_c) \right\}. \tag{2.6}$$

Finally, the normalization (partition function) becomes:

$$Z(\mathbf{x}) = \sum_{y} \prod_{C_t \in C} \prod_{\Psi_c \in C_t} \Psi_c(\mathbf{x}_c, \mathbf{y}_c; w_t). \tag{2.7}$$

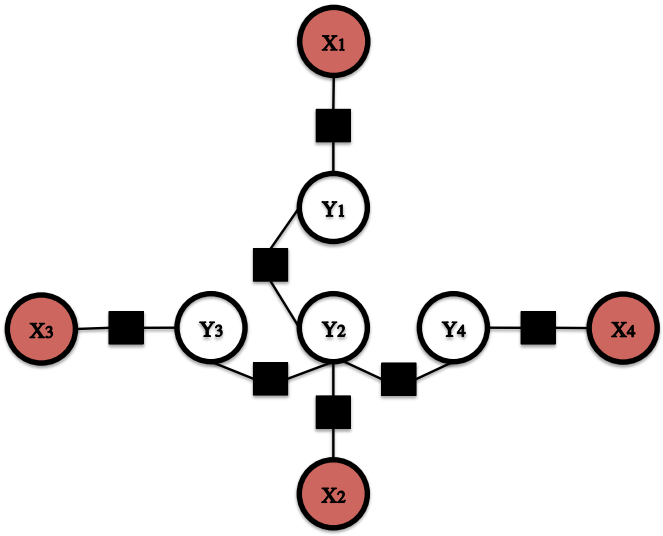


Figure 2.3: **Factor graph for CRF**: The factor graph specifies the conditional distribution.

The conditional random field (CRF) of Eq. (2.5) is defined by the potential functions as well as its parameters $\mathbf{w} = \{w_{t,k}\}$, where $\mathbf{w} \in \mathbb{R}^D$. In our problem on human pose estimation, we assume a single feature function function for each factor $K(t) = 1$ and thus D also corresponds to the number of factors. In this phase, we have specified the structure of the CRF by means of a factor graph and assumed that its parameters $\mathbf{w}$ are already estimated. The next important step is to select an inference algorithm for making predictions. These predictions can correspond to labels or the marginal distributions of the output variables. Moreover, inference can be also required for learning the parameters of the CRF. Finally, it is worth mentioning that CRFs are related to energy minimization. In particular, solving for the labels of a CRF with maximum probability is equivalent to energy minimization [130].

2.1.2 Potential Functions

The potential functions (or factors in term of factor graph) are designed for modelling the dependencies between the random variables and are usually defined based on the problem. Within the context of human pose estimation, we could propose a model with unary, pairwise and ternary potential functions. The *unary* potential functions would represent the relation between input (i.e. observation) $\mathbf{x}$ and output $\mathbf{y}$ variables, while the *pairwise* and *ternary* potential functions the relation between output variables. Given n output variables $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, a unary potential is denoted by $\phi_i(y_i, \mathbf{x})$, pairwise $\psi_{i,j}(y_i, y_j)$ and ternary $\psi_{i,j,k}(y_i, y_j, y_k)$ for the rest of the thesis. An example of a unary potential would a body part detector, while pairwise and ternary potential functions would encode a human body prior by modelling the relations between body parts. A variant of this model are pictorial structures [63], which comprises the

base of our model for 3D human pose estimation, introduced in the following chapters. Next, we present the tasks of inference and parameter estimation for a CRF and then present the pictorial structures model.

2.1.3 Inference

The aim of building a probabilistic graphical model is to make predictions based on the observation $\mathbf{x}$. The means to make the predictions is *inference*. Inference can be performed on the factor graph in two different ways, depending on the type of predictions. In the first case, the goal is to predict the labels of the random variables $\mathbf{y}$, given the observation $\mathbf{x}$ and a model with learnt parameters $\mathbf{w}$. This is *maximum aposteriori* (MAP) inference, where we look for the labels that maximize the posterior probability of Eq. (2.5), given by:

$$\arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}, \mathbf{w}), \tag{2.8}$$

where $\mathbf{y} \in \mathcal{Y}$ and $\mathbf{x} \in \mathcal{X}$. Instead of maximization, one could also turn the problem of inference to marginalization. This is the *probabilistic inference*, where the goal is to estimate the partition function $Z(\mathbf{x})$ and also the marginal distributions of the factors. The marginalization can be obtained by:

$$\sum_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}, \mathbf{w}). \tag{2.9}$$

In human pose estimation, the MAP inference would look for the most probable configuration of body pose. Furthermore, the probabilistic inference would estimate all the marginals of the human body in terms of body parts.

To accomplish both types of inference, there have been proposed several algorithms, but in general the problem is known to be NP-hard [156] for general factor graphs. However, the problem becomes tractable when imposing constraints on the graph structure. Moreover, exact inference is possible for tree-structured graphs (without loops between the nodes). In the *exact inference* problem, the forward-backward algorithm [144] is used for computing the partition function $Z(\mathbf{x})$ and marginals, while the Viterbi algorithm [64] is applied for estimating the most likely labels. In general, the inference algorithms that are applied in graphical models can be applied to a CRF as well [98]. Nevertheless, the complexity of the graph can make the task of inference computational inefficient. For instance, learning the parameters of the CRF by maximum likelihood usually requires multiple iterations of inference. As a result, a computationally efficient inference algorithm is important not only for the prediction task but also for learning the parameters of the CRF. Nevertheless, learning the parameters of the CRF or making predictions using structured prediction facilitates the task of inference [57, 176, 130].

In general graphs and graphs with loops in particular, exact inference is possible, but it can be inefficient in practice. A graph with loops can be transformed into a tree graph using the junction tree algorithm [106] and then exact inference is feasible. However, the transformations in the junction tree algorithm rely on clustering the variables and can always result in very large

clusters, where inference requires exponential time. An alternative solution to exact inference for general graphs is *approximate inference*. In the past, there have been proposed many algorithm for approximate inference [98], but two well known categories of algorithms are *Monte Carlo* and *variational* algorithms. Monte Carlo algorithms are based on sampling in order to have an approximation of the distribution of interest. They guarantee a solution from the distribution of interest, but they require a very respectful amount of computation time. A well-known class of Monte Carlo algorithms are the Markov Chain Monte Carlo (MCMC) methods [98]. In variational methods, inference is formulated as an optimization problem and the goal is to minimize an energy function. The outcome of the optimization is an approximation of the distribution of interest. Variational methods are faster in comparison to Monte Carlo algorithms and thus well-suited for CRFs. A well-established variational algorithm for approximate inference is belief propagation , which operates on tree-structured or loopy graphs [137]. Moreover, the forward-backward and Viterbi algorithms are particular cases of belief propagation and thus belief propagation can be applied for exact inference as well. The basic idea of belief propagation is to transfer messages between the factors as well as the variables in order to compute the MAP or the marginal distributions. The message passing is done by eliminating gradually factors or variables, from the bottom to the top (in a tree structure). Belief propagation estimates the exact solution in tree-structured graphs using the max-product algorithm [98], but it has demonstrated good convergence for loopy graphs as well, by using the sum-product algorithm [98]. We choose belief propagation for inference for the problem of human pose estimation.

The task of inference is performed after specifying the graph structure and learning the parameters $\mathbf{w}$. Next, we present how the learning of the parameters is done and also its connection to inference.

2.1.4 Parameter Estimation

Until this point, it has been assumed that the parameters $\mathbf{w}$ of the CRF are given. In this part, we discuss the *parameter estimation* of the CRF, which is also referred as parameter learning. The parameters of the CRF should be selected carefully so that the conditional distribution of Eq. (2.5) will be as close as possible to the true distribution, measured by the Kullback−Leibler (KL) divergence [100]. The true distribution is unknown, but we usually have a set of training data samples $\mathcal{D} = \{(\mathbf{x}^s, \mathbf{y}^s)\}_{s=1,\dots,S}$ of the unknown distribution for learning the parameters. Moreover, it is assumed that the training samples are independent and identically distributed (i.i.d).

Probabilistic learning Learning the parameters of the CRF from training data has been usually accomplished in a probabilistic manner using *maximum likelihood*, where this type of training is also called CRF training [130]. The main idea is to choose the parameters $\mathbf{w}$ that maximize the probability of the training data under a model. Since the modelled distribution is conditional, the likelihood can be also called *maximum conditioned likelihood*. Furthermore,

we can replace the likelihood with the log-likelihood because the logarithm is a monotonically increasing function and the maximum is at the same point. This action facilitates the gradient computation, which is part of the likelihood maximization. The conditional log-likelihood is given by:

$$\mathcal{L}(w) = \sum_{s=1}^{S} \sum_{C_t \in C} \sum_{\Psi_c \in C_t} \sum_{k=1}^{K(t)} w_{t,k} f_{t,k}(\mathbf{x}_c^s, \mathbf{y}_c^s) - \sum_{s=1}^{S} \log Z(\mathbf{x}^s), \tag{2.10}$$

where we assume that the factor graph G has all components connected (e.g. a tree-structured model). For the case where the training data can be represented by disconnected components in the factor graph G, the summation over S is not required. Due to the large number of parameters $\mathbf{w}$ in Eq. (2.10), a regularization term is necessary. The role of the regularization is to penalize weights with big norm and avoid overfitting. A usual choice is $L2$, but $L1$ can be also used to regularize the parameters of the CRF during training. Moreover, a prior on the parameters can be also introduced for determining how much the big weights should be penalised. A Gaussian prior is often chosen for that purpose. By integrating an $L2$ regularizer and λ prior in Eq. (2.10), we obtain:

$$\mathcal{L}(w) = \sum_{s=1}^{S} \sum_{C_t \in C} \sum_{\Psi_c \in C_t} \sum_{k=1}^{K(t)} w_{t,k} f_{t,k}(\mathbf{x}_c^s, \mathbf{y}_c^s) - \sum_{s=1}^{S} \log Z(\mathbf{x}^s) - \lambda \|\mathbf{x}\|, \tag{2.11}$$

where $\lambda = \frac{1}{2\sigma^2}$ and σ depends on the size of the training data. Obtaining $Z(\mathbf{x}^s)$ is complex and thus Eq. (2.11) does not have a closed form solution. For that reason we rely on numerical optimization for finding a minimal solution. Very importantly, the maximum likelihood is a convex function and local minima is global minima at the same time. The partial derivatives of Eq. (2.11) are given by:

$$\frac{\partial \mathcal{L}}{\partial w_{t,k}} = \sum_{s=1}^{S} \sum_{\Psi_c \in C_t} f_{t,k}(\mathbf{x}_c^s, \mathbf{y}_c^s) - \sum_{s=1}^{S} \sum_{\Psi_c \in C_t} \sum_{\mathbf{y}'_c} f_{t,k}(\mathbf{x}_c^s, \mathbf{y}_c^s) p(\mathbf{y}'_c \mid \mathbf{x}) - \frac{w_{t,k}}{\sigma^2}. \tag{2.12}$$

In order to compute $Z(\mathbf{x}^s)$ as well as the marginal distribution $p(\mathbf{y}'_c \mid \mathbf{x})$, inference is required for each training instance. This an expensive step which can be relaxed with an efficient inference algorithm such as belief propagation.

The optimization of $\mathcal{L}(w)$ can be performed using any gradient descent method because the function is differentiable. The simplest way is with steepest descent optimization. However, the number of parameters of the CRF are usually large and are required many iterations until convergence. An alternative way to optimize $\mathcal{L}(w)$ is with second-order gradient descent. A well-known second-order gradient descent method is Newton's method [27]. It requires fewer iterations but it comes with a higher computational cost, due to the computation and inversion of the Hessian matrix at every iteration. In current techniques of optimization, there are have been proposed quasi-Newton methods, which are approximations of second-order gradient descent, for more efficient optimization. For instance, BFGS [27] and Limited-memory BFGS (L-BFGS) [114] compute an approximation of the Hessian matrix for relaxing

the computations. Furthermore, conjugate gradients is another method for gradient descent approximation [27]. Finally, stochastic gradient descent [35] is probably the most effective way to perform the optimization because it relies on batches for estimating the gradient. A batch is composed of a small number of training samples that jointly contribute to the gradient computation, instead of considering the gradient of each sample independently.

Approximation training Learning the parameters $\mathbf{w}$ of the CRF by maximizing the conditional likelihood requires the graph G to be tractable. In cases where this assumption does not hold (e.g. complex graphs with higher order dependencies and loopy graphs), working directly with the likelihood can be computationally expensive or even not possible. For that reasons, there have been proposed other objectives for approximate training of the CRF. One popular way of approximate training is to substitute the likelihood with an approximation of it, a *surrogate likelihood* [170] (also referred as pseudo-likelihood [130]). The surrogate likelihood is easier and faster to compute. Moreover, it is differentiable and consequently can be optimized using a gradient-based method. A second alternative to maximum likelihood is a direct approximation of the marginals distributions. The main idea is to use a generic inference algorithm that approximates the marginals $p(\mathbf{y}'_c \mid \mathbf{x})$ of the gradient in Eq. (2.12). The approximated marginals can be used for performing gradient descent. Both ways of approximations training can be realized using belief propagation or Markov Chain Monte Carlo (MCMC) sampling. Recently, margin-based parameter learning has gained a lot of attention within the framework of structured prediction [130]. Below, we discuss this methodology of parameter learning which we also adopt in our model.

Margin-based parameter learning The problem is defined again similar to probabilistic parameter learning. We are given a set of training data $\mathcal{D} = \{(\mathbf{x}^s, \mathbf{y}^s)\}_{s=1,\dots,S}$ of the unknown probability distribution that we will use for learning the parameters $\mathbf{w}$ of the CRF. In addition, we assume a loss function $\Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ which measures the difference between the ground-truth label $\mathbf{y}$ and prediction $\mathbf{y}'$ for a training sample. Learning the ideal parameters $\mathbf{w}$ would result in minimizing the loss function $\Delta(\mathbf{y}, \mathbf{y}')$. The loss Δ can be defined as Bayes risk and then minimized using structural risk minimization [178]. In particular, we seek for a prediction function $\mathfrak{f}$ that minimizes the regularized empirical risk, given by:

$$R(\mathfrak{f}) + \frac{C}{S} \sum_{s=1}^{S} \Delta((y)^n, \mathfrak{f}((x)^n)) \tag{2.13}$$

where the first term corresponds to the regularization and the second to the empirical estimation of the expected risk. The regularization helps for preventing overfitting and it is usually an $L2$ norm, while the prediction function has the form $\mathfrak{f} = \arg\max_{\mathbf{y}} g(\mathbf{x}, \mathbf{y}, \mathbf{w})$. The minimization of Eq. (2.13) is feasible using structured support vector machine (S-SVM) training [176].

We define the compatibility function $g(\mathbf{x}, \mathbf{y}, \mathbf{w}) = \langle \mathbf{w}, f(\mathbf{x}, \mathbf{y}) \rangle$, where $f(\mathbf{x}, \mathbf{y})$ is the feature function similar to Eq. (2.3). The compatibility function is linear as

well as equivalent to the product of all factors of Eq. (2.4), assuming again that each factor has one feature function. The compatibility function is parametrized by the parameters $\mathbf{w}$. Structured support vector machine training is performed by minimizing the parameters $\mathbf{w}$ and is expressed by:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{S}\sum_{s=1}^{S} \mathcal{L}(\mathbf{x}^n, \mathbf{y}^n, \mathbf{w}), \tag{2.14}$$

where

$$\mathcal{L}(\mathbf{x}^n, \mathbf{y}^n, \mathbf{w}) = \max_{\mathbf{y}} \Delta(\mathbf{y}^n, \mathbf{y}) - g(\mathbf{x}^n, \mathbf{y}^n, \mathbf{w}) + g(\mathbf{x}^n, \mathbf{y}, \mathbf{w}). \tag{2.15}$$

The regularization constant $C > 0$ is a hyper-parameter. The Eq. (2.15) also expresses the Hinge loss adapted for structured output [130]. Consequently, the minimization of Eq. (2.14) is similar to training a support vector machine [45] by maximizing the margin between different labels. Other than the Hinge loss, there are have been proposed different types of losses based on the problem. The most popular loss function is the zero-one (0-1) loss $\Delta(y, y') = 1_{y \neq y'}$, where $y, y' \in \mathcal{Y}$. This type of loss penalizes equally every incorrect prediction. Furthermore, the hierarchical multi-class loss penalizes fewer incorrect predictions that are close to the true label and Hamming loss is well-suited for segmentation.

The optimization of Eq. (2.14) using gradient-based method, as with CRF training, is not possible, because Eq. (2.14) is not differentiable. However, it is a convex function and thus can be minimized using convex optimization [28]. For instance, sub-gradient descent minimization [26] can be used for minimizing Eq. (2.14), but the convergence is generally slow. To overcome the limitation of non-differentiability, there has been proposed a formulation of the S-SVM with *slack variables*. To that end, we define a vector $\boldsymbol{\xi} = (\xi^1, \xi^2, \ldots, \xi^S) \in \mathbb{R}^S$ of S auxiliary variables which are called slack variables. Then the S-SVM training is accomplished by:

$$(\mathbf{w}^*, \boldsymbol{\xi}^*) = \arg\min_{\mathbf{w}, \boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{S}\sum_{s=1}^{S} \xi^s \tag{2.16}$$

subject to:

$$g(\mathbf{x}^s, \mathbf{y}^s, \mathbf{w}) - g(\mathbf{x}^s, \mathbf{y}, \mathbf{w}) \geq \Delta(\mathbf{y}^s, \mathbf{y}) - \xi^s, \tag{2.17}$$

for $s = 1, \ldots, S$. This optimization is possible with gradient-based methods, but it is very complex because of the introduced constraints which is also difficult to fit in the memory. The problem of the number of constraints was the principal motivation for the cutting plane algorithm [96, 176]. The algorithm searches for the optimal parameters $\mathbf{w}$ and number of constraints at the same time. During the optimization, more constraints are added progressively based on the most violated one. When the optimum solution is met, the algorithm terminates and no more constraints are included. The algorithm has a good convergence rate and we also use it for our problem. In some cases where the value of C is large, the convergence is weak. To solve this problem, there

has been the one slack formulation of S-SVM. Finally, an S-SVM can be also kernelized [130], but is not necessary in our problem.

We have discussed the three important steps for designing a CRF: specifying the structure by means of a factor graph, making predictions through inference and learning the parameters of the CRF. In Chapter 5, we make use of the CRF framework for performing 3D human pose estimation from multiple views. Next, we present pictorial structures, a very popular model for 2D human pose estimation that is based on graphical modelling. Starting from this model, we propose a 3D pictorial structures model.

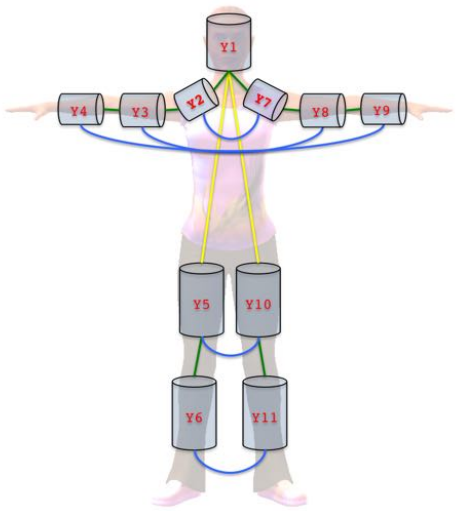


Figure 2.4: **Human body graphical model**: In the presented graph 11 variables are used to represent the body parts. The pairwise relations are expressed by edges of the graph.

2.2 Pictorial Structures

The model of *pictorial structures* has been introduced in the 70s by Fischler and Elschlager for modelling the human body [63], but it was popularized much later by Felzenszwalb and Huttenlocher [59]. More recently, Andriluka et al. [8] have revisited the model for giving it a learning-based perspective. Pictorial structures have been actively applied on 2D human pose estimation for many years and have formed the state-of-the-art in this problem [10, 52, 61, 91, 141, 195]. In all cases, the idea of the model is the same: the body is decomposed into a set of N body parts (Fig. 2.4) and the goal is to find the most plausible body configuration $\mathbf{y} = (y_0, \ldots, y_N)$ using the observation $\mathbf{x}$ and a body prior. The observation has been varied between different type of body features and body part classifiers. The most promising results have been achieved using classifiers as body part detectors. The body prior has been, for most of the cases, a 2D Gaussian distribution that models the relations between body parts.

Finally, pictorial structures have usually modelled these relations up to pairs of body parts.

According to the CRF formulation, we can build a factor graph for the human body by assuming that each body part is an output variable and the observation forms the input variables. Given the fact that the image space can be large, based on the 2D discretization, and the number of body parts is usually around 10 for the whole body, the task of inference is intractable. However, Felzenszwalb and Huttenlocher have made inference tractable using a distance transform and expressing all pairwise relations as Gaussians [58].

Figure 2.5: **2D human pose estimation**: Applying pictorial structures for single human 2D pose estimation.

The pictorial structures model has been widely applied to single human 2D body pose estimation (Fig. 2.5). In this thesis, we propose a 3D pictorial structures model for single and multiple human pose estimation, by deriving inspiration from the original 2D model.

2.3 Applications in Vision Problems

Besides pictorial structures, probabilistic graphical models and CRFs in particular have been applied into plethora of computer vision problems. For instance, a very well known problem that has been extensively addressed with CRFs is image segmentation [103, 129]. Other popular applications of CRFs are on face pose estimation [194], object localization [32], multi-class classification [70], stereo [150] and optical flow [167].

Modelling the aforementioned problems with probabilistic graphical models is usually subject to generative models. Below, we shortly discuss the properties of the generative models and compare them with the related category of the discriminative models.

2.4 Discriminative & Generative Models

In the literature for human pose estimation and computer vision in general, there have been proposed many algorithms that are based on *generative* or *discriminative* models [127]. Both type of models are used to make predictions from the posterior $p(\mathbf{y} \mid \mathbf{x})$. A generative method learns a model of the joint probability $p(\mathbf{x}, \mathbf{y})$ and then the posterior $p(\mathbf{y} \mid \mathbf{x})$ is estimated using the Bayes rule and a prior model. The prediction with a generative model corresponds to the most likely output (e.g. label) $\mathbf{y}$ after performing inference. On the other hand, a discriminative model learns a direct map between the input (i.e. observation) $\mathbf{x}$ and output $\mathbf{y}$ for modelling the posterior $p(\mathbf{y} \mid \mathbf{x})$. Both models are powerful and have demonstrated promising results. The discriminative models require considerable amount of training data from the true, but unknown distribution, while this is not the case for the generative models. On the other hand, the inference in the generative models can be complicated.

In this work, we make use of both models for the problem of human pose estimation. At first, we propose two discriminative models for performing 2D human pose estimation. During our evaluation, we have noticed that our results using discriminative models are very promising in comparison to related works that rely on generative models. Later, we propose a generative model for 3D human pose estimation that gives state-of-the-art results as well. As it is presented in the thesis, both models can result in very good performance. Combining both models has been proven to be the most efficient way [141, 184]. In Chapter 6, we also combine a discriminative 2D human model with a generative 3D human model for human pose estimation in the operating room (OR).

3

Single-view Human Pose Estimation

The first step towards multi-view human pose estimation is single-view human pose estimation. In this chapter, we address the problem of estimating the 2D human body pose from a single image. We aim to infer body poses, defined by body parts, in the 2D space across different views in order to afterwards build a 3D space of body poses for multi-view human pose estimation.

To tackle the problem of 2D human pose estimation, we propose a discriminative model (also called holistic model within the field of human pose estimation) that predicts the body pose using random forests. Random forests are an ensemble learning method that has been mainly used for classification and regression tasks [46]. It gradually builds and evolves a set of decision trees based on an objective function. In our problem, we propose a discriminative model that predicts body configurations, in terms of pixel coordinates, from an image with a localized person. For that reason, we rely on a regression forest which is learnt from a set of training data. In particular, our model learns the appearance of the human body from image patches. The patches are randomly chosen from a tank of candidate patches and then used for training the regression forest. During training, a mapping between image features and human poses is learnt, defined by joint offsets. During prediction, the joints offset are estimated with a mode-seeking algorithm. In the following sections of this chapter, we present in detail our algorithm and the principles of the regression forest.

At first, the literature is reviewed and the related work on 2D human pose estimation is presented. We discuss about holistic (i.e. discriminative) and part-based (i.e. generative) approaches. Then we present our algorithm which is robust to occlusion or noisy data. These properties are demonstrated during the evaluation part. Moreover, we compare our holistic model with related work on three publicly available datasets. In the last part the Chapter, we summarize the advantages and disadvantages of our model and discuss aspects that can be further addressed.

3.1 Introduction to 2D Human Pose Estimation

Estimating the body pose is a fundamental problem in computer vision community [123]. It has a wide range of potential applications such as surveillance, motion capture and behaviour analysis. Based on the application, there can be huge amount of human appearance variations. Furthermore, real-life environments usually include dynamic background or clutter. In order to address these challenges, most of the recent methods rely on modelling the human body from an ensemble of parts using generative models [8, 92, 141, 97, 186]. During the past, there have been proposed many discriminative methods as well, which had experienced generalization limitations. However, the discriminative methods have come to the fore again with the advances of deep learning [22].

In general, there are two main categories of methods for 2D human pose estimation: holistic (i.e. discriminative) and part-based (i.e. generative). In both categories, the human pose is defined by means of a body skeleton that is composed of a number of body joints or parts. On one hand, the part-based approaches synthesise the body skeleton using a set of parts, where the most popular model of this category is pictorial structures [8, 59, 63]. Most of the state-of-the-art approaches for human pose estimation have used to rely on pictorial structures [21, 92, 141, 186], but recently deep learning has defined new standards in the field of human pose estimation [22]. The part-based approaches have delivered promising results on standard evaluation datasets, but they build on complex appearance and body prior models. Training a part-based model can demand a lot of computational power and require long training time. Furthermore, the inference time can be also a problem for real-time applications.

On the other hand, the holistic approaches predict the body skeleton by learning a direct map between image features and body poses [2, 125, 146, 177]. These approaches usually face problems with occlusion or noise because they require a big amount of data from the target distribution. Moreover, they usually generalize up to the level at which unknown poses start to appear. Nevertheless, Random Forests [38] have been proven to generalize well to unobserved poses [72, 157]. For that reason, we also use them for our framework.

In this chapter, we address the problem of 2D human pose estimation in still images, by relying on a holistic model. We propose to learn an appearance model of the human body from image patches. The patches, which are randomly chosen from a bounding box around a localised individual, are used for extracting HOG features and training a regression forest [38]. At the training time, a direct mapping between image features and human poses is learnt, where a body pose is defined by joint offsets. During prediction, we recover the body pose under occlusion or from noisy data (Figure 3.1). Moreover, we propose an efficient algorithm for estimating the mode of the joint density function from the aggregated leaf samples during the prediction task.

In the experimental section, we demonstrate that a holistic approach is not limited to complete data for performing accurate human pose estimation. To

show this property of our holistic approach, we evaluate our method on two publicly available datasets which include self-occlusion, appearance and pose variations. In addition, we propose a challenging dataset which is different from the existing datasets because of its low resolution and noise in the data. We compare our method with the state-of-the-art approaches and achieved better or similar results.

Next, we continue with the presentation of the related work on 2D human pose estimation, followed by the proposed method and the experimental section. Finally, we draw useful concussions at the last part of this Chapter.

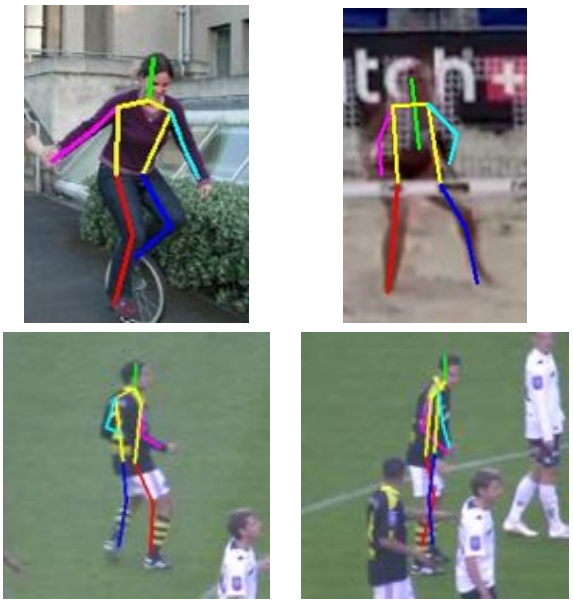


Figure 3.1: **Human Poses**: Qualitative results of our method on different data samples. We recover human poses with large appearance and motion variations. Furthermore, our algorithm handles (b)-(c) self-occlusion or (d) noisy input data.

3.2 Related Work

There is a tremendous amount of approaches that tackle the problem of human pose estimation from still images [123]. During the presentation of the related work, we follow the categorization of the methods into holistic and part-based. Finally, we present methods that rely on deep learning and fall in both categories.

Part-based approaches: Starting from the part-based methods, pictorial structures models have become the current state-of-the-art in human pose estimation in the last decade. They have been introduced in the 70s [63], but popularized [8, 59]. In the pictorial structures models, the human body is decomposed into a set of body parts, prior on the body pose. The goal is to infer the most

plausible body configuration given the image likelihoods and a prior. The problem is usually formulated using a conditional random field (CRF), where the unary potential functions include, for example, body part classifiers, and the pairwise potential functions are based on a body prior. There has been proposed several methods on improving the pictorial structures model. For example, one idea is by using better appearance models [10, 52, 149]. This has also been done by using random forest classifiers for body parts [95] or regression forests [48]. Shape-based body parts have generally achieved better performance [195]. Another direction for improvement is to introduce richer prior models using mixture of models [92, 186] or fully connected graphs [25]. Recently, the idea of modelling the body part templates jointly has been also introduced in [48, 168]. In [48], two layers of random forests capture the information between different body parts, while in [168] the body parts are sharing similar shape. Both directions of improving pictorial structures have resulted in strong local appearance and prior models. However, part-based models, such as pictorial structures, fail to capture the whole anatomy of the human body. They model parts of it and then try to synthesize the body pose. This means that fully occlued parts might not be able to derived. Further, the part-based models have evolved by building on computationally expensive and complex techniques.

Holistic approaches: Unlike part-based methods, the holistic approaches rely on learning and predicting the joint positions of the human skeleton at one step. They usually rely on learning a direct mapping between image features and human poses. For instance, mapping exemplars to human poses, became the standard way on holistic pose estimation [69, 125, 155]. The disadvantage of the exemplar-based approaches is the necessity for accurate matching of the whole body. To solve this problem method based on classification [2], regression [177] and segmentation [85] have been proposed. However, these methods can be sensitive to noisy input and cannot generalise to unknown poses. In order to cope with these problems, holistic approaches have relied on random forests [72, 146, 157] in combination with patch based features. In depth domain, random forests have been used for body classification [157] and regression [72]. In both cases, a holistic model has been proposed for classifying the body joints [157] or predicting their position [72] in the 3D space. In the image domain, random forests have been introduced for human body pose classification [146].

Deep learning approaches: Deep learning has become very popular in different problems of computer vision community. In human pose estimation, there have been proposed several method that rely on the holistic or part-based idea. In both cases, the common principle is to learn the image features that model the appearance of the human body jointly with a classifier or regressor. In part-based models, the body is decomposed again into a set of parts and the goal is to infer the correct body configuration from the observation using a CRF formulation [42, 134, 173]. However, instead of body part detectors that are uncoupled for the image features, now deep part detectors are trained. They

serve as unary potential functions and also as an image-based body prior for the computation of the pairwise potential functions. In the holistic approaches, deep learning is used for learning the complex mapping between an image and body pose. The recent advances in the automatic extraction of high level features using deep learning [111, 140, 174] has boosted the holistic methods to deliver state-of-the-art results. More specifically, Toshev et al. [174] have proposed a cascade of ConvNets for 2D human pose estimation in still images. Furthermore, temporal information has been included to the ConvNet training for more accurate 2D body pose estimation [140] and the use of ConvNets for 3D body pose estimation from a single image has also been demonstrated in [111].

Finally, the combination of holistic and part-based methods has been explored by introducing the concept of Poselets [36] in the pictorial structures framework [141, 184]. These approaches have proposed an intermediate representation but they still do not capture the whole anatomy of the human body. Furthermore, deep learning has not yet used for actively combining part-based with holistic methods.

In our work, we adapt the idea of regression forests to the image domain and learn to map image features to 2D human poses. To the best of our knowledge, we are the first ones who apply a regression forest to image data for estimating the body joints at once. The big advantage of our method in comparison to other holistic approaches is our ability to cope with incomplete data. For the rest of this Chapter, we present our regressor that relies on random forests, but in the next chapter we formulate the same problem using deep learning.

3.3 Random Forest

Random forest has become very popular for human pose estimation from depth data [46, 72, 157]. In this work, we build on a regression forest for extracting the human pose from image data. Below, we explain the basic principles of a regression forest and the way we apply it to our problem.

3.3.1 Regression Forest

A regression forest is an ensemble of regression trees T that estimates continuous output. The goal of training a regression forest is to learn a mapping between image patches and the parameter space. In our paradigm, the parameter space $\mathbb{R}^{2\times N}$ consists of a set of N joints in the 2D space. The body skeleton is defined by the joints and the image patches are estimated using HOG features [47]. We choose the HOG features as descriptors because of their robustness in different task such as object detection [57], tracking [12] and classification [34].

During training, a pool of randomly extracted image patches P with associated skeleton joint offsets serves as input to each tree. The patches are extracted from random positions within a bounding box that localises the human. The body pose is also expressed in the coordinate system of the bounding box.

Then, a tree is built from a set of nodes which include binary split functions. Each node encloses a split function θ which is defined on the values of the HOG features of the patch. The HOG feature vector of the image patch is extracted as in [57]. The binary split function determines if a p sample image patch will go to the left P_l or right P_r subset of samples. In particular, the split function is a threshold on one dimension of the HOG feature vector. Among the dimensions of the HOG feature vector, the threshold that gained the best split defines the split function:

$$\theta^* = \arg\max_{\theta} g(\theta), \tag{3.1}$$

where $g(\theta)$ corresponds to the information gain. The information gain measures how well the split function divides the training data into two subsets P_l and P_r. Consequently, the criterion for choosing the split function is to maximize the information gain $g(\theta)$ by optimally splitting the input image patches of the current node. The information gain is formulated as:

$$g(\theta) = H(P) - \sum_{i \in \{l,r\}} \frac{|P_i(\theta)|}{|P|} H(P_i(\theta)), \tag{3.2}$$

where $H(P)$ is the entropy. The entropy is estimated by the sum-of-squares-differences:

$$H(P) = \sum_{p \in P} \sum_{j} \left\| \mathbf{v}_{p,j} - \boldsymbol{\mu}_j \right\|_2^2, \tag{3.3}$$

where the vector $\mathbf{v}_{p,j}$ includes the offsets for each joint j from the image patch centre and $\boldsymbol{\mu}_j$ denotes the mean for each joint offset. In order to estimate the mean $\boldsymbol{\mu}_j$, we set a threshold ρ that considers only joints close to the sampled patch, as in [72]. Finally, the tree grows until it reaches the maximum depth, the minimum number of samples per leaf or the information gain for the node drops below a threshold. The same process is a repeated for all the trees of the forest. Finally, we store the offsets of all body joints in the leaves.

3.3.2 Method Parameters

In order to efficiently train the regression forest, there is a number of parameters that has to be determined during training. We have observed that the performance is highly depended on these parameters. Below, we discuss these parameters.

Image Patches: The size of all image patches is predefined during training and prediction. Thus, all the HOG feature vectors have the same size. We discretize the image gradients into 9 bins and follow the implementation from [57].

Scale Invariance: The training persons in different training images are apparently of different sizes, but they are all localized by a bounding box. In our

experiments, we have noticed that training a scale invariant regression forest is not a straightforward task. Thus, we scale all the data with respect to the height of the bounding box which usually corresponds to the height of the person. This allows us to capture pose variations of different humans using a common scale. Since we assume a localized person, we scale at the prediction phase as well.

Threshold ρ: We argue that a split function has a more local than a global role. For that reason, samples having large offsets are penalized by a threshold. We set it experimentally to 0.8 of the human bounding box height and exclude the joints that are outside this radius.

3.3.3 Prediction

In the prediction phase, the individual is localised and rescaled based on a bounding box. Similar to training, random pixel location are generated as input to the regression forest. An image patch is extracted for each random location and the HOG feature vector is then computed. In each tree, the split functions direct, left or right, the input image patch until it reaches the leaf in which we have stored the vectors that predict the body joint offsets. After performing this step for multiple random patches, the next step is to aggregate the votes of the leaves of the different trees.

For a certain joint, finding the most probable joint offset out of all candidates corresponds to estimating the mode of the density function, defined by the aggregated joint offsets. The most common algorithm for estimating the mode is Mean Shift [44]. However, Mean Shift is a computationally expensive algorithm and can require significant amount of time for convergence, given a plethora of samples at the leaves. To overcome this limitation, we propose the *dense-window* algorithm which is a greedy approach for estimating the mode of a density function from the samples. The *dense-window* algorithm relies on a sliding window search in which convergence is deterministic. It only depends on the step of the sliding window and scales linearly with the number of the samples.

To enable fast estimation, the *dense-window* algorithm discretizes all the 2D predictions for every joint on a grid such that every grid cell stores the number of predictions that lie within this cell. The runtime is linear to the number of joint predictions s. Then, an integral matrix is generated for each cell in order to accumulate its votes. All the cells together form an integral image. Now, the window containing the maximum number of points can be found by sliding the window over the integral image. This can be done in $O(m^2)$ time where m is the resolution of the grid. We set experimentally the sliding window to 0.1 of the person's bounding box height and the grid resolution to 100x100 pixels. The complexity of this algorithm is $O(s + m^2)$ which is much faster than $O(Ts^2)$ of Mean Shift, where T is the number of iterations.

3.4 Experiments

Most of the current method on human pose estimation, from still images, have relied on part-based models [10, 141, 186]. Through our experimental evaluation, we stress that holistic human pose estimation leads to high performance as well. In this section, we analyse our holistic model, evaluate on three datasets and compare it with recent approaches.

First, we present the results for estimating the parameters of the regression forest. We perform all the experiments only on the training images of the Image Parse [186] dataset to avoid parameter overfitting. Then, we compare our method with a related approach that relies on body part classification through forests on the KTH Football dataset [95]. In order to show the power of our model in comparison to part-based methods, we evaluate on the Image Parse dataset as well. Finally, we propose the new and challenging Volleyball dataset, which has noisy and low resolution data. We evaluate our approach on it and compare with the part-based method [186].

Evaluation metrics: In all experiments, we employ the PCP (percentage of correctly estimated parts) performance measure, which is the standard metric used in human pose estimation [60]. We distinguish two variants of the PCP score according to the literature [142]. In *strict* PCP score, the PCP score of a limb, defined by a pair of joints, is considered correct if the distance between *both* estimated joint locations and true limb joint locations is at most 50% of the length of the ground-truth limb, while the *loose* PCP score considers the *average* distance between the estimated joint locations and true limb joint locations. In this chapter, we mainly use make of the *loose* PCP score in order to keep up with the related work.

3.4.1 System Parameters

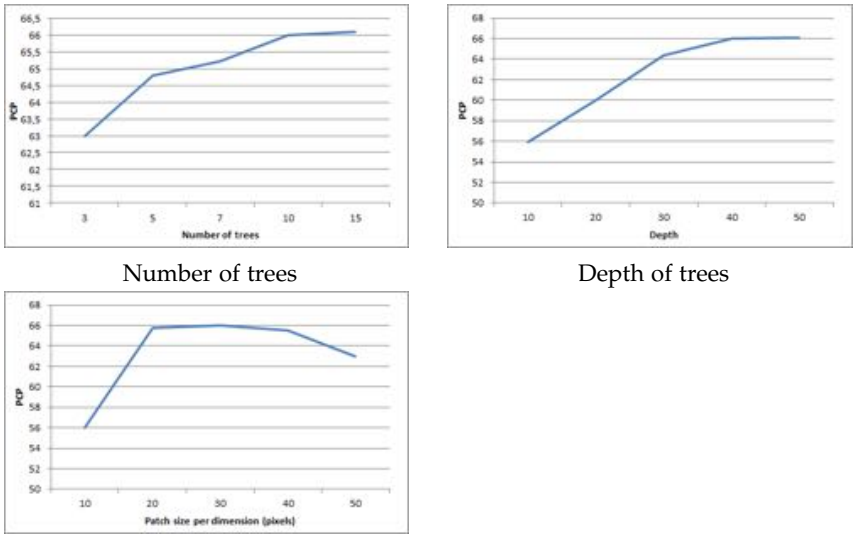
At first, we choose the parameters of the regression forest by evaluating on Image Parse [186]. We focus on determining the number and depth of the trees, as well as the size of the window of the image patch. Figure 3.2 presents the results.

Based on the results of the Figure 3.2, we have chosen to use 15 trees with a depth of 40. The trees are very deep due to the high variation in terms of appearance and motion of the human poses. The patch size is set to 30 pixels per dimension.

Finally, we have evaluated the prediction step with the Mean Shift and *dense-window* algorithm and we ended up with almost identical results.

3.4.2 Football Dataset

In this experiment, we compare our method with the part-based method which relies on classification forests [95]. In this work the forest classifies each pixel in the image as a specific body joint. Afterwards, a body prior model (i.e. pictorial structures) helps to improve the final result. The results are summarized in Table 3.1. For the method of Yang and Ramanan [186] which is based on pictorial structures, we have used the available on-line code.



Number of trees

Depth of trees

Image patch size

Figure 3.2: **Forest parameters**: We have estimated the parameters of the regression forest on the training dataset of the Image Parse dataset [186]. The number and the depth of treess, and the size of the image patch are explored.

Table 3.1: **KTH Football**: The evaluation with the *loose* PCP results is presented for different body parts.

	Head	Torso	Upper Arms	Lower Arms	Upper Legs	Lower Legs	Avg.
Our method	0.86	**0.98**	0.88	0.57	0.92	0.80	0.84
Yang&Ramanan [186]	0.84	0.98	0.86	0.55	0.89	0.73	0.80
Kazemi et al. [95]	0.94	0.96	0.90	0.69	0.94	0.84	0.87
Kazemi et al. [95] + Prior	**0.96**	0.98	**0.93**	**0.71**	**0.97**	**0.88**	**0.90**

For most of the body parts, we achieve similar results with the classification forest of [95]. In our formulation, we do not rely on a body prior model for smoothing the results. In Figure 3.3 some of our results on the KTH football dataset are presented.

3.4.3 Image Parse Dataset

The Image Parse dataset [186] is one of the most standard datasets for human pose estimation from images. We make use of it in different parts of the thesis. It includes images of humans with different appearance and pose (see in Figure 3.5). In Table 3.2, we present our results and compare with several part-based approaches.

Our method achieves similar results to the other approaches with the great difference that we use smaller amount of training data. We have used the set of 100 train images, which only flipped for doubling the training data, for our regression forest. This is significantly lower in contrast to Pischulin et al. ([141],[142]), where they train with 1000 images. Similarly, Johnson and Everingham [92] train with 10000 images. The reason for achieving similar

Table 3.2: **Image Parse**: The evaluation with the *strict* PCP results is presented for different body parts. Our method achieves competitive results with respect to the related work.

	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arms	Head	Fully body
Our method	85.8	**80.9**	**70.5**	55.0	22.5	71.3	63.5
Andriluka et al.[8]	86.3	66.3	60.0	54.6	35.6	72.7	59.2
Yang&Ramanan [186]	82.9	69.0	63.9	55.1	35.4	**77.6**	60.7
Pischulin et al. [141]	**92.2**	74.6	63.7	54.9	39.8	70.7	62.9
Pischulin et al. [142] + [141]	90.7	80.0	70.0	59.3	37.1	77.6	66.1
Johnson&Everingham [92]	87.6	74.7	67.1	**67.3**	**45.8**	76.8	**67.4**

results is that Random Forests can generalise to unknown poses. The only case where we have lower performance is at the lower arms due to the blurry input.

Figure 3.3: **KTH Football**: Qualitative results of our algorithm on some samples. The main feature of the dataset is the motion variation.

3.4.4 Volleyball Dataset

We propose the Volleyball dataset [1] for 2D human pose estimation. The dataset is composed of 800 training image of men and 205 testing images of women playing volleyball. We have used two different volleyball matches to create the dataset. The main feature of this dataset is the low quality and noisy image data. In Figure 3.5, we demonstrate some samples of the Volleyball dataset and the inferred body poses. By evaluating on this type of input data, we would like to highlight that our holistic model can cope with incomplete data.

We have evaluated our method on the Volleyball dataset using the PCP evaluation score. In order to compare with another approach, we have trained and tested the code of Yang and Ramanan [186]. The results are summarized in

[1] `http://campar.in.tum.de/Chair/SingleHumanPose`

Table 3.3: **Volleyball**: The evaluation with the *loose* PCP results is presented for different body parts. We only part where we lack of performance are the lower arms.

	Head	Torso	Upper Arms	Lower Arms	Upper Legs	Lower Legs	Avg.
Our method	**97.5**	**81.4**	**54.4**	19.3	**65.1**	**81.2**	**63.8**
Yang&Ramanan[186]	76.1	80.5	40.7	**33.7**	52.4	70.5	59.0

Table 3.3. We perform better for most of the body parts but we have achieved worse results for the lower arms. This happens because the lower arms are often fully occluded and then the forest predicts an average pose.

Figure 3.4: **Failure cases**: We present cases where our model wrongly predicted the body pose.

3.5 Conclusions

We have presented a holistic model for human pose estimation from 2D images. The model has been built on random forests and HOG features from image patches. It has been demonstrated that our model delivers promising results by evaluating on two standard datasets and comparing with other approaches. We have also introduced a challenging dataset which main feature is the noise and the low quality of image data. In all datasets, we have shown that our holistic approach can perform well and compete equally with the most recent part-based approaches. However, decoupling the process of feature designing and extraction from model learning creates limitations (Figure 3.4), which can

be overcome using deep learning. Convolutional Neural Networks, a very promising deep learning algorithm, have verified this claim by showing that training a classifier and learning the features at the same time can lead to high performance [99].

In the next chapter, we continue with the single-view human pose estimation, but address the problem of 2D human pose estimation using deep learning. The goal is to learn a regression model simultaneously with the features. Instead of using engineered features (e.g. HOG), we rely on raw data to learn features for the problem of human pose estimation.

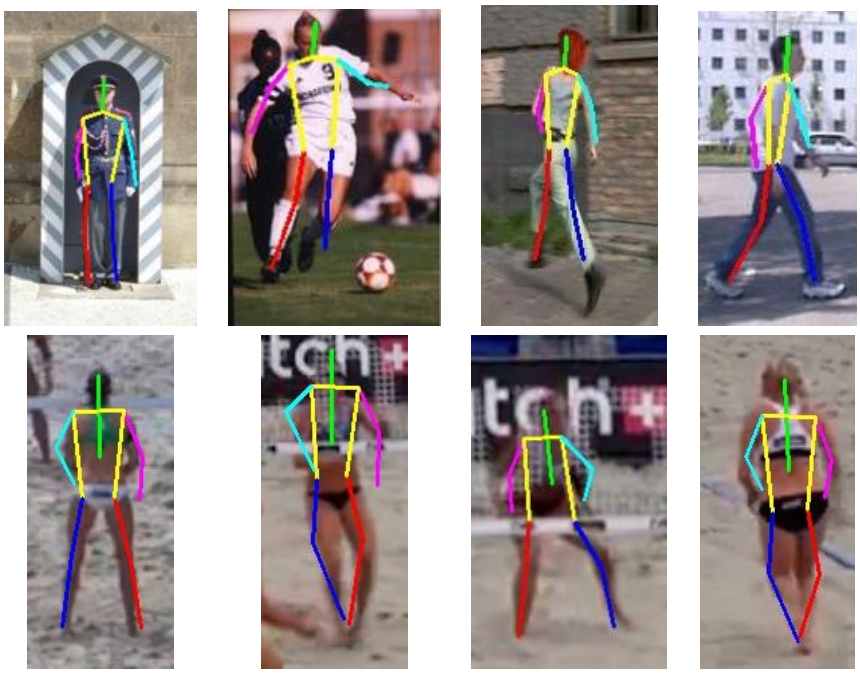


Figure 3.5: **More results**: Qualitative results of our algorithm on some samples from Image Parse (top row) and Volleyball (bottom row) datasets. The dataset has large appearance variation.

4

Deep Single-view Human Pose Estimation

We step up our efforts on single-view human pose estimation by exploring the advances of deep learning in computer vision. Deep learning is a new area of machine learning that models high-level abstractions in data modalities using deep architectures. The main idea is to learn representations from raw data using complex and non-linear transformations [22]. The most popular architectures for realizing deep learning are deep belief networks (DBNs) and convolutional deep neural networks (ConvNets) with many applications in computer vision, speech recognition and natural language processing (NLP). Both models are a type of neural network. Although they have been introduced a few decades ago, they have gained more reputation the last years mainly due to the advances of the hardware systems. Deep architectures require big amount of training data and consequently the training time can exponentially grow.

In this chapter, we consider the convolutional neural networks for tackling the problem of 2D human pose estimation. Convolutional neural networks (ConvNets) have successfully contributed to improve the accuracy of regression-based methods for computer vision tasks such as human pose estimation, landmark localization, and object detection. In regression tasks, the network optimization has been usually performed with the L2 loss and without considering the impact of outliers on the training process. Over the chapter, we examine the impact of the outliers on the ConvNet performance, applied on 2D human pose estimation. Then, we propose a robust loss function in comparison to the L2 loss for improving the performance and convergence. In addition to the robust loss, we introduce a coarse-to-fine model, which processes input images of progressively higher resolutions for improving the accuracy of the regressed values. At the last part of the chapter, our algorithm is evaluated on publicly available datasets, similar to Chapter 3, for demonstrating faster the convergence and better generalization of our robust loss function. Moreover, a comparison with related methods is presented, where we show that our

method achieves promising results.

4.1 Introduction to Deep Learning for Regression

Deep learning has played an important role in the computer vision field in the last few years. In particular, several methods have been proposed for challenging tasks, such as classification [94, 99], detection [71], categorization [191], segmentation [116], feature extraction [41, 153] and pose estimation [42]. State-of-the-art results in these tasks have been achieved with the use of Convolutional Neural Networks (ConvNets) trained with backpropagation [107]. Moreover, the majority of the tasks above are defined as classification problems, where the ConvNet is trained to minimize a softmax loss function [42, 94, 99]. Besides classification, ConvNets have been also trained for regression tasks such as human pose estimation [111, 174], object detection [171], facial landmark detection [169] and depth prediction [54]. In regression problems, the training procedure usually optimizes an $L2$ loss function plus a regularization term, where the goal is to minimize the squared difference between the estimated values of the network and the ground-truth. However, it is generally known that $L2$ norm minimization is sensitive to outliers, which can result in poor generalization depending on the amount of outliers present during training [83]. Without loss of generality, we assume that the samples are drawn from an unknown distribution and outliers are sample estimations that lie at an abnormal distance from other training samples in the objective space [124]. Within our context, outliers are typically represented by uncommon samples that are rarely encountered in the training data, such as rare body poses in human pose estimation, unlikely facial point positions in facial landmark detection or samples with imprecise ground-truth annotation. In the presence of outliers, the main issue of using the $L2$ loss in regression problems is that outliers can have a disproportionally high weight and consequently influence the training procedure by reducing the generalization ability and increasing the convergence time. As a result, the $L2$ loss is recommended for training data without outliers, but this is not a realistic scenario in computer vision problems.

We propose a loss function that is robust to outliers for training ConvNets on regression tasks. Our motivation originates from Robust Statistics, where the problem of outliers has been well-studied over the past decades, and several robust estimators have been proposed for reducing the influence of outliers in the process of model fitting [83]. Particularly in a ConvNet model, a robust estimator can be used in the loss function minimization, where training samples with unusually large errors are downweighted, such that they minimally influence the training procedure. It is worth noting that the training sample weighting provided by the robust estimator is done without any hard threshold between inliers and outliers. Furthermore, weighting training samples also conforms with the idea of curriculum [23] and self-paced [101] learning, where each training sample has different contribution to the minimization process, depending on its error. Nevertheless, the advantage in the use of a robust estimator, over the concept of curriculum or self-paced

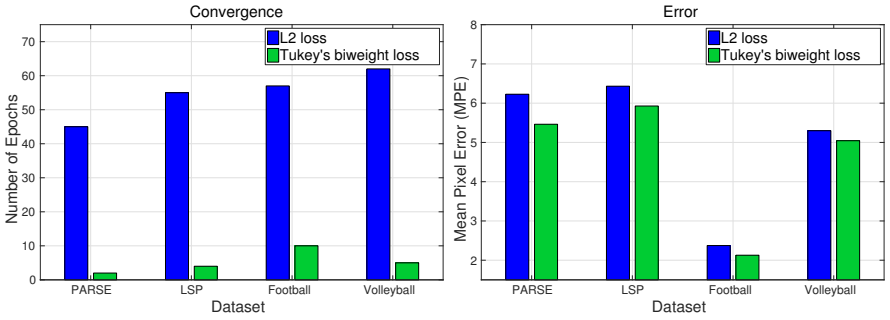


Figure 4.1: **Comparison of** *L*2 **and** *Tukey's biweight* **loss functions**: We compare our results (Tukey's biweight loss) with the standard *L*2 loss function on the problem of 2D human pose estimation (PARSE [187], LSP [91], Football [95] and Volleyball [15] datasets). On top, the convergence of *L*2 and Tukey's biweight loss functions is presented, while on the bottom, the graph shows the mean pixel error (MPE) comparison for the two loss functions. For the convergence computation, we choose as reference error, the smallest error using *L*2 loss (blue bars in bottom graph). Then, we look for the epoch with the closest error in the training using Tukey's biweight loss function.

learning, is that the minimization and weighting are integrated in a single function. Finally, the non-convexity of estimators robust to outliers, which is usually considered to be a problem for convex model-fitting problems, is not an issue in ConvNet training because it already minimizes a non-convex loss function.

We argue that training a ConvNet using a loss function that is robust to outliers results in faster convergence and better generalization (Figure A.1). We propose the use of *Tukey's biweight* function, a robust M-estimator, as the loss function for the ConvNet training in regression problems (Figure 4.5). Tukey's biweight loss function weights the training samples based on their residuals (we use the terms residual and error interchangeably, even if the terms are not identical, with both standing for the difference between the true and estimated values). Specifically, samples with unusually large residuals (i.e. outliers) are downweighted and consequently have small influence on the training procedure. In a similar way, inliers with insignificant residuals are also downweighted in order to prevent instabilities around local minima. Therefore, samples with residuals that are not too high or too small (i.e. inliers with significant residuals) have the largest influence on the training procedure. In the training, this influence is represented by the gradient magnitude of Tukey's biweight loss function, where in the backward step of backpropagation, the gradient magnitude of the outliers is low, while the gradient magnitude of the inliers is high except for the ones close to the local minimum. In Tukey's biweight loss function, there is no need to define a hard threshold between inliers and outliers. It only requires a tuning constant for suppressing the residuals of the outliers. We normalize the residuals with the median absolute deviation (MAD) [180], a robust approximation of variability, in order to preassign the tuning constant and consequently be free of parameters.

To demonstrate the advances of Tukey's biweight loss function, we rely on

the problem of 2D human pose estimation from still images. Similar to Chapter 3, we formulate the problem as a regression task where the goal is to predict the body joints, as a set of pixel coordinates. We learn a direct mapping between the input image and the body pose using a Convnet regressor. Moreover, our discriminative method is based on a novel coarse-to-fine model. The first stage in this model is based on an estimation of all output variables using the input image, and the second stage relies on an estimation of different subsets of the output variables using higher resolution input image regions extracted using the results of the first stage. As we later show in the experiments, where we evaluate our method on four publicly available datasets, the robust loss function allows for faster convergence and better generalization compared to the $L2$ loss; and 2. the coarse-to-fine model produces comparable to better results than the state-of-the-art in the four datasets above.

4.2 Related Work on Deep Regression

Training a convolutional neural network (ConvNet), in a supervised manner, requires a loss function to be minimized. The type of loss is determined in accordance with the addressed problem. For regression-based tasks, there have been proposed several deep learning approaches [151]. In this section, we discuss regression-based deep learning methods and position our model with respect to the related work.

A large number of regression-based deep learning algorithms have been recently proposed, where the goal is to predict a set of interdependent continuous values. For example, in object and text detection, the regressed values correspond to bounding boxes for localisation [87, 171], in human pose estimation, the values represent the positions of the body joints on the image plane [111, 140, 174], and in facial landmark detection, the predicted values denote the image locations of the facial points [169]. In all these cases, a ConvNet has been trained using an $L2$ loss function, without considering its vulnerability to outliers. Moreover, it is interesting to note that some regression methods combine the $L2$-based objective function with a classification function, which effectively results in a regularization of $L2$ and increases its robustness to outliers. For example, Zhang et al. [192] have introduced a ConvNet that is optimized for landmark detection and attribute classification. They have showed that the combination of softmax and $L2$ loss functions improves the network performance when compared to the minimization of $L2$ loss alone. Wang et al. [183] have used a similar strategy for the task of object detection, where they combine the bounding box localization (using an $L2$ norm) with object segmentation. The regularization of the $L2$ loss function has been also addressed by Gkioxari et al. [73], where the function being minimized comprises a body pose estimation term (based on $L2$ norm) and an action detection term. Finally, other methods have also been proposed to improve the robustness of the $L2$ loss to outliers, such as the use of complex objective functions in depth estimation [54] or multiple $L2$ loss functions for object generation [1]. However, to the best of our knowledge, none of the proposed approaches handles directly the presence of outliers during training with the use of a robust loss function,

like we propose in this paper.

Figure 4.2: **Our Results** Our results on 2D human pose estimation on the PARSE [187] dataset.

The main contribution of this chapter is the introduction of Tukey's biweight loss function for regression problems based on ConvNets. We focus on 2D human pose estimation from still images (Figure 4.2), and as a result our method can be classified as a holistic approach. In Chapter 3, we have presented related methods on 2D human pose estimation. Among the related work, there are deep learning approaches as well [140, 174]. Our proposed method is close to the cascade of ConvNets from [174]. However, we optimize a robust loss function instead of the $L2$ loss of [174] and empirically show that this loss function leads to more efficient training (i.e faster convergence) and better generalization results.

Next, we present our robust loss function and the coarse-to-fine model for 2D human pose estimation. We present the theoretical background of our method and also discuss the structure of our ConvNet.

4.3 Robust Deep Regression

In this section, we introduce the proposed robust loss function for training ConvNets on regression problems. Inspired by M-estimators from Robust Statistics [31], we propose the use of Tukey's biweight function as the loss to be to be minimized during the network training.

The input to the network is an image $\mathbf{x} : \Omega \rightarrow \mathbb{R}$ and the output is a real-valued vector $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ of N elements, with $y_i \in \mathbb{R}$. Given a training dataset $\{(\mathbf{x}_s, \mathbf{y}_s)\}_{s=1}^{S}$ of S samples, our goal is the training of a ConvNet, represented by the function $\phi(.)$, under the minimization of Tukey's biweight loss function with backpropagation [148] and stochastic gradient descent [35]. This training process produces a ConvNet with learnt parameters $\boldsymbol{\theta}$ that is effectively a mapping between the input image $\mathbf{x}$ and output $\mathbf{y}$, represented by:

$$\hat{\mathbf{y}} = \phi(\mathbf{x}; \boldsymbol{\theta}), \tag{4.1}$$

where $\hat{\mathbf{y}}$ is the estimated output vector. Next, we present the architecture of the network, followed by Tukey's biweight loss function. In addition, we introduce a coarse-to-fine model for capturing features in different image resolutions for improving the accuracy of the regressed values.

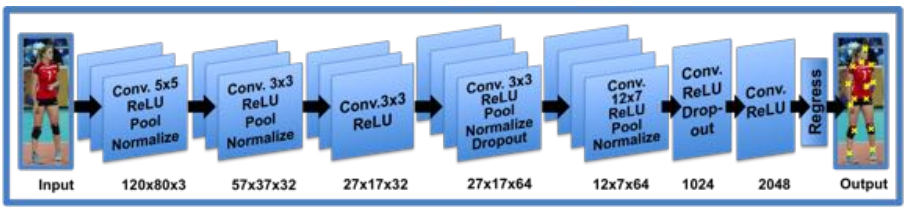


Figure 4.3: **Network and cascade structure**: Our network consists of five convolutional layers, followed by two fully connected layers. We use relative small kernels for the first two layers of convolution due to the smaller input image in comparison to [99]. Moreover, we use a small number of filters because we have observed that regression tasks required fewer features than classification [99].

4.3.1 Convolutional Neural Network and Architecture

A convolutional neural network (ConvNet) is essentially an artificial neural network (ANN) similar to a multilayer perceptron (MLP) [80], which has been inspired by the function of the visual cortex [65, 82]. It has been introduced for handwritten digit recognition [107], but is has been successfully applied on many other problems as well. The particular features of a ConvNet, in comparison to an ANN, is the sparse connectivity and shared weights. The sparse connectivity refers to the connection between neurons of adjacent layers, which is constrained on a local level. As a result, neurons are enforced to develop spatial relations though the receptive fields. A receptive filed, which can be also called filter, has different sizes. Moreover, it can be local or global based on the image and filter size. The filters in a ConvNet are formed by weights that are shared for the whole image. This is the second feature of ConvNet that sustains the number of the parameters to a few millions. As a result, training a ConvNet also corresponds to learning kernels for extracting features. The first layers usually learn low-level features (e.g. edges and lines), while the deep levels higher-level features (e.g. faces).

A ConvNet is composed of several layers of convolution as well as other type of layers; and the training is achieved with backpropagation [107]. Below, we summarize the standard layer types of a ConvNet, which we also use in our method.

Convolutional layer: This is the part where actually learning takes place. The parameters (i.e. weights) of a filter are learnt through training with back-propagation. A Convolutional layer has usually many filters that also define the capacity of the network.

Activation layer: The standard activation function is the rectified linear unit (ReLU) [126] that is given by $f(x) = \max(0, x)$. There can be also used a hyperbolic tangent or sigmoid activation function like in neural networks.

Pooling layer: Pooling is used for down-sampling the input image and consequently reducing the computations. It also introduces a kind of translation invariance only for classification problems. In order to down-sample the input image, it usually computes the max or average feature value of an image region.

Dropout layer: Dropout is a recent method for avoiding overfitting and also regularizing the network [163]. The idea is that in the input layer a number of neurons (usually 50% of the neurons) is randomly deactivated.

Loss layer: This the last layer of the networks and it is defined according to the task. In classification tasks, a softmax loss is usually is chosen for predicting classes, while in regression tasks the $L2$ loss is the appropriate loss for predicting real-valued labels.

ConvNet Regressor: Our network takes as input an RGB image and regresses a N-dimensional vector of continuous values. As it is presented in Figure 4.3, the architecture of the network consists of five convolutional layers, followed by two fully connected layers and the output that represents the regressed values. The structure of our network is similar to Krizhevsky's [99], but we use smaller kernels and fewer filters in the convolutional layers. Our fully connected layers are smaller as well, but as we demonstrate in the experimental section, the smaller number of parameters is sufficient for the regression tasks considered in this paper. In addition, we apply local contrast normalization, as proposed in [99], before every convolutional layers and max-pooling after each convolutional layer in order to reduce the image size. We argue that the benefits of max-pooling, in terms of reducing the computational cost, outweighs the potential negative effect in the output accuracy for regression problems. Moreover, we use dropout [163] in the fourth convolutional and first fully connected layers to prevent overfitting. The activation function for each layer is the rectified linear unit (ReLU), except for the last layer, which uses a linear activation function for the regression. Finally, we use our robust loss function for training the network of Figure 4.3.

4.3.2 Robust Loss Function

The training process of the ConvNet is accomplished through the minimization of a loss function that measures the error between ground-truth and estimated values (i.e. the residual). In regression problems, the typical loss function used is the $L2$ norm of the residual, which during backpropagation produces a gradient whose magnitude is linearly proportional to this difference. This means that estimated values that are close to the ground-truth (i.e. inliers) have little influence during backpropagation, but on the other hand, estimated values that are far from the ground-truth (i.e. outliers) can bias the whole training process given the high magnitude of their gradient, and as a result adapt the ConvNet to these outliers while deteriorating its performance for the inliers. Recall that we consider the outliers to be estimations from uncommon training

samples that lie at an abnormal distance from other sample estimations in the objective space. This is a classic problem addressed by Robust Statistics [31], which is solved with the use of a loss function that weights the training samples based on the residual magnitude. The main idea is to have a loss function that has low values for small residuals, and then usually grows linearly or quadratically for larger residuals up to a point when it saturates. This means that only relatively small residuals (i.e. inliers) can influence the training process, making it robust to the outliers that are mentioned above.

There are many robust loss functions that could be used, but we focus on *Tukey's* biweight function [31] because of its property of suppressing the influence of outliers during backpropagation (Fig. 4.5) by reducing the magnitude of their gradient close to zero. Another interesting property of this loss function is the soft constraints that it imposes between inliers and outliers without the need of setting a hard threshold on the residuals. Formally, we define a residual of the i^{th} value of vector $\mathbf{y}$ by:

$$r_i = y_i - \hat{y}_i, \tag{4.2}$$

where $\hat{y}_i$ represents the estimated value for the i^{th} value of $\mathbf{y}$, produced by the ConvNet. Given the residual r_i, Tukey's biweight loss function is defined as:

$$\rho(r_i) = \begin{cases} \frac{c^2}{6}\left[1-(1-(\frac{r_i}{c})^2)^3\right] & \text{, if } |r_i| \leq c \\ \frac{c^2}{6} & \text{, otherwise} \end{cases}, \tag{4.3}$$

where c is a tuning constant, which if is set to $c = 4.6851$, gives approximately 95% asymptotic efficiency as $L2$ minimization on the standard normal distribution of residuals. However, this claim stands for residuals drawn from a distribution with unit variance, which is an assumption that does not hold in general. Thus, we approximate a robust measure of variability from our training data in order to scale the residuals by computing the median absolute deviation (MAD) [83]. MAD measures the variability in the training data and is estimated as:

$$\text{MAD}_i = \underset{k\in\{1,...,S\}}{\text{median}}\left(\left|r_{i,k} - \underset{j\in\{1,...,S\}}{\text{median}}(r_{i,j})\right|\right), \tag{4.4}$$

for $i \in \{1, ..., N\}$ and the subscripts k and j index the training samples. The MAD_i estimate acts as a scale parameter on the residuals for obtaining unit variance. By integrating MAD_i to the residuals, we obtain:

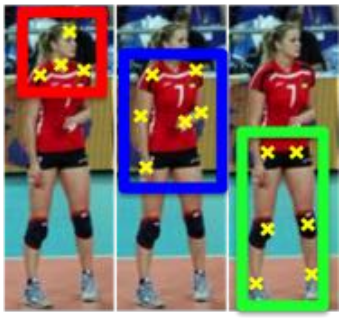
$$r_i^{\text{MAD}} = \frac{y_i - \hat{y}_i}{1.4826 \times \text{MAD}_i}, \tag{4.5}$$

where we scale MAD_i by 1.4826 in order to make MAD_i an asymptotically consistent estimator for the estimation of the standard deviation [83]. Then, the scaled residual r_i^{MAD} in Eq. (4.5) can be directly used by Tukey's biweight loss function Eq. (4.3). We fix the tuning constant based on MAD scaling and thus our loss function is free of parameters. The final objective function based

on Tukey's loss function and MAD_i estimate is given by:

$$E = \frac{1}{S} \sum_{s=1}^{S} \sum_{i=1}^{N} \rho \left(r_{i,s}^{\text{MAD}} \right). \tag{4.6}$$

We illustrate the functionality of Tukey's biweight loss function in Fig. 4.5, which shows the loss function and its derivative as a function of sample residuals in a specific training problem. This is an instance of the training for the LSP [91] dataset that is further explained in the experiments.



Coarse-to-Fine Model

Figure 4.4: **Coarse-to-fine Model**: The three images (Coarse-to-Fine Model) show the $C = 3$ image regions and respective subsets of $\hat{\mathbf{y}}$ used by the cascade of ConvNets in the proposed coarse-to-fine model.

4.3.3 Coarse-to-Fine Model

We adopt a coarse-to-fine model, where initially a single network $\phi(.)$ of Eq. (4.1) is trained from the input images to regress all N values of $\hat{\mathbf{y}}$, and then separate networks are trained to regress subsets of $\hat{\mathbf{y}}$ using the output of the single network $\phi(.)$ and higher resolution input images. Effectively, the coarse-to-fine model produces a cascade of ConvNets, where the goal is to capture different sets of features in high resolution input images, and consequently improve the accuracy of the regressed values. Similar approaches have been adopted by other works [54, 173, 174] and shown to improve the accuracy of the regression. Most of these approaches refine each element of $\hat{\mathbf{y}}$ independently, while we employ a different strategy of refining subsets of $\hat{\mathbf{y}}$. We argue that our approach constrains the search space more and thus facilitates the optimization.

More specifically, we define C image regions and subsets of $\hat{\mathbf{y}}$ that are included in theses regions (Fig. 4.3). Each image region $\mathbf{x}^c$, where $c \in \{1, ..., C\}$, is cropped from the original image $\mathbf{x}$ based on the output of the single ConvNet of Eq. (4.1). Then the respective subset of $\hat{\mathbf{y}}$ that falls in the image region c is transformed to the coordinate system of this region. To define a meaningful set of regions, we rely on the specific regression task. For instance, in 2D human pose estimation, the regions can be defined based on the body anatomy (e.g. head and torso or left arm and shoulder); similarly, in facial landmark

localization the regions can be defined based on the face structure (e.g. nose and mouth). This results in training C additional ConvNets $\{\phi^c(.)\}_{c=1}^{C}$ whose input is defined by the output of the single ConvNet $\phi(.)$ of Eq. (4.1). The refined output values from the cascade of ConvNets are obtained by:

$$\hat{\mathbf{y}}_{ref} = \text{diag}(\mathbf{z})^{-1} \sum_{c=1}^{C} \phi^c \left(\mathbf{x}^c; \boldsymbol{\theta}^c, \hat{\mathbf{y}}(l^c)\right), \tag{4.7}$$

where $l^c \subset \{1, 2, \ldots, N\}$ indexes the subset c of $\hat{\mathbf{y}}$, the vector $\mathbf{z} \in \mathbb{N}^N$ has the number of subsets in which each element of $\hat{\mathbf{y}}$ is included and $\boldsymbol{\theta}^c$ are the learnt parameters. Every ConvNet of the cascade regresses values only for the dedicated subset l^c, while its output is zero for the other elements of $\hat{\mathbf{y}}$. To train the ConvNets $\{\phi^c(.)\}_{c=1}^{C}$ of the cascade, we use the same network structure that is described in Sec. 4.3.1 and the same robust loss function of Eq. (4.6). Finally, during inference, the first stage of the cascade uses the single ConvNet $\phi(.)$ of Eq. (4.1) to produce $\hat{\mathbf{y}}$, which is refined by the second stage of the cascade with the ConvNets $\{\phi^c(.)\}_{c=1}^{C}$ of Eq. (4.7). The predicted values $\hat{\mathbf{y}}_{ref}$ of the refined regression function are normalized back to the coordinate system of the image $\mathbf{x}$.

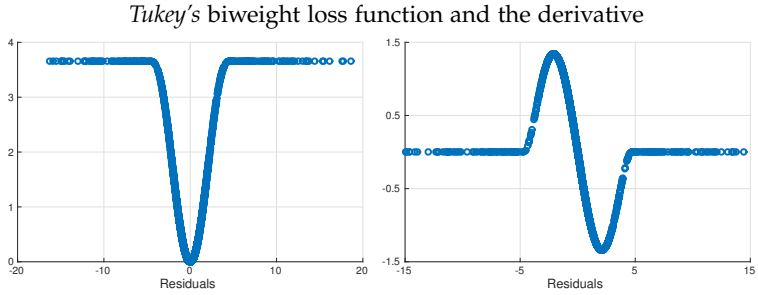


Figure 4.5: **Tukey's biweight loss function**: Tukey's biweight loss function (left) and its derivative (right) as a function of the training sample residuals.

4.3.4 Training Details

The input RGB image to the network has resolution 120×80, as it is illustrated in Fig. 4.3. Moreover, the input images are normalized by subtracting the mean image estimated from the training images[1]. We also use data augmentation in order to regularize the training procedure. To that end, each training sample is rotated and flipped (50 times) as well as a small amount of Gaussian noise is added to the ground-truth values $\mathbf{y}$ of the augmented data. Furthermore, the same data is shared between the first cascade stage for training the single ConvNet $\phi(.)$ and second cascade stage for training the ConvNets $\{\phi^c(.)\}_{c=1}^{C}$. Finally, the elements of the output vector of each training sample are scaled to

[1]We have also tried the normalization based on the division by the standard deviation of the training data, but we did not notice any particular positive or negative effect in the results.

the range $[0, 1]$. Concerning the network parameters, the learning rate is set to 0.01, momentum to 0.9, dropout to 0.5 and the batch size to 230 samples.

The initialisation of the ConvNets' parameters is performed randomly, based on an unbiased Gaussian distriubtion with standard deviation 0.01, with the result that many outliers can occur at the beginning of training. To prevent this effect that could slow down the training or exclude samples at all from contributing to the network's parameter update, we increase the MAD values by a factor of 7 for the first 50 training iterations (around a quarter of an epoch). Increasing the variability for a few iterations helps the network to quickly reach a more stable state. Note that we have empirically observed that the number of iterations needed for this MAD adjustment does not play an important role in the whole training process and thus these values are not hard constraints for convergence.

4.4 Experiments

We evaluate Tukey's biweight loss function for the problem of 2D human pose estimation from still images. For that purpose, we have selected four publicly available datasets, namely PARSE [187], LSP [91], Football [95] and Volleyball [15]. All four datasets include sufficient amount of data for training the ConvNets, except for PARSE which has only 100 training images. For that reason, we have merged LSP and PARSE training data, similar to [91], for the evaluation on the PARSE dataset. For the other three datasets, we have used their training data independently. In all cases, we train our model to regress the 2D body skeleton as a set of joints that correspond to pixel coordinates (Figure 4.9). We assume that each individual is localized within a bounding box with normalized body pose coordinates. Our first assumption holds for all four datasets, since they include cropped images of the individuals, while for the second we have to scale the body pose coordinates in the range $[0, 1]$. Moreover, we introduce one level of cascade using three parallel networks ($C = 3$) based on the body anatomy for covering the following body parts: 1) head - shoulders, 2) torso - hands, and 3) legs (see Figure 4.4). In the first part of the experiments, a baseline evaluation is presented, where Tukey's biweight and the standard $L2$ loss functions are compared in terms of convergence and generalization. Then, we compare the results of our proposed coarse-to-fine model with state-of-the-art methodologies.

Experimental setup: The experiments have been conducted on an Intel i7 machine with a GeForce GTX 980 graphics card. The training time varies slightly between the different datasets, but in general it takes 2-3 hours to train a single ConvNet. This training time scales linearly for the case of the cascade. Furthermore, the testing time of a single ConvNet is 0.01 seconds per image. Regarding the implementation of our algorithm, basic operations of the ConvNet such as convolution, pooling and normalization are based on MatConvNet [179].

Evaluation metrics: We rely on the mean pixel error (MPE) to measure the performance of the ConvNets. In addition, we employ once more the PCP (percentage of correctly estimated parts) performance measure using the

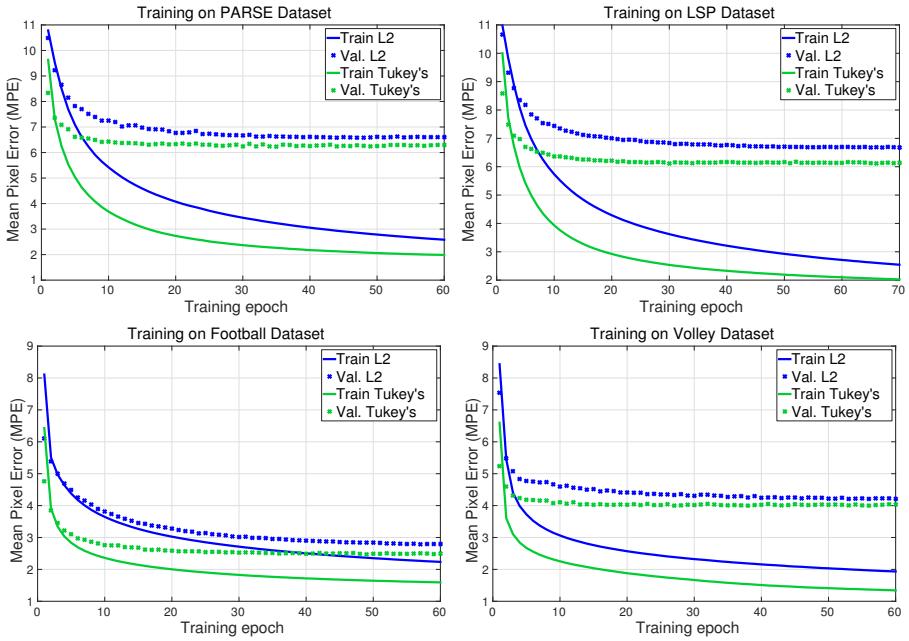


Figure 4.6: **Comparison of** $L2$ **and** *Tukey's biweight* **loss functions**:In all datasets (PARSE [187], LSP [91], Football [95] and Volleyball [15]), *Tukey's biweight* loss function shows, on average, faster convergence and better generalization than $L2$. Both loss functions are visualised for the same number of epochs.

two variants of *strict* and *loose* PCP score (defined in Chapter 3). During the comparisons with other methods, we explicitly indicate which version of the PCP score is used (Table 4.1).

4.4.1 Baseline Evaluation

In the first part of the evaluation, the convergence and generalization properties of Tukey's biweight loss functions are examined using the single ConvNet $\phi(.)$ of Eq. (4.1), without including the cascade. We compare the results of the robust loss with $L2$ loss using the same settings and training data of PARSE [187], LSP [91], Football [95] and Volleyball [15] datasets. To that end, a 5-fold cross validation has been performed by iteratively splitting the training data of all datasets (none of the datasets includes by default a validation set), where the average results are shown in Figure 4.6. Based on the results of the cross validation which is terminated by early stopping [108], we have selected the number of training epochs for each dataset. After training by using all training data for each dataset, we have compared the convergence and generalization properties of Tukey's biweight and $L2$ loss functions. For that purpose, we choose the lowest MPE of $L2$ loss and look for the epoch with the closest MPE after training with Tukey's biweight loss function. The results are summarized in Figure A.1 for each dataset. It is clear that by using Tukey's biweight loss, we obtain notably faster convergence (note that on the PARSE dataset it is

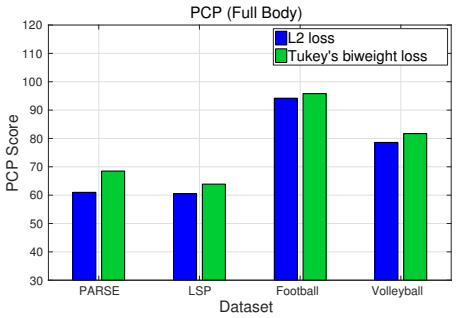


Figure 4.7: **PCP Comparison of** $L2$ **and** *Tukey's* **biweight loss functions**: The PCP scores of the full body as presented as a complementary metric of MSE from Figure A.1. The scores in PARSE [187] and LSP [91] datasets correspond to the *strict* PCP, while in Football [95] and Volleyball [15] to *loose* PCP in order to keep up with the literature in the comparisons.

20 times faster). This speed-up can be very useful for large-scale regression problems, where the training time usually varies from days to weeks. Besides faster convergence, we also obtain better generalization, as measured by the error in the validation set, using our robust loss function (see Figure A.1). More specifically, we achieve 12% smaller MPE using Tukey's biweight loss functions in two out of four datasets (i.e PARSE and Football), while we are around 8% better with the LSP and Volleyball datasets. In addition, we present the full body PCP scores in Figure 4.7, since this is the most common evaluation metric in human pose estimation, and similar conclusions can be drawn compared to the MSE error.

4.4.2 Comparison with other Methods

In this part, we evaluate our robust loss functions using the coarse-to-fine model represented by the cascade of ConvNets (Figure 4.4), presented in Sec. 4.3.3, and compare our results with the state-of-the-art from the literature, on the four aforementioned datasets (PARSE [187], LSP [91], Football [95] and Volleyball [15]). For the comparisons, we use the *strict* and *loose* PCP scores, depending on which evaluation metric was used by the state-of-the-art. The results are summarized in Table 4.1, where the first row of each evaluation shows our result using a single ConvNet $\phi(.)$ of Eq. (4.1) and the second row, the result using the cascade of ConvNets $\{\phi^c(.)\}_{c=1}^{C}$ of Eq. (4.7), where $C = 3$.

PARSE: This is a well-known dataset to assess 2D human pose estimation methodologies and thus we are able to show the results from most of the current state-of-the-art, as displayed in Table 4.1a. While our result is 68.5% for the full body regression using a single ConvNet, our final score is improved by around 5% with the cascade of ConvNets. We achieve the best score in the full body regression as well as in most body parts. Closer to our performance is another deep learning method by Ouyang et al. [134] that builds on part-based models and deep part detectors. The rest of the compared methods are also part-based, but our holistic model is simpler to implement and at the same

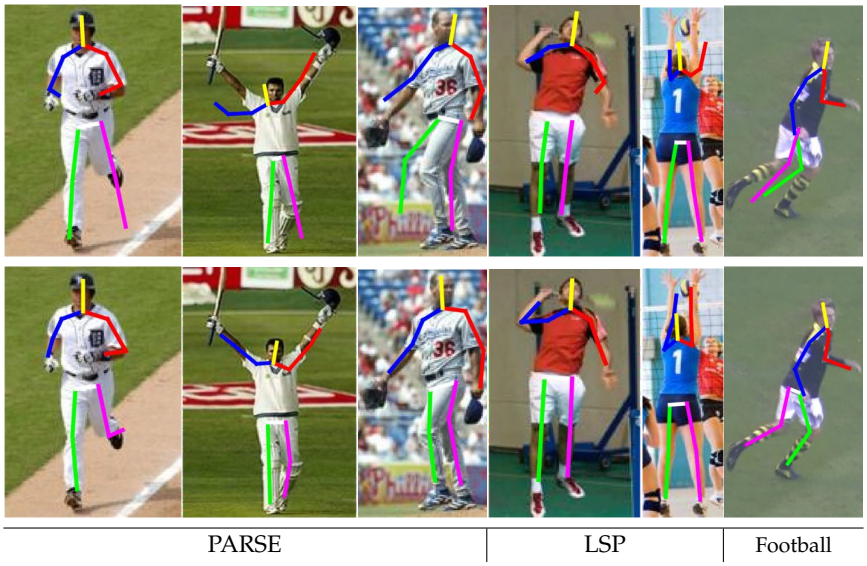


Figure 4.8: **Model refinement**: Results produced by our proposed method before (top row) and after (bottom row) the refinement with the cascade for the PARSE [187], LSP [91] and Football [95] datasets. We train $C = 3$ ConvNets for the cascade $\{\phi^c(.)\}_{c=1}^{C}$, based on the output of the single ConvNet $\phi(.)$.

time is shown to perform better (Figure 4.2 and 4.8).

LSP: Similar to PARSE, this is another standard dataset to assess human pose estimation methodologies. In LSP dataset, our approach shows a similar performance, compared to the PARSE dataset, using a single ConvNet or a cascade of ConvNets. In particular, the PCP score using one ConvNet increases again by around 5% with the cascade of ConvNets, from 63.9% to 68.8% for the full body evaluation (Table 4.1b). The holistic approach of Toshev et al. [174] is also a cascade of ConvNets, but it relies on $L2$ loss minimization and a different network structure. On the other hand, the Tukey's biweight loss being minimized in the training of our network brings better results in combination with the cascade. Note also that we have used 4 ConvNets in total for our model in comparison to the 29 networks used by Toshev et al. [174]. Moreover, considering the performance with respect to body parts, the best PCP scores are shared between our method and the one of Chen & Yuille [42]. The part-based model of Chen & Yuille [42] scores best for the full body, head, torso and arms, while we obtain the best scores on the upper and lowers legs. We show some results on this dataset in Figure 4.8 and 4.9.

Football: This dataset has been introduced by Kazemi et al. [95] for estimating the 2D pose of football players. In this dataset, our results (Table 4.1c) using one ConvNet are almost optimal (with a PCP score of 95.8%) and thus the improvement using the cascade is smaller in comparison to the two datasets above. However, it is important to notice that effective refinements are achieved with the use of the cascade of ConvNets, as demonstrated in Figure 4.8 and 4.9.

Table 4.1: **Comparison with other approaches**: We compare our results using one ConvNet (first row in each dataset) and the cascade of ConvNets (second row). The scores of the other methods are the ones reported in their original papers.

(a) **PARSE Dataset** The evaluation metric on PARSE dataset [187] is the *strict* PCP score.

Method	Head	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arm	Full Body
Ours	78.5	95.6	82.0	75.6	61.5	36.6	68.5
Ours (cascade)	**91.7**	**98.1**	**84.2**	**79.3**	66.1	41.5	**73.2**
Andriluka et al. [8]	72.7	86.3	66.3	60.0	54.6	35.6	59.2
Yang&Ramanan [187]	82.4	82.9	68.8	60.5	63.4	42.4	63.6
Pishchulin et al. [142]	77.6	90.7	80.0	70.0	59.3	37.1	66.1
Johnson et al. [91]	76.8	87.6	74.7	67.1	67.3	45.8	67.4
Ouyang et al. [134]	89.3	89.3	78.0	72.0	**67.8**	**47.8**	71.0

(b) **LSP Dataset** The evaluation metric on LSP dataset [91] is the *strict* PCP score.

Method	Head	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arm	Full Body
Ours	72.0	91.5	78.0	71.2	56.8	31.9	63.9
Ours (cascade)	83.2	92.0	**79.9**	**74.3**	61.3	40.3	68.8
Toshev et al. [174]	-	-	77.0	71.0	56.0	38.0	-
Kiefel&Gehler [97]	78.3	84.3	74.5	67.6	54.1	28.3	61.2
Yang&Ramanan [187]	79.3	82.9	70.3	67.0	56.0	39.8	62.8
Pishchulin et al. [142]	85.1	88.7	78.9	73.2	61.8	45.0	69.2
Ouyang et al. [134]	83.1	85.8	76.5	72.2	63.3	46.6	68.6
Chen&Yuille [42]	**87.8**	**92.7**	77.0	69.2	**69.2**	**55.4**	**75.0**

(c) **Football Dataset** The evaluation metric on Football dataset [95] is the *loose* PCPscore.

Method	Head	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arm	Full Body
Ours	97.1	99.7	99.0	98.1	96.2	87.1	95.8
Ours (cascade)	**98.3**	**99.7**	**99.0**	**98.1**	**96.6**	**88.7**	**96.3**
Yang&Ramanan [187]	97.0	99.0	94.0	80.0	92.0	66.0	86.0
Kazemi et al. [95]	96.0	98.0	97.0	88.0	93.0	71.0	89.0

(d) **Volleyball Dataset** The evaluation metric on Volleyball dataset [15] is the *loose* PCP score.

Method	Head	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arm	Full Body
Ours	90.4	**97.1**	**86.4**	**95.8**	74.0	58.3	**81.7**
Ours (cascade)	89.0	95.8	84.2	94.0	**74.2**	**58.9**	81.0
Yang&Ramanan [187]	76.1	80.5	52.4	70.5	40.7	33.7	56.0
Belagiannis et al. [15]	**97.5**	81.4	65.1	81.2	54.4	19.3	60.2

Figure 4.9: **Additional results**: Samples of our results on 2D human pose estimation are presented for the LSP [91] (first row), Football [95] (second row) and Volleyball [15] (third row) datasets.

Volleyball: Similar to the Football dataset [95], our results on the Volleyball dataset are already quite competitive using one ConvNet (Table 4.1d), with a PCP score of 81.7%. On this dataset, the refinement step has a negative impact to our results (Table 4.1d). We attribute this behaviour to the interpolation results of the cropped images, since the original images have low resolution (last row of Figure 4.9). Further results can be found in Appendix A.

4.5 Conclusions

In this chapter, we have introduced *Tukey's biweight* loss function for the robust optimization of ConvNets in regression-based problems. Using 2D human pose estimation as testbed, we have empirically shown that optimizing with this loss function, which is robust to outliers, results in faster convergence and better generalization compared to the standard $L2$ loss, which is a common loss function used in regression problems. The proposed cascade of ConvNets has also heped to improve the accuracy of the final regression result. The combination of our robust loss function with the cascade of ConvNets pro-

duces comparable or better results than the recent approaches (including deep learning methods) in all four evaluation datasets on 2D human pose estimation.

We have considered the idea of deep learning applied on single-view 2D human pose estimation. From our results, It is clear that simultaneously learning features from raw data and training a classifier results in much better performance than combining engineered features with a classifier, including our approach with the regression forest in Chapter 3. We have observed that ConvNets and deep learning in general can improve the current standards in human pose estimation as well as in many other vision tasks [151].

In the last two chapters, we have seen that a discriminative method delivers very promising results, given sufficient amount of training data that captures the target distribution. However, a sufficient amount of training data is not always available. In the next chapter, we shift our research to multi-view human pose estimation. The problem is more complex because of the multiple views and inference in 3D space and thus requires more training data for learning a model. For that reasons, we consider generative models or a combination between generative and discriminative models.

5 Multi-View Human Pose Estimation

In the two previous chapters, the problem of 2D human pose estimation from a single-view has been addressed. In this chapter, we continue to 3D human pose estimation, given a multi-view camera system. Furthermore, we consider to have multiple individuals in our scene that interact with each other. The transition from single to multiple human pose estimation and from the 2D to 3D space is challenging due to a much larger state space, occlusions and across-view ambiguities when not knowing the identity of the humans in advance. To address these problems, we propose a novel part-based model for 3D pose estimation of multiple humans from multiple views. In this method, we first create a reduced body part search space by triangulation of corresponding pairs of parts, obtained by part detectors for each camera view. In order to resolve ambiguities of wrong and mixed parts of multiple humans after triangulation and also those coming from false positive detections, we introduce a 3D pictorial structures (3DPS) model. The 3DPS model is a generative model that relies on 3D body part observation and the prior model that it is learnt from training data. Our model builds on a conditional random field (CRF) with multi-view unary potentials and a prior model that is integrated into pairwise and ternary potential functions. To balance the potentials' influence, the model parameters are learnt using a Structured Support Vector Machine (S-SVM).

We consider the shift from a discriminative to generative model very beneficial as a consequence of the high dimensional input and output space. In a generic multi-view setup, there is large appearance and body posture variation which obstructs the formation of a discriminative 3D human model. Instead, our generative model learns separately the body appearance as a set of parts, while the body posture is modelled using pairs or triplets of body parts. The 3DPS model is generic and applicable to both single and multiple human pose estimation. In our experimental section, we demonstrate the model properties in both cases by evaluating the 3DPS model in different environments and

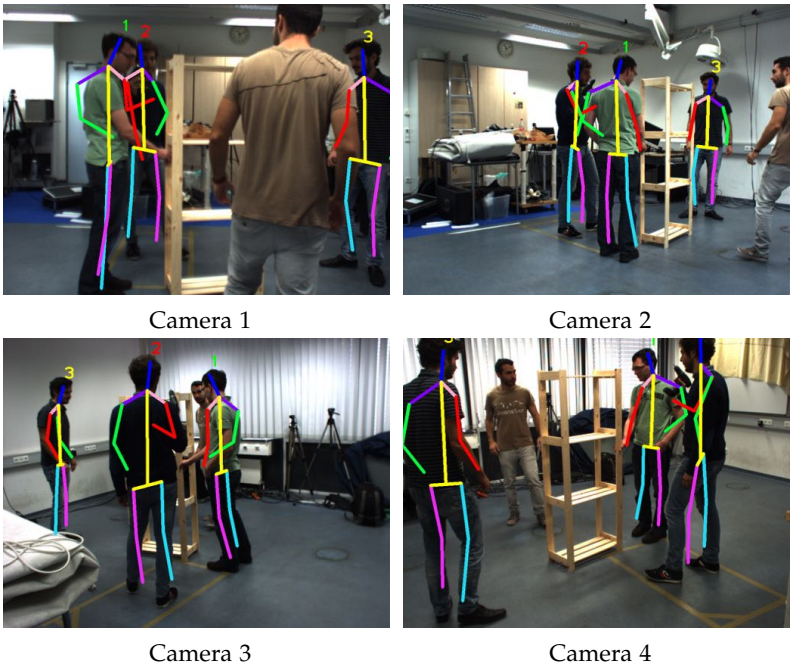


Figure 5.1: **Shelf dataset**: Our results on 3D pose estimation of multiple individuals projected in 4 out of 5 views of the Shelf dataset [16].

multi-view scenarios as well.

5.1 Introduction

Determining the 3D human body pose has been of particular interest, because it facilitates many applications such as human tracking, motion capture and analysis, activity recognition and human-computer interaction. Depending on the input modalities and number of employed sensors different methods have been proposed for single human 3D pose estimation [5, 9, 40, 121, 157, 161]. Nevertheless, estimating jointly the 3D pose of multiple humans from multi-views, remains an open problem (Figure 5.1). In this chapter, we tackle this problem for real-life environments.

We assume a multi-view setup that is calibrated and the projection matrices are available. Moreover, it is assumed that the individuals can be localized by means of tracking [24] or detection [57]. Our task is to perform 3D pose estimation of multiple individuals, where the body pose is defined as a set of parts and each part has a state that is described by a rotation and translation [16]. This task is complex given the plethora of body part states and individuals in the 3D space. Instead of exploring a large state space of all possible translations and rotations of the human body parts in 3D space, we propose a more efficient approach. We create a set of 3D body part hypotheses by triangulation of corresponding body joints sampled from the posteriors

of 2D body part detectors in all pairs of camera views. In this way, our task becomes simpler and requires inferring a correct human body pose from a set of 3D body part hypotheses without exploring all possible rotations and translations of the body parts. However, we have to consider that the identity of the individuals is not available in advance (when building the reduced state space). Another problem (i.e. [5, 40]) is the separation between left-right and front-back of the body anatomy because of the different camera views. Given all these constraints, it is easy to understand that the state space of body parts hypotheses can include many type of incorrect hypotheses other than the correct ones.

In order to address these challenges, we introduce a 3D pictorial structures (3DPS) model that infers body poses of multiple humans from a reduced state space of body part 3D hypotheses. The 3DPS model is based on a conditional random field (CRF) in which a random variable corresponds to a body part. In our model, a body part is defined from three parameters that stand for its 3D position. The unary potential functions are computed from the confidence of the 2D part-based detectors and reprojection error of the corresponding body parts. We propose additionally the visibility unary potential for modelling occlusions and resolving geometrical ambiguities. Furthermore, we introduce the temporal consistence unary term for constraining the 3D body part hypotheses with respect to the inferred body poses. The pairwise and ternary potential functions integrate a human body prior in which the relation between the body parts is modelled. We constrain the symmetric body parts to forbid collisions in 3D space by introducing an extra pairwise collision potential. Since we employ multiple potential functions, the necessity to weight them correctly arises. For that reason, we rely on a Structured SVM (S-SVM) [176] to learn the parameters of the model. Finally, the inference on our graphical model is performed using belief propagation. We parse the 3D pose of each individual by first localizing it and then sampling from the marginal distributions. Our only assumption is to have every body part correctly detected from at least two views in order to recover the part during triangulation. We build our model on our earlier work on 3D pictorial structures [16], but with a different body part parametrisation. Instead of defining the body part in terms of 3D position and orientation as in [16], we keep only the position parameters and implicitly encode the orientation in the factor graph. The 3DPS model is generic and applicable to both single and multiple human pose estimation. Moreover, inference of multiple human skeletons does not deteriorate despite the ambiguities, which are introduced during the creation of the state space.

In this chapter our contribution is the 3D pictorial structures (3DPS) model that can handle multiple humans using multi-view potential functions. To that end, we learn the parameters of our model with a Structured SVM (S-SVM) formulation. We also introduce a discrete state space for fast inference using 2D part detectors, instead of exploring a finely discretized 3D space. Very importantly, we do not assume that we have information about the identity of the humans in advance. Experimental results on HumanEva-I [159], KTH Multiview Football II [40], Campus [24] and Shelf [16] datasets demonstrate state-of-the-art results in comparison to related work, for single and multiple

3D human pose estimation. The Campus dataset [24] pre-exists, but we have introduced 3D body pose annotation for it in [16]. In addition, we have introduced the Shelf dataset in [16], due to the lack of standard datasets for multiple human pose estimation from multiple views.

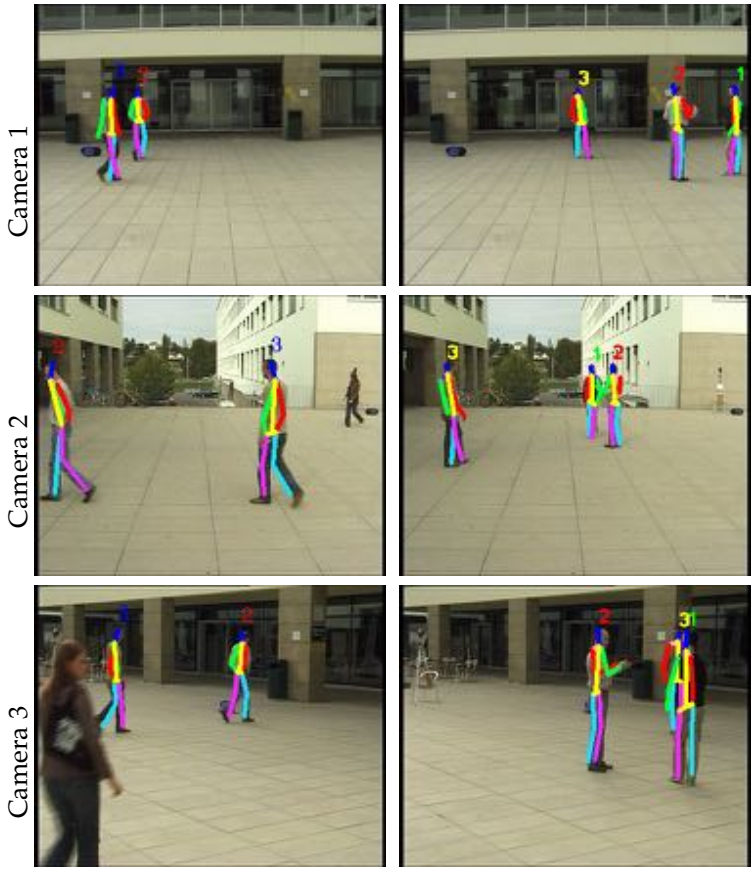


Figure 5.2: **Campus dataset**: Our results on 3D pose estimation projected in all views for the Campus dataset [24]. On the result of Camera 3 on the right column, the projected poses of Actor 1 and 3 overlap in the image plane.

Next, we review the related work on 3D human pose estimation. We mainly focus on single-human method because the related work on multiple human pose estimation from multiple views is limited. Afterwards, it follows the presentation of the 3DPS model.

5.2 Related Work

There is a plethora of literature on human pose estimation [122, 160]. In the previous chapters, we have focused on the related work of 2D pose estimation. Now, we focus on single and multiple human 3D pose estimation work.

The categorization in discriminative and generative approaches is also

common for 3D human body pose estimation approaches as well. In the discriminative category, a mapping between image or depth features and 3D human body poses is learnt [2, 76, 81, 157, 162, 172, 188]. These types of methods are unstable to corrupted data because of classification failures. They also only generalize up to the level in which unknown poses start to appear. Nonetheless, we have earlier shown (see Chapter 3) that this is not always the cases. Random forests trained on depth data have been proven to generalise well to unknown poses [105, 157]. However, current depth sensors are not useful for providing reliable depth information outdoors, where single and multiple cameras are still widely accessible. Deep learning can be a powerful tool for building discriminative models (see Chapter 4). It has demonstrated prominent results for 2D human pose estimation [42, 173, 174]. However, deep learning has been only recently introduced to the 3D human pose estimation using a single camera [111]. In our cases, we rely on a multi-view setup and training a discriminative model under this settings is not straightforward.

Most of the generative approaches rely on a kinematic chain where the parts of the object are rigidly connected. The problem is often coupled with tracking. In such approaches, the human skeleton is represented either in a high-dimensional state space [37, 49, 67, 121, 143, 158, 172], embedded in low dimensional manifolds bound to the learnt types of motion [188] or building hierarchically the body skeleton [109, 193]. Since these methods rely on tracking, they require initialisation and cannot recover in case of tracking failures.

There is another family of generative approaches, also called bottom-up or part-based, in which the human body is assembled from parts [4, 7, 9, 161]. These methods are referred to as pictorial structures and they do not imply rigid connections between the parts. Pictorial structures is a generic framework for object detection which has been extensively explored for 2D human body pose estimation [8, 9, 53, 59, 63, 186]. Deriving the 3D human pose from a single-view is possible by learning a mapping between poses in the 2D and 3D space [162] or lifting 2D poses [9], but this is not generic enough and is restricted to particular types of motion. Based on a multi-view setup, several recent approaches have been introduced that extend the pictorial structures to 3D human body pose estimation. The main challenge in extending pictorial structures to the 3D space is the large state space that has to be explored. Burenius et al. [40] have introduced an extension of pictorial structures to the 3D space and analysed the feasibility of exploring such a huge state space of possible body part translations and rotations. In order to make the problem computationally tractable, they impose a simple body prior that limits the limb length and assumes a uniform rotation. Adding a richer body model would make the inference much more costly due to the computation of the pairwise potentials. Consequently, the method is bound to single human pose estimation and an extension to multiple humans is not obvious. The follow-up work of Kazemi et al. [95] has introduced better 2D part detectors based on learning with randomized forest classifiers, but still relied on the same optimization as in [40]. In both works, the optimization is performed several times due to the inability of the detector to distinguish left from right and front from

back. As a result, the inference should be performed multiple times while changing identities between all the combinations of the symmetric parts. In case of multiple humans, either having separate state spaces for each person or exploring one common state-space, the ambiguity of mixing symmetric body parts among multiple humans becomes intractable. Both works [40, 95] have evaluated on a football dataset that they have introduced and it includes cropped players with simple background. We have evaluated our approach on this dataset as well. Another approach for inferring the 3D human body pose of a single person from multiple views has been proposed by Amin et al. [5]. Their main contribution lies in the introduction of pairwise correspondence and appearance terms defined between pairs of images. This leads to improved 2D human body pose estimation and the 3D pose is obtained by triangulation. Though this method obtained impressive results on HumanEva-I [159], the main drawback of the method is the dependency on the camera setup in order to learn pairwise appearance terms. Moreover, the inference is performed in the 2D space for each view separately. In contrast, we propose a 3D model in which the inference is performed directly in the 3D space.

The loose-limbed model of Sigal et al. [161] is similar to our model. It represents the human as a probabilistic graphical model of body parts. The likelihood term of the model relies on silhouettes (i.e. background subtraction) and applies only to single human pose estimation. This model is tailored to work with the Particle Message Passing method [166] in a continuous state space that makes it specific and computationally expensive. In contrast, we propose a 3DPS model which is generic and works well both on single and multiple human pose estimation. We resolve ambiguities imposed by multiple human body parts. Additionally, we operate on a reduced state space that makes our method fast.

5.3 Method

In this section, we first present the 3D pictorial structures (3DPS) model as a conditional random field (CRF). One important feature of the model is that it handles multiple humans whose body parts lie in the same 3D space. First, we present how we reduce the 3D space to a smaller discrete state space. Next, we describe the potential functions and the parameters of the 3DPS model, emphasising on how this model addresses challenges of single and multiple human 3D pose estimation in multi-views. Finally, we discuss the inference method that we employ to extract 3D body poses.

In our earlier work [16], we have introduced a 3DPS model for estimating the human body pose inspired by the original work on pictorial structures to define the body part as a limb and model its position and orientation. In our revisited 3DPS model, we reduce the parameterization of the body part to include only the 3D position. A body part can be interpreted as a physical body joint, other than the head body part. To model relation between body limbs in terms of translation and rotation, we define accordingly factors of pairs or triplets of parts in our factor graph (Figure 5.3). Consequently, the orientation is implicitly encoded in the factor graph. This human body parameterization

facilitates the inference task, and besides, it has demonstrated state-of-the-art results in 2D human pose estimation [186]. Finally, similar to other pictorial structures methods [8, 9, 40, 53, 5, 161], we have equally weighted the potential functions of the model in our earlier work on 3D pictorial structures [16]. In this work, we learn the parameters of our model using a Structured SVM (SSVM) solver [176] in order to balance the influence of the potential functions.

5.3.1 3D Pictorial Structures Model

The *3D pictorial structure (3DPS)* model represents the human body as an undirected graphical model. In particular, we model the human body as a CRF of n random variables in which each variable Y_i corresponds to a body part. An edge between two variables denotes conditional dependence of the body parts and can be described as a physical constraint. For instance, the lower limb of the arm is physically constrained to the upper one. To model the body constraints, a set of potential functions has been defined based on pairs or triplets of body parts (Figure 5.3). The body pose in 3D space is given by the configuration $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)$, where the state of each variable $Y_i \in \Lambda_i$ represents the 3D position of the body part and is taken from the discrete state space $\Lambda_i \subset \mathbf{R}^3$. The state space Λ_i is constructed based on 2D body part detection across multiple views.

Considering an instance of the observation $\mathbf{x} \in \mathbf{X}$ that corresponds to multiple-image evidence, a parameter vector $\mathbf{w} \in \mathbf{R}^D$, a set of reference poses $\mathbf{p}$ and a body configuration $\mathbf{y} \in \mathbf{Y}$, we define the posterior as:

$$
\begin{aligned}
p(\mathbf{y} \mid \mathbf{x}, \mathbf{w}, \mathbf{p}) = \frac{1}{Z(\mathbf{x}, \mathbf{w}, \mathbf{p})} \prod_{i}^{n} (\phi_i^{conf}(y_i, \mathbf{x}) \cdot \phi_i^{repr}(y_i, \mathbf{x}) \cdot \\
\phi_i^{vis}(y_i, \mathbf{x}) \cdot \phi_i^{temp}(y_i, p_i))^{w_i} \prod_{(i,j) \in E_{tran}} \psi_{i,j}^{tran}(y_i, y_j)^{w_{ij}} \\
\prod_{(i,j,k) \in E_{rot}} \psi_{i,j,k}^{rot}(y_i, y_j, y_k)^{w_{ijk}} \prod_{(i,j) \in E_{col}} \psi_{i,j}^{col}(y_i, y_j)^{w_{ij}}
\end{aligned} \tag{5.1}
$$

where $Z(\mathbf{x})$ is the partition function, E_{tran} and E_{rot} are the graph edges that model the kinematic constraints between the body parts and E_{col} are the edges that model the collision constraints between symmetric parts. The reference body poses $\mathbf{p}$ correspond to inferred poses from previous time steps.

The unary potentials are composed of the detection confidence $\phi_i^{conf}(y_i, \mathbf{x})$, reprojection error $\phi_i^{repr}(y_i, \mathbf{x})$, multi-view part visibility $\phi_i^{vis}(y_i, \mathbf{x})$ and the temporal consistence potential functions $\phi_i^{temp}(y_i, p_i)$. The pairwise and ternary potential functions encode the body prior model by imposing kinematic constraints on the translation $\psi_{i,j}^{tran}(y_i, y_j)$ and rotation $\psi_{i,j,k}^{rot}(y_i, y_j, y_k)$ between the body parts. Symmetric body parts are constrained not to collide with each other by the collision potential function $\psi_{i,j}^{col}(y_i, y_j)$. The parameters w_i, w_{ij} and w_{ijk} of the model correspond to weights for the unary, pairwise and ternary potential functions. In total, our model has 14 unary, 19 pairwise and 6 ternary potential fucntions ($D = 39$).

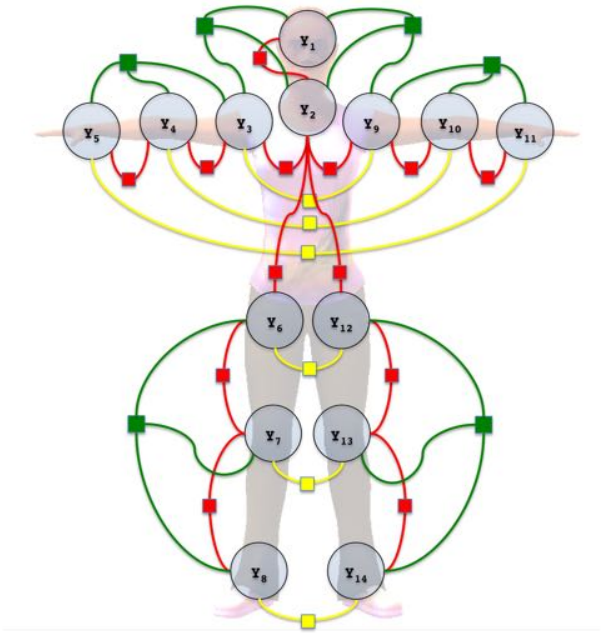


Figure 5.3: **Factor graph for the human body**: We use 14 variables in our graphical model to represent the body parts. A body part in our model corresponds to a physical body joint, other than the head part. The factors denote different types of constraints and are illustrated with different colours. The kinematic constrains are presented with red (translation) and green (rotation) edges (factors). The collision constraints are represented with yellow edges. The unary factors have not been drawn for simplicity reasons.

Next, we first define the state space, unary, pairwise and ternary potential functions and secondly describe how we learn the parameters of our model. Finally, we conclude with the inference of single or multiple individuals.

State space The state space Λ_i of a variable Y_i comprises the h hypotheses that each variable can take. A hypothesis corresponds to a 3D body part position in the global coordinate system. In order to be computationally efficient, we discretise the 3D space using body part detectors in each view separately. The 2D part detectors produce a posterior probability distribution of body parts in the 2D space. By sampling a number of samples from this distribution, we create 2D body part hypotheses in every view.

Assuming a calibrated system of c cameras, the 3D discrete state space is formed by triangulation of corresponding 2D body parts detected in multi-views. The triangulation step is performed for all combinations of view pairs. For each body part state space Λ_i, there is a number of hypotheses that can be associated to it. Not knowing the identity of humans creates wrong hypotheses stemming from the triangulation of the corresponding body parts of different individuals. Note that such wrong hypotheses can look correct in the 3D space and even create a completely fake body skeleton when different people are in a similar pose, as shown in Figure 5.4. Furthermore, the 2D part detectors

produce many false positive detections which result in the creation of wrong hypotheses. The total number of 3D hypotheses is given by:

$$h = n * {n_{samples}}^2 * c * (c-1)/2 \tag{5.2}$$

where $n_{samples}$ are the number of samples of the part detector. Generally, the number of hypotheses scales with the number of camera views c, and the number of sampled 2D body parts $n_{samples}$, but in general remains small enough for fast inference (Figure 5.5).

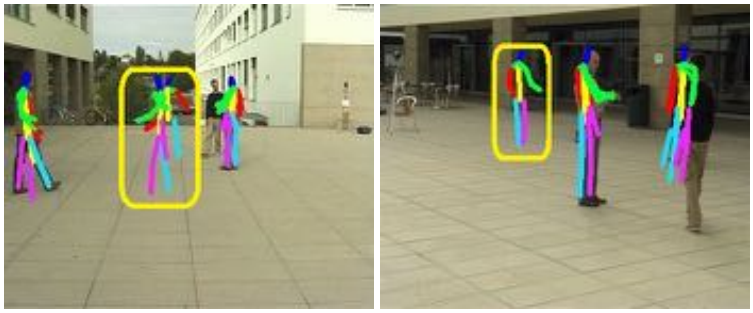


Figure 5.4: **State space**: The body part hypotheses are projected in two views. Fake hypotheses which form reasonable human bodies are observed in the middle of the scene (yellow bounding box). These are created by intersecting the body parts of different humans with similar poses because the identity of each person is not available during the formation of the state space.

Unary potentials In our approach, the unary potential functions are designed to score in a multi-view setup with multiple humans. A set of features for each 3D body part hypothesis contributes to the estimation of the unary functions. Firstly, every hypothesis has a confidence which is defined as the average of the triangulated 2D part detections' confidence. The average confidence forms the detection confidence function $\phi_i^{conf}(y_i, \mathbf{x})$. Secondly, given the triangulated hypothesis $y_i \in \mathbb{R}^3$ of the body part i detected in two camera views A and B at the locations $\mathbf{p_A} \in \mathbf{R}^2$ and $\mathbf{p_B} \in \mathbf{R}^2$, the reprojection error [79] is measured from the following geometric error cost function:

$$C(y_i; \mathbf{x}) = d(\mathbf{p_A}, \pi(y_i, \mathbf{x}, A))^2 + d(\mathbf{p_B}, \pi(y_i, \mathbf{x}, B))^2 \tag{5.3}$$

where d corresponds to the Euclidean distance, and $\pi(y_i, \mathbf{x}, A)$ and $\pi(y_i, \mathbf{x}, B)$ are the projections of the part y_i in view A and B. In order to express the reprojection error as a score, a sigmoid function is employed and integrated into the reprojection error potential function $\phi_i^{repr}(y_i, \mathbf{x})$. The final potential function becomes:

$$\phi_i^{repr}(y_i, \mathbf{x}) = \frac{1}{1 + \exp(C(y_i; \mathbf{x}))}. \tag{5.4}$$

To take advantage of the multi-view information, we introduce the body part multi-view visibility potential $\phi_i^{vis}(y_i, \mathbf{x})$ which weights a hypothesis based on the number of views in which it has been observed. To compute the

number of views, we project the hypothesis to each view and search in a small radius ($\sim$ 5 pixels) for an instance of the detector. Then, we normalize the estimated number of visible views with respect to the total number of cameras. Consequently, hypotheses that occur from ambiguous views (e.g. opposite cameras) or false positive hypotheses (Figure 5.4) are implicitly penalized by obtaining a smaller visibility weight. Thus, the visibility term is complementary to the reprojection error.

The above unary potential functions are computed based on the observation from the current time step. To impose temporal consistency with previous inferred body poses $\mathbf{p}$ and the current 3D hypotheses, we introduce the temporal consistence function $\phi_i^{temp}(y_i, p_i)$, which acts as a regulariser between the inferred and candidate 3D part hypotheses. However, wrongly inferred body parts can occur as well. To account for both situations, we propose to consider the geometric distance between the 3D hypothesis of the part i and the inferred part p_i, if it is below a threshold c, which we have set experimentally to 10 cm. The role of the threshold c is to define a perimeter in which correct hypotheses can lie and thus it not a hard constraint. Finally, the geometric distance is reformulated as a score using a sigmoid function:

$$\phi_i^{temp}(y_i, p_i) = \begin{cases} \frac{1}{1+\exp(d(y_i,p_i))} \text{ if } d(y_i, p_i) < c \\ \epsilon \text{ otherwise} \end{cases} \tag{5.5}$$

where $d(y_i, p_i)$ is the Euclidean distance between the 3D part hypothesis and previously inferred parts and ϵ a small constant for numerical stability during the inference.

The main benefit of the unary potential functions' formulation is to make use of the multi-view information. The confidence of the part detector, which also contributes to the creation of the 3D hypotheses, is the most important potential function. However, false positive detections or triangulations with geometric ambiguity should be penalized. This is achieved by the reprojection and multi-view visibility potential functions. For instance, a wrongly detected 2D part, with a high detection confidence, should normally have a high reprojection error. Hence, the score of the reprojection potential of a false positive is low. Furthermore, 3D part hypotheses that have been created from different individuals with similar poses can have small reprojection error but they are penalized from the multi-view visibility potential. Finally, true positive part detections of different individuals create wrong body part hypotheses with high detection confidence, but they are penalized by the body prior potential functions.

Pairwise and ternary potentials The paradigm of pictorial structures in the 2D space has successfully modelled the relations between body parts in terms of location and orientation [9, 59, 63]. Recently, the body parts have been defined only using the location parameters and the orientation has been encoded in the body prior [5] or in a mixture of parts [186]. In the revisited 3DPS model, we follow the same idea of body part parametrisation and model the constraints between physical body limbs in the factor graph (Figure 5.3). In particular, we define two type of constraints: kinematic and collision. The kinematic

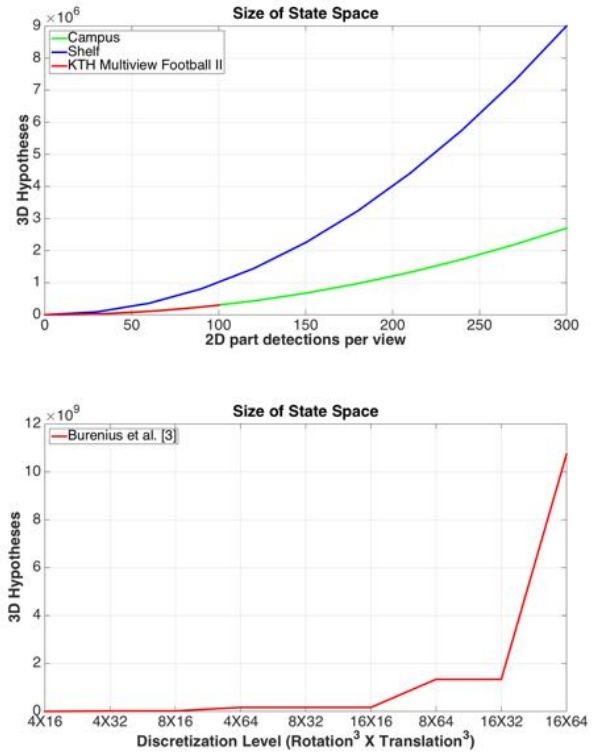


Figure 5.5: **Size of the state space**: On the top graph, the size of the state space for three different datasets is presented, based on the number of sampled 2D part detections per view and individual. On the bottom graph, the size of the state space is presented according to the 3D space discretisation of [40]. It is clear that using a part detector as input results in magnitudes smaller state space in comparison to 3D space discretisation in terms of rotation and translation. In both cases, 10 body parts have been considered for the computation of the final number of 3D hypotheses. In [40], a discretisation of 8^3X32^3 ($Rotation^3 \times Translation^3$) has been chosen as a compromise between performance and speed. In our case, we have sampled 40 2D parts for all the experiments.

constraints are modelled using translation and rotation transformations, while the collision constraints prevent the symmetric body parts from colliding with each other due to false positive detections.

Starting with the kinematic constraints, the translation potential models the translation of the part i to the local coordinate system of the part j. A multivariate Gaussian is used to capture this transformation and it is given by:

$$\psi_{i,j}^{tran}(y_i, y_j) = \mathcal{N}(y_{ij}^T \mid \mu_{ij}^T, \Sigma_{ij}^T), \tag{5.6}$$

where $y_{ij}^T = y_i - y_j$, μ_{ij}^T is the mean and Σ_{ij}^T is the covariance. The main diagonal of the covariance is only used for relaxing the computations during the inference. Assuming that each body part belongs to a body limb, the translation brings one limb to the local coordinate system of the other. For the rotation, we consider the case of a hinge joint (i.e. 1DoF) between two body

limbs. To that end, a triad of body parts is used to form two body limbs with a shared joint. The rotation across one axis can be easily computed from the dot product of the two body part vectors. The rotation potential function is modelled using a one dimensional Gaussian distribution:

$$\psi^{rot}_{i,j,k}(y_i, y_j, y_k) = \mathcal{N}(y^R_{ijk} \mid \mu^R_{ijk}, \sigma^R_{ijk}), \tag{5.7}$$

where $y^R_{i,j,k} = arccos(dot(y_i - y_j, y_k - y_j))$, μ^R_{ijk} is the mean and σ^R_{ijk} the variance. Moreover, we consider positive angles for the computation of the potential function. In order to model the whole rotational space, a von Mises distribution would be required. In our experiments, we have seen that an approximation with a Gaussian is sufficient.

In addition, we model the relation between the symmetric body parts to avoid collisions between them. This problem occurs because of false positive (FP) detections that more often occur for the symmetric body parts. We model the relation of the symmetric body parts by learning their Euclidean distance using another one dimensional Gaussian distribution which expressed from the collision potential function:

$$\psi^{col}_{i,j}(y_i, y_j) = \mathcal{N}(d(y^{col}_{ij}) \mid \mu_{ij}, \sigma^{col}_{ij}), \tag{5.8}$$

where $d(y^{col}_{ij})$ corresponds to the Euclidean distance between the part i and j, μ^{col}_{ij} is the mean and σ^{col}_{ij} the variance.

We use ground-truth data to learn the pairwise and ternary potential functions as well as the number of the previous time steps for the temporal consistence potential function. Since the world coordinate system is cancelled by modelling the relation of the body parts in terms of local coordinate systems, we are less dependent on the camera setup, in contrast to [5]. Moreover, our prior model is stronger than a binary voting for a body part configuration [40] and less computational expensive than [161]. During inference of multiple humans, our prior model constrains the body parts of each individual to stay connected.

5.3.2 Margin-based Parameters Learning

The 3DPS model has several potential functions of different nature and magnitude. Moreover, some potential functions are more error prone than others. Consequently, arises the necessity to balance the influence of the potentials within the final model. The parameters $\mathbf{w}$ of the the model weight accordingly the unary, pairwise and ternary potential functions. To learn the parameters, we pose our problem as regularised risk minimisation and use a Structured SVM (S-SVM) solver.

Our goal is to learn a weight for each potential function, given a set of training samples S with labels $y^s = \{-1, 1\}$. For each training sample, a feature vector $\mathbf{\Phi}(\boldsymbol{\phi}^s, \boldsymbol{\psi}^s)$ with the concatenation of all potential functions is formed. Finally, the $0-1$ loss function has been chosen and integrated into the energy

function, which is given by:

$$\min_{\mathbf{w}} \frac{1}{2} \parallel \mathbf{w} \parallel^2 + \frac{C}{S} \sum_{s=1}^{S} \xi^s$$

$$s.t. \quad \max(0, 1 - y^s \langle \mathbf{w}, \mathbf{\Phi}(\boldsymbol{\phi}^s, \boldsymbol{\psi}^s) \rangle) \leqslant \xi^s. \quad (5.9)$$

where ξ^s are the slack variables and C a constant. Finally, the optimisation is done with the cutting plane algorithm [62]. Margin-based parameter estimation has been successfully applied in segmentation [119, 129] and more recently in 2D human pose estimation [42] with different type of loss functions. During the experiments, we have observed that the $0-1$ loss has fitted well to our problem. Moreover, during the evaluation we demonstrate that weighting the potential function has an important impact on the final result.

5.3.3 Inference of Multiple Humans

The final step for obtaining the 3D pose of multiple individuals is the inference. The body part hypotheses of all humans share the same state space. In addition, the state space includes completely wrong hypotheses due to the unknown identity of the individuals and false positive detections as well. However, our potential functions count for these problems and penalize each hypothesis accordingly, allowing us to parse each human correctly.

Here, we seek to estimate the posterior probability (5.1) using belief propagation. Since our graphical model does not have a tree structure, we employ the loopy belief propagation algorithm [30] to estimate the marginal distributions of the body part hypotheses. Sampling a solution directly from the marginals is not possible, since the hypotheses of different individuals lie together in the same state space. For that reason, we introduce a human localisation prior $\mathbf{h}$ for sampling from the marginals. We first localise each individual in each view and associate the localized bounding boxes across different views in order to recover the identity of each individual. To that end, we could use a human detector similar to [16], but we employ a multi-view human tracker [24] for more accurate bounding-box localisation and individual across-view association, as in [21]. Given the localization bounding boxes $\mathbf{h}$ of all individuals across all views, we look for the samples of each individual with the highest probability from the marginals:

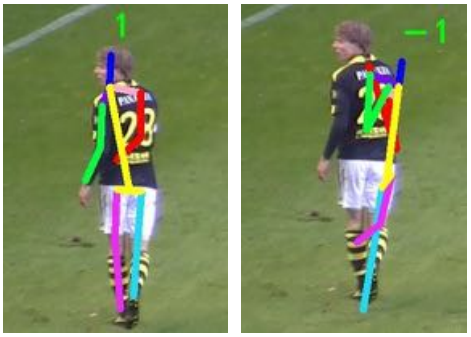
$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}, \mathbf{w}, \mathbf{p}, \mathbf{h}) \quad (5.10)$$

where $\hat{\mathbf{y}}$ is a subset of hypotheses that corresponds to the body parts of each individual. For each individual and for each body part, we look for the sample with the highest probability which at the same time is projected inside the localization bounding boxes across all or most views. Since the number of hypotheses in the state space is limited and the marginals are sorted, the above computation is inexpensive. Gradually, the 3D body poses of all individuals are parsed.

Despite the fact that the tracker provides each individual's identity, we do not make use of it for forming a local state space for each individual, in the

beginning. Instead, we prefer to build a global state space by triangulating all the possible combinations of body part detections between view pairs. The reason is that the localisation can result in bounding boxes **h** with body parts of different individuals in case of occlusion. In that case, the inference from a local state space would result on inferred poses with body parts from different individuals, which would still look realistic. However, this problem does not occur with the global state space, where our model resolves the ambiguities between different individuals with the geometric potential functions.

Our framework for multiple human 3D pose estimation applies exactly the same on single human pose estimation. In the next section, we demonstrate it by evaluating our model both on single and multiple human 3D pose estimation.



Positive Negative

Figure 5.6: **Training sample**: On the left column a positive training sample is presented, while on the right column a negative one. We choose negative samples which form reasonable human poses, instead of randomly sampling from the image space.

5.4 Results

In this section, we evaluate our approach on single and multiple human pose estimation on four datasets. Our model is composed of several potential functions which contribute differently to the final result. At first, we perform an evaluation of the potential functions and afterwards compare our method to related approaches. For single human pose estimation, we use the HumanEva-I [159] and KTH Multiview Football II [40] datasets, while we evaluate on the Campus [24] and Shelf [16] datasets for multiple human pose estimation.

The model that we employ for the experiments is composed of 14 body parts (Figure 5.3). For each evaluation dataset, we use the training data to learn our model's unary, pairwise and ternary terms as well as the model parameters. For learning the parameters of the model, we generate positive and negative examples according to the ground-truth of each dataset. On one hand, we consider as positive, samples that are very close to the ground-truth body pose in each view. On the other hand, the negative samples still form human body poses, but they do not correspond to the correct one, as it is

depicted by Figure 5.6. Our part detector is based on the 2D part detector of [5] for all datasets other than KTH Multiview Football II [40]. In the KTH Multiview Football II dataset, there is a big amount of 2D training data which allows us to train a deep part detector similar to [174, 111]. The part detector of [5] is generative model that is based on pictorial structures from which we sample from the posteriors. The deep part detector is based a discriminative model similar to Chapter 4 which is combined with a deep part classifier, as in [111], for obtaining a confidence for the regressed parts. The human localisation is done with the tracker of [24]. For all the experiments, we have set the number of loaded samples ($n_{samples}$) of the 2D part detector to 40, since we have experimentally observed that it is a good compromise between performance and speed. Finally, we employ the *loose* PCP (percentage of correctly estimated parts) performance measure for evaluating our results.

5.4.1 Potential Functions Contribution

The 3DPS models is composed of unary, pairwise and ternary potential functions. We perform an in-depth evaluation of the potential functions and analyse their contribution to the model. On single human pose estimation, we use the KTH Multiview Football II [40] dataset for analysing the behaviour of the potential functions. On multiple human pose estimation, we choose the Campus [24] and Shelf [16] datasets. Since, we observe different human motion across the individuals in both datasets, the evaluation is done for each individual separately.

Our model has in total seven potential functions: the confidence, reprojection, visibility, temporal consistence, translation, collision and the rotation. We start with a basic model that is composed of a single potential function. Gradually, we aggregate in the model all the potentials and report the performance of different body parts at each step. Finally, the full model is composed of all unary, pairwise and ternary terms. The results are summarised in Tables 5.1 and 5.2. Below, it follows an analysis for each term separately. In addition, the results for each dataset are summarised in Figure 5.7.

Confidence: This is the most important potential function of the 3DPS model. The part-detectors' confidence contributes to the state space generation and confidence potential as well. As a result, a weak part detector would have a big effect on the whole performance. However, this is not the case in our model. The two/multi-view potentials and strong body prior efficiently penalises 3D hypotheses, which occurred from false positive part detections. For instance, the weak performance of part detectors in the Campus dataset for Actor 2 and 3 (Table 5.1c and 5.1d) is surpassed with the use of the other potentials. On the other hand, the already good performance of the part detectors in the Shelf dataset (Table 5.2a, 5.2b and 5.2c) or KTH Multiview Football II (Table 5.1a) has the most dominant influence to the final result, which does not improve significant by adding the other potential functions.

Reprojection & visibility: The reprojection error potential function makes use of two-views for estimating a score, while the visibility makes use of all views. Consequently, these terms are affected by the accuracy of the

camera calibration. Moreover, geometric ambiguities due the camera poses (e.g. opposite cameras) can negatively affect the behaviour of these potentials (Figure 5.11). This is particularly the case for the reprojection potential in the Shelf dataset where the lower arms of Actor 1 (Table 5.2a) or the upper arms of Actor 2 (Table 5.2b) loose some performance due to geometric ambiguities. This type of ambiguities occur less often by better camera positioning (e.g. Campus dataset) or employing more views. Finally, the visibility potential, which relies on all views, always improves the final result.

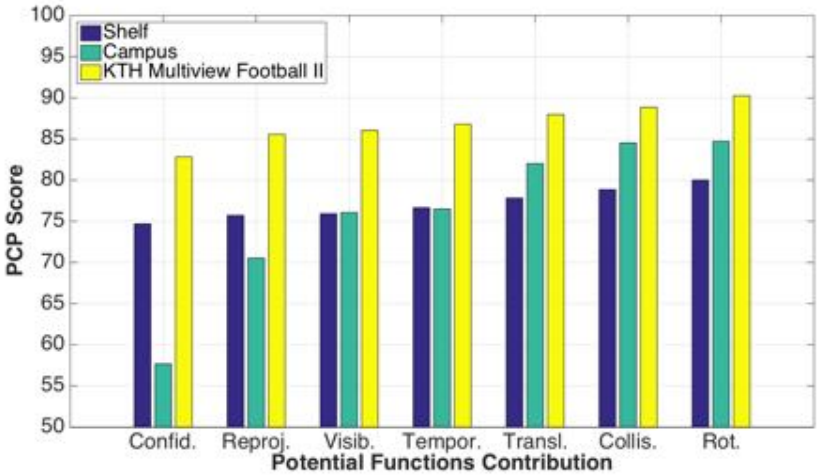


Figure 5.7: **Potentials' contribution**: The contribution of each potential function is presented for the KTH Multiview Football II [40], Campus [24], Shelf [16] datasets. The performance measurement is the PCP score. The horizontal axis corresponds to the aggregation of the potential functions (confidence, reprojection, visibility, trermporal consistency, translation, collision, rotation). For the Campus and Shelf datasets, the average PCP score of all individuals is presented. Adding more potential functions to the base model (only confidence) gives considerable improvement to the KTH Multiview Football II and Campus datasets, while the improvement is smaller in the Shelf dataset.

Temporal consistency: Sustaining the consistency within the inferred body poses has a positive impact to the performance of our model, for most of the cases. In particular, the temporal consistence term has bigger contribution to the model in cases of uniform motion. For instance, the performance is improved more in walk gait (Table 5.1c and 5.1d) than playing football (Table 5.1a). However, the performance of our model on some body parts (e.g. Actor 1 in Campus - Table 5.1b) decreases by adding the temporal consistence term. The reason for this result is the threshold c that we have set during training. This threshold defines if an inferred part will be considered as correct or not. In order to keep the number of the model parameters low, we have set a single c for all body parts and all evaluation datasets as well. The motion of the body parts is nevertheless different. Furthermore, the motion between different individuals varies as well. Therefore, setting a single threshold for all body parts is not optimal, but it guarantees a more generic model.

Translation & collision: The body prior is divided into two pairwise and a ternary potentials. The most influential prior term is the translation potential function. It improves the performance in all datasets. The second pairwise term, the collision potential, helps to identify 3D hypotheses which came out from the triangulation of false positive detections of symmetric body parts. In some cases, the collision potential has small contribution or cuts down the performance of the upper legs (i.e. Actor 3 of Shelf dataset - Table 5.2c) because of false positive detections, which still fit well to the human body model.

Rotation: This ternary term requires triplets of body parts in order to be computed. Thus, it is the most expensive potential, in terms of computations. However, the contribution of this potential to the final result is not proportional to its cost. For example, the performance is improved around 1% in the Shelf dataset (Table 5.2a, 5.2b and 5.2c), while the improvement is much less in Campus (Table 5.1b, 5.1c and 5.1d), where the pose variation is confined to walking. On the other hand, it appears to be more valuable in the case of large body pose variation such as in the KTH Multiview Football II dataset. However, there are some cases in which the rotation potential reduces the performance of some body parts due to incorrect hypotheses which fit well to the rotation prior model.

Overall performance: Through the above analysis of the potential functions, fruitful conclusions are drawn. In general, the confidence of the part detectors is very crucial to the final result, but a weak part detector can be significantly refined using our 3DPS model. Moreover, the reprojection error potential is more affected from the camera pose, while the visibility term compensates in cases of geometric ambiguity. Finally, the body prior mainly benefits from the translation and collision potentials, while the rotation potential contributes more in case of large body pose variation (Figure 5.7). In order to modulate less the model, we have used the same prior model, in terms of the translation, rotation or collision, for all body parts. However, the results highlight that some body parts do not benefit in all cases from this assumption. For example, using a single Gaussian distribution as rotation potential has reduced the head performance for half of the examined cases. Thus, a combination of individual prior models for different body parts could result in better performance, but create a less generic model.

Discussion: One fundamental assumption of the 3DPS model is a calibrated multi-view setup. As it is reflected from the results, calibration errors or geometric ambiguities influence the model performance. Thus, a further step would be to assume an uncalibrated setup, where the goal would be to infer both the 3D body pose and camera pose at the same time [55]. Furthermore, more robust part detectors would have a big impact on the model performance. At this stage, deep learning methods [42, 78] could contribute to more robust part detections for a single-view or combined views.

5.4.2 Single Human Pose Estimation

We evaluate our method on single human 3D pose estimation for demonstrating that it performs as well as start-of-the-art multi-view approaches [5, 40]. The

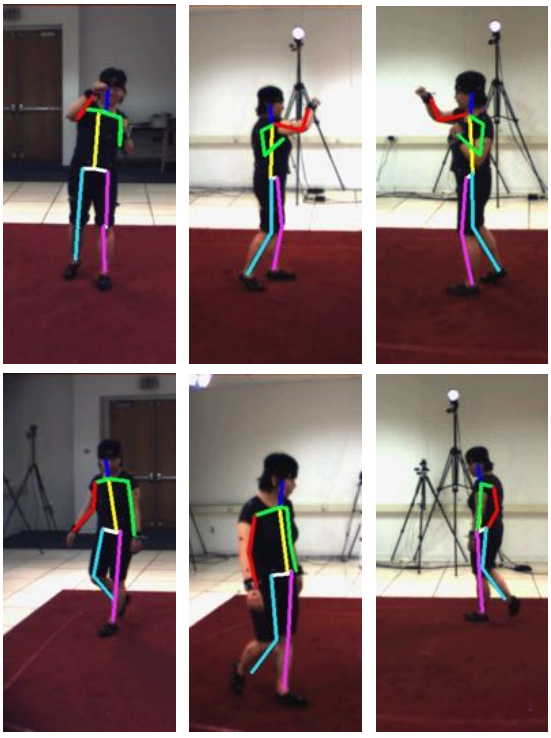


Figure 5.8: **HumanEva-I dataset**: The 3D estimated body pose is projected across each view for the Box and Walking sequences.

purpose of this experiment is to highlight that we can achieve similarly good or even better results than other methods with an enriched, in terms of potentials, human model.

HumanEva-I: We evaluate on Box and Walking sequences (Figure 5.8) of the HumanEva-I [159] dataset and compare with [5] and [161]. We share similar appearance term only for the 2D single view part detection with [5] and employ different body models. Table 5.3 summarizes the results of the average 3D joint error in millimetres. In this dataset, we have used the aforementioned evaluation measure in order to keep up with the related work. Notably, Amin et al. [5] report very low average error, and we also achieve similar results. Cases in which we have observed failures are related to lack of correct detected parts from at least two cameras.

KTH Multiview Football II: In this dataset, we evaluate on Player 2 as in the original work of Burenius et al. [40] and the follow up work of [95]. We follow the same evaluation protocol and estimate the PCP (percentage of correctly estimated parts) scores for each set of cameras (Figure 5.9). The results are summarized in Table 5.4. We outperform the method of [40] on both cameras setups, using a richer body model and a radically smaller state space (Figure 5.5). In [40], the 3D space is discretised in terms of rotation and translation at different discretisation levels. However, a fine discretisation is

Camera 1 Camera 2 Camera 3

Figure 5.9: **KTH Multiview Football II dataset**: The 3D estimated body pose is projected across each view. The results comes from the inference with all cameras.

required for accurate results. On the other hand, our discrete state space is significantly smaller without the cost of a reduced performance. Finally, the more accurate part detectors of [95] improve the results, but we still obtain superior performance.

In addition, learning the parameters of the model brings a considerable improvement in comparison to our previous work [16]. Our approach runs on around 1 fps for single human 3D pose estimation, given the 2D detections. All the experiments are carried out on a standard Intel i7 2.40 GHz laptop machine and our method is implemented in C++ with loop parallelizations.

5.4.3 Multiple Human Pose Estimation

The problem of multiple human 3D pose estimation has not been extensively addressed yet. Most of the related work has focused on single human 3D pose estimation [5, 40, 161]. Moreover, the number of available datasets in the literature for multiple human pose estimation from multiple views is very limited. Recently, we have proposed two multiple human datasets [16]: the Shelf and Campus. We evaluate our method on these datasets using the PCP (percentage of correctly estimated parts) and compare with related work as well. Since we are not aware of another method which performs multiple human 3D pose estimation, we choose single human approaches [5, 6] to compare to and perform 3D pose estimation for each human separately. Of course, this way of evaluation is not to our favour because evaluating on each human separately, knowing their identity, excludes body part hypotheses that belong to other humans and simplifies the inference.

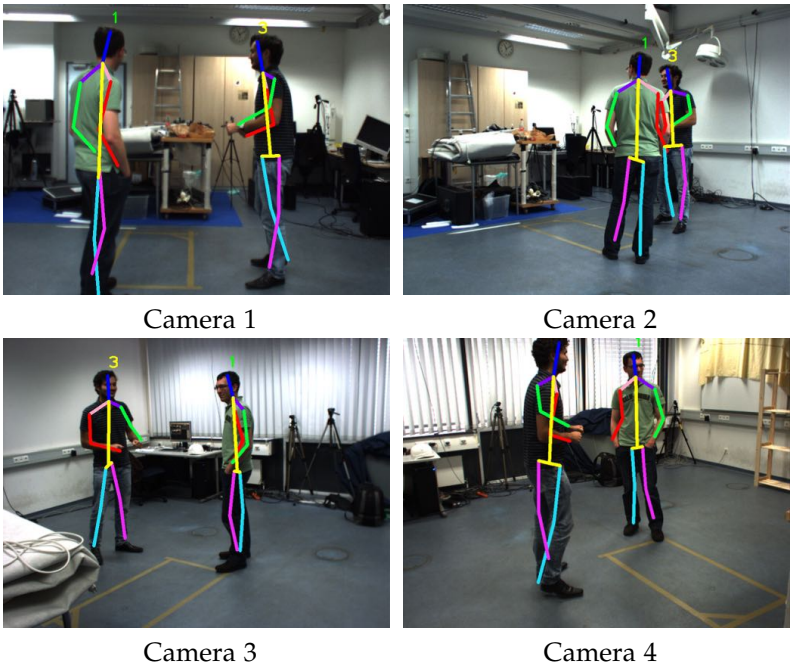


Figure 5.10: **Shelf dataset**: Our results projected in 4 out of 5 views of the Shelf dataset [16].

Campus: In this dataset, three different individuals (Figure B.3) share a common state space. In [16], we have demonstrated that putting the 3D hypotheses of all individuals to the same state space does not result in a reduced performance using the 3DPS model. In addition, in this work we show that learning the parameters of the model improves further the final result (Table 5.5a). The performance of the 3DPS model on Actor 1 distinguishes itself from the other two for the accurate body pose estimation in most of the evaluated frames. Actor 2 follows with similar results, while Actor 3 looses some performance due to the weak localisation of the torso and lower arms. The reason for the reduced performance can be found in the analysis of the potential functions in Table 5.1c. It is observed that the part detectors for the torso and lower arms are weak for the Actor 2. In comparison to [5] where the inference is done separately for each Actor, we have in general better limb localisation and we perform similar for the rest of the body parts.

Shelf[1]**:** Similarly to the Campus dataset, three individuals compose the Shelf dataset (Figure 5.1 and 5.10). The head and torso are localised correctly for all individuals for most of the time, while the lower arms and upper legs are the most difficult parts to localise. Going back to the analysis of the potential function in the Tables 5.2a, 5.2b and 5.2c, one observes weak behaviour of the part detectors for these body parts. This is a common fact for all three Actors. Comparing to [6], we perform mainly better on the arms and lower legs.

[1]The dataset and additional material is available at: `http://campar.in.tum.de/Chair/MultiHumanPose`.

Furthermore, the inference is done separately for each individual in [6], while we assume a common state space for all individuals. Finally, we demonstrate in this dataset as well that the parameter learning using a Structured SVM (S-SVM) considerably improves the final result in comparison for our earlier work on 3D pictorial structures [16]. Further results are presented in Appendix B and also the human localization results in Appendix C.

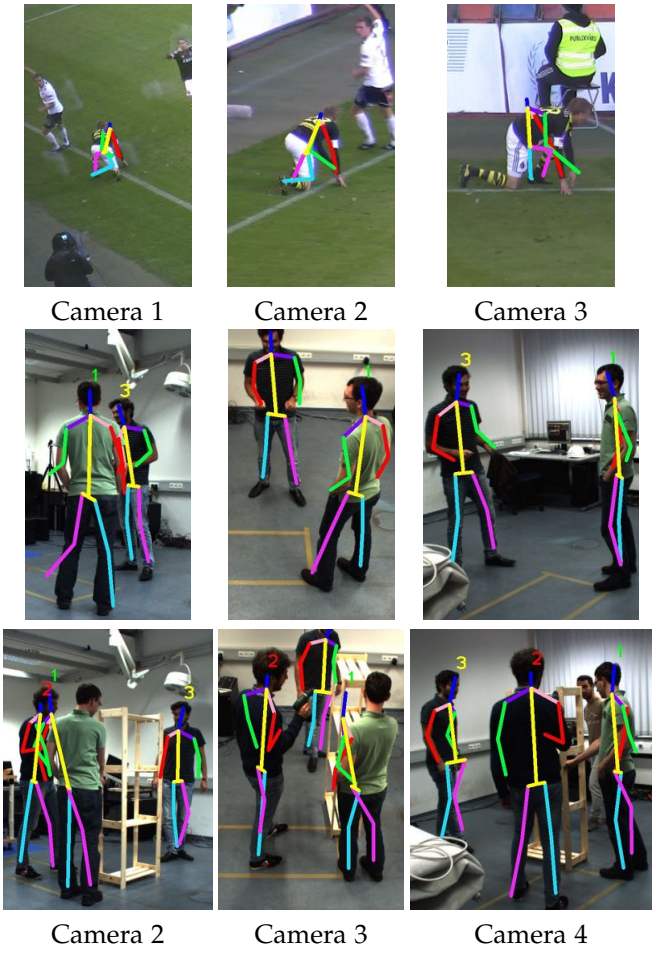


Figure 5.11: **Failure cases**: On the top row, it is presented a wrongly inferred body pose due to geometric ambiguities and false part localisation (KTH Multiview Football II dataset). On the middle row, the lower limb of Actor 1 looks correct from Camera 3 and 4, but it is actually wrongly localised again due to geometric ambiguity (Shelf dataset). On the bottom, the body pose of Actor 1 is wrong due to 3D hypotheses which occurred from false positive part detections.

5.5 Conclusions

We have presented a 3D pictorial structures (3DPS) model for recovering the 3D human body pose of multiple individuals from multiple camera views. Our generative model relies on 2D part detectors for forming a discrete state space which allows fast inference. The 3DPS model is composed of a set of potential functions, which make use of two- and multi-view observations. To correctly weight the influence of the potential functions, we have used a Structured SVM (S-SVM) to learn the parameters of our model. This model has been applied to multiple human pose estimation without knowing the identity of the individuals in advance. Self and natural occlusions are handled by our algorithm by only relying on noisy part detectors for each camera view. Finally, the model is applicable to single human pose estimation, where we have demonstrated competitive results to the related work.

As future work, we would like to focus on the inference part of the method. The multiple human inference is conditioned on the localization of the individuals using detection or tracking. In the examined scenarios, localizing humans has been a relative easy task. Although many robust human localization methods have been proposed, we believe that in very crowded environments the localization could be a drawback for our method. Under this limitation, we would not correctly parse all the individuals during inference. For that reason, we would like to explore the direction where perform inference without the dependence on the localized individuals.

After introducing a model that copes with a multi-view setup surrounded by multiple individuals, we choose a more challenging scenario as an application. In the next chapter, we introduce a unique dataset for human pose estimation in the operating room (OR). Then, we apply our model for estimating the body pose of multiple humans in this very challenging environment.

Table 5.1: **Potentials's aggregation**: The aggregated PCP (percentage of correctly estimated parts) scores are presented for the potential functions. Each column corresponds to an additional potential function.

(a) **KTH Multiview Football II**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Upper Arms	89.49	92.06	92.29	93.69	96.26	96.83	97.96
Lower Arms	56.78	63.55	64.02	65.89	66.82	68.93	71.86
Upper Legs	96.50	97.43	97.66	97.90	98.83	99.30	99.40
Lower Legs	88.55	89.25	90.19	89.72	90.02	90.32	91.80
Average	82.83	85.57	86.04	86.80	87.98	88.85	90.26

(b) **Campus - Actor 1**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	68.97	96.55	96.55	100.00	100.00	100.00	96.55
Torso	75.86	82.76	89.66	93.10	89.66	89.66	93.10
Upper Arms	63.79	86.21	93.10	75.86	96.55	98.28	96.55
Lower Arms	53.45	72.41	75.86	65.52	86.21	91.38	86.21
Upper Legs	79.31	87.93	96.55	89.66	96.55	93.10	93.10
Lower Legs	82.76	91.38	89.66	87.93	89.66	89.66	96.55
Average	70.69	86.21	90.23	85.35	93.11	93.68	93.68
All parts	70.35	85.52	89.66	83.10	92.76	93.45	93.45

(c) **Campus - Actor 2**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	74.12	87.65	94.71	91.18	100.00	100.00	98.24
Torso	41.18	46.47	47.06	45.88	46.47	47.65	48.82
Upper Arms	66.76	76.47	81.47	89.41	89.12	93.82	97.35
Lower Arms	13.24	17.94	20.88	33.82	32.35	41.47	42.94
Upper Legs	60.00	65.29	69.41	68.53	75.59	75.59	75.00
Lower Legs	86.76	87.65	87.65	90.00	90.29	90.59	89.41
Average	57.01	63.58	66.86	69.80	72.30	74.85	75.29
All parts	56.88	62.88	66.06	70.06	72.12	75.06	75.65

(d) **Campus - Actor 3**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	30.69	67.65	81.55	80.58	89.32	94.17	93.20
Torso	30.69	56.86	69.90	73.79	79.61	83.50	85.44
Upper Arms	51.49	66.67	75.73	79.13	79.61	88.83	89.81
Lower Arms	41.09	50.49	59.71	66.99	68.45	77.18	74.76
Upper Legs	60.40	66.67	75.24	80.10	85.92	84.95	91.75
Lower Legs	57.43	62.75	64.56	65.53	81.07	81.55	76.21
Average	45.30	61.85	71.12	74.35	80.66	85.03	85.20
All parts	48.22	61.77	70.19	73.79	79.90	84.27	84.37

Table 5.2: **Potentials's aggregation**: The aggregated PCP (percentage of correctly estimated parts) scores are presented for the potential functions. Each column corresponds to an additional potential function.

(a) **Shelf - Actor 1**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	92.39	94,02	94.02	94.57	95.65	96.17	96.29
Torso	100.00	100,00	100.00	100.00	100.00	100.00	100.00
Upper Arms	79.08	80,71	80.43	80.98	80.16	82.24	82.24
Lower Arms	57.34	56,79	58.97	62.23	62.50	65.30	66.67
Upper Legs	40.49	40,76	40.49	41.58	44.29	42.90	43.17
Lower Legs	80.43	81,25	81.25	82.34	85.60	85.79	86.07
Average	74.96	75,59	75.86	76.95	78.03	78.73	79.07
All parts	70.71	71,30	71.63	72.88	74.08	74.86	75.26

(b) **Shelf - Actor 2**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	68.42	57.89	57.89	57.89	68.42	68.42	78.95
Torso	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Upper Arms	84.21	81.58	81.58	84.21	84.21	84.21	82.58
Lower Arms	31.58	36.84	36.84	36.84	34.21	42.11	47.37
Upper Legs	47.37	47.37	47.37	47.37	47.37	50.00	50.00
Lower Legs	73.68	76.32	76.32	76.32	78.95	78.95	78.95
Average	67.54	66.67	66.67	67.11	68.86	70.62	72.98
All parts	64.21	64.21	64.21	64.74	65.79	67.90	69.68

(c) **Shelf - Actor 3**

	Unary				Pairwise		Ternary
	Confidence	Reprojection	Visibility	Temporal	Translation	Collision	Rotation (Full model)
Head	74.00	92.00	94.00	94.00	95.00	96.00	98.00
Torso	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Upper Arms	90.00	90.00	90.00	91.00	91.50	92.00	93.15
Lower Arms	85.00	86.50	86.00	89.50	88.00	90.00	92.30
Upper Legs	49.50	50.00	50.00	50.00	50.00	52.30	56.50
Lower Legs	91.00	91.00	91.50	91.00	95.00	96.00	97.00
Average	81.58	84.92	85.25	85.92	86.58	87.72	89.49
All parts	80.50	82.70	82.90	83.70	84.40	85.66	87.59

Table 5.3: **Human-Eva I**: The average 3D joint error in millimetres (mm) is presented.

Sequence	Walking	Box
Amin et al. [5]	54.5	47.7
Sigal et al. [161]	89.7	-
Proposed	68.3	62.7

Table 5.4: **KTH Multiview Football II**: The PCP (percentage of correctly estimated parts) scores using 2 and 3 cameras are presented. One can observe that we have mainly better results for the upper limbs. In addition, learning the parameters of the CRF helps to improve the final result in comparison to [16].

	2 Cameras			3 Cameras			
	Bur. [40]	Bel. [16]	Proposed	Bur. [40]	Bel. [16]	Kaz. [95]	Proposed
Upper Arms	53	64	96	60	68	89	98
Lower Arms	28	50	68	35	56	68	72
Upper Legs	88	75	98	100	78	100	99
Lower Legs	82	66	88	90	70	99	92
Average	62.7	63.8	87.5	71.2	68.0	89.0	90.3

Table 5.5: **State-of-the-art comparison**: The PCP (percentage of correctly estimated parts) scores are presented for different related work and the proposed method. The global score of all individuals takes additionally into consideration the number of occurrence for each individual.

(a) **Campus dataset**

	Amin et al. [5]			Belagiannis et al. [16]			Proposed		
	Actor 1	Actor 2	Actor 3	Actor 1	Actor 2	Actor 3	Actor 1	Actor 2	Actor 3
Head	64.58	78.84	38.52	93.62	97.40	81.26	96.55	98.24	93.20
Torso	100.00	100.00	100.00	49.94	41.13	69.67	93.10	48.82	85.44
Upper Arms	94.80	84.66	83.71	82.85	90.36	77.58	96.55	97.35	89.81
Lower Arms	66.67	27.25	55.19	77.80	39.65	61.84	86.21	42.94	74.76
Upper Legs	100.00	98.15	90.00	86.23	73.87	83.44	93.10	75.00	91.75
Lower Legs	81.25	83.33	70.37	91.39	89.02	70.27	96.55	89.41	76.21
All body parts	85.00	76.56	73.70	82.01	72.43	73.72	93.45	75.65	84.37
Average (Actors)		78.42			75.79			84.49	
All individuals (global PCP)		76.61			73.82			81.08	

(b) **Shelf dataset**

	Amin et al. [6]			Belagiannis et al. [16]			Proposed		
	Actor 1	Actor 2	Actor 3	Actor 1	Actor 2	Actor 3	Actor 1	Actor 2	Actor 3
Head	93.75	100.00	85.23	89.30	72.10	94.66	96.29	78.95	98.00
Torso	100.00	100.00	100.00	90.20	92.80	96.35	100.00	100.00	100.00
Upper Arms	73.08	73.53	86.62	72.16	80.11	91.00	82.24	82.58	93.15
Lower Arms	32.99	2.94	60.31	60.59	44.20	89.00	66.67	47.37	92.30
Upper Legs	85.58	97.06	97.89	37.12	46.30	45.80	43.17	50.00	56.50
Lower Legs	73.56	73.53	88.73	70.61	71.80	94.50	86.07	78.95	97.00
All body parts	72.42	69.41	85.23	66.05	64.97	83.16	75.26	69.68	87.59
Average (Actors)		75.69			71.39			77.51	
All individuals (global PCP)		77.3			71.75			79.00	

6

Human Pose Estimation in the Operating Room

In the previous chapters, we have addressed the problem of human pose estimation in different environments, unbounded to a particular application. In this chapter, though, we focus on the scenario of the operating room (OR), where we aim to estimate the 3D body pose of the surgeons and staff. This problem is of particular interest and importance for the surgical workflow analysis [3, 135]. Surgical workflow models are build in order to derive and analyse statistical properties of a surgery for recovering the phase of the operation, staff training, data visualization, report generation and monitoring. Building a workflow model requires sufficient amount of data from different sources and sensors. For example, measurements are collected from instruments, medical and monitoring devices [136]. A multi-view camera system that automatically derives the 3D body pose of the surgeons and staff is another input modality to the framework of the surgical workflow modelling. To build such a system, we rely on the proposed methods from the previous chapters. We make use of the 3D pictorial structures (3DPS) model (Chapter 5) and the 2D deep regressor (Chapter 4) on estimating the 3D pose of the medical staff inside OR. In practice, we combine the deep regressor 2D body pose predictions with the 3DPS model for estimating the pose of multiple individuals from multiple views.

Installing a camera system inside the OR, capturing data and annotating it is a complicated process. Moreover, the access to this kind of data is usually very limited. In this chapter, we present a unique dataset for human pose estimation in the operating room (OR). We have captured a simulated medical operation using a multi-view camera system (Figure 6.1). This dataset is used for evaluation of our methods on multiple human pose estimation from multiple views. In the experimental section, we report our results that show successful applicability of our models in the OR. Moreover, we demonstrate visually that the derived 3D body poses can be beneficial for the identification of the actions of the medical staff, in this challenging environment.

6.1 Introduction

Several times within the previous chapters, we have exhibited the benefits of estimating automatically the human body pose. In this chapter, the problem of human pose estimation inside the OR is tackled, where the goal is to derive the 3D body pose of the surgeons and staff. We choose this application for the following reasons: Firstly, the OR environment is complex, dynamic and crowed. Therefore, we consider it as a challenging scenario for investigating the robustness of our algorithms. Secondly, human pose estimation in OR serves as an additional signal within the framework of surgical workflow analysis. The 3D estimated body poses of multiple individuals over time can comprise features for learning workflow models. Our claim is also supported by the fact that the body pose has been characterized as a very discriminative feature for action recognition, a related task to workflow modelling [89].

Figure 6.1: **OR dataset**: We introduce the OR dataset for human pose estimation in the operating room (OR).

The lack of any publicly available dataset for our task has motivated us to create the operating room (OR) dataset. The OR dataset simulates a medical operation (not a particular) in a real OR. Five experts simulate different phases of a medical operation using a phantom as patient. The dataset is composed of two surgeons and a staff of three. The surgeons and staff interact with each other in a dynamic environment, which has been captured using multiple RGB cameras. In Sec. 6.2, we will provide the details of the OR dataset for multiple human body pose estimation.

Estimating the 3D body pose would be possible using marker-based systems as well [159]. However, this is not feasible in a real OR where sterilization is a limiting factor. After discussions with several surgeons, we have also noticed that placing markers on the medical staff is not ergonomic. Consequently, a marker-less system fits well to the OR scenario. In order to perform human pose estimation in OR, we rely on our methodology from Chapter 4 and 5. In Chapter 4, we have introduced a convolutional neural network (ConvNet)

for regressing body poses of localized individuals. We adapt this model to our current problem and use it for predicting 2D body poses in OR. For each camera view, the individuals are first localized using a tracker [24] and then the 2D body pose is extracted using the ConvNet of Chapter 4. In our application, we make use of the ConvNet without the coarse-to-fine model for relaxing the computations. In addition, we use a second ConvNet with the same architecture (Figure 4.3), but different loss function. The second ConvNet is used as a body part classifier and thus we use a softmax loss to train it. As a result, the 2D body pose predictions of the first ConvNet are classified using the second ConvNet, for obtaining a confidence value for each prediction.

Similar to Chapter 5, we define a 3D pictorial structures (3DPS) model, where each body part corresponds to a body joint. After obtaining 2D predictions from all views using the ConvNets, we rely on the 3DPS model for inferring the 3D body pose. In Sec. 6.3, we provide all the details about the models and the training process.

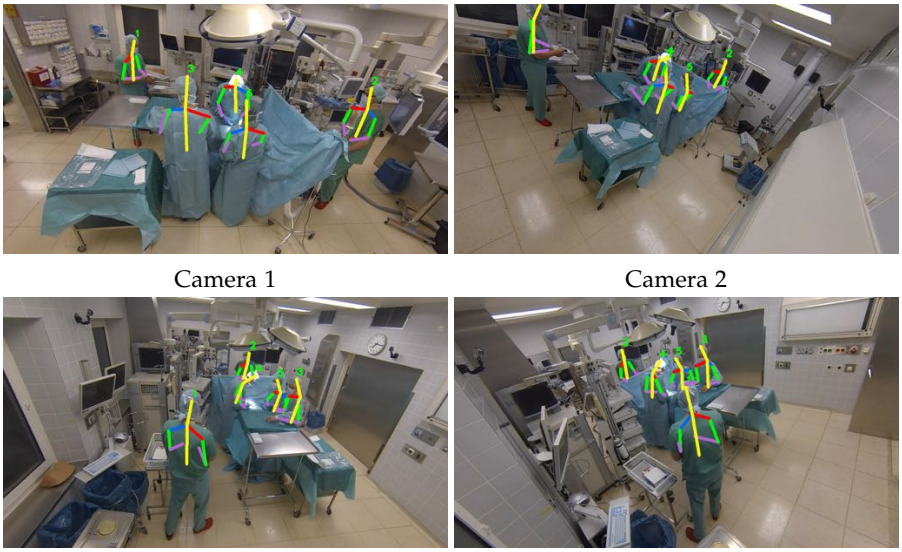


Figure 6.2: **OR dataset with results on 3D human pose estimation**: The dataset is composed of five cameras, while here we present the results on four cameras. The 3D body poses are projected across the camera views.

6.2 OR Dataset

The operating room (OR) dataset is composed of five RGB cameras positioned in different locations of a real OR. In Figure 6.2, we present samples of the same time step in different camera views. The main idea is to simulate different phases of an operation, where there is active collaboration between the surgeons and staff. Note that we do not aim to recover the pose of the full body due to

high occlusion in the lower body. Our goal is perform upper body 3D pose estimation instead, which can be valuable for the task of workflow modelling.

6.2.1 Acquisition and Annotation

We have mounted five GoPro® cameras on the walls of an OR for capturing the dataset. The cameras do not have an internal wired synchronization system and thus we have manually synchronized them after the recordings. The camera calibration has been done using the geometrical pattern of the floor[1] [79]. To derive the gound-truth 2D body pose (Figure 6.1), we manually annotated the image data for all camera views. Afterwards, we performed triangulation for generating the 3D body pose ground-truth. In total, we performed two different recordings for creating training and testing datasets. In the first recording all individuals are associated with a particular role, while the roles are changed in the second recording. Hence, we create additional variation in the body motion of the individuals. Since the lighting is controlled in OR, the time difference between the recordings of the training and testing datasets does not have any effect on the recording environment. Finally, we performed the calibration, synchronization and annotations tasks for both recordings.

6.2.2 Dataset

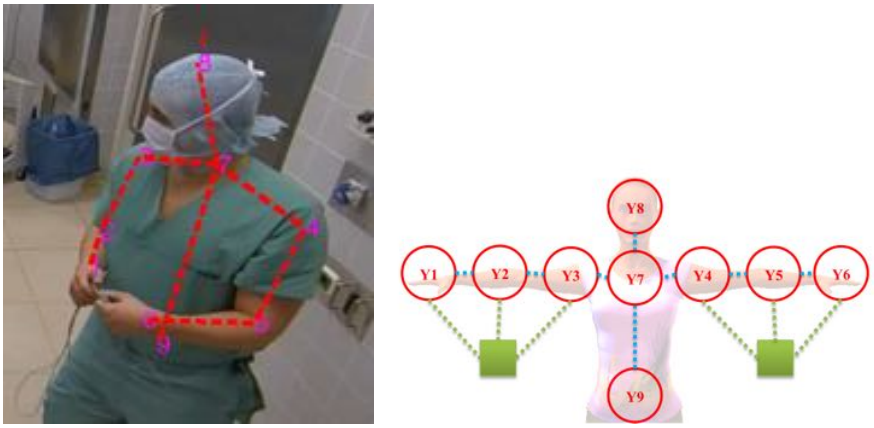
One recording comprises the training dataset and the other the testing dataset. During training our models, we have also formed a validation dataset for selecting the hyper-parameters of the models. The training dataset includes 3000 images with up to 5 individuals for each camera view. Similarly, the testing dataset has up to 5 individuals in the scene but, it is composed of 4000 frames. In both cases, we provide annotation in every 10th frame. Next, we describe the training of our models for this type of data and then present the experimental outcome.

6.3 Experiments

The application of human pose estimation in the OR comes with a specific type of image data. For that purpose, we cannot employ generic models from the previous chapters. It is required to train new models both for 2D and 3D human pose estimation. In this section, we discuss the training details of the ConvNets and CRF. At first, the training is performed for the 2D model so that body pose candidates are obtained in the image plane and for each camera view. Then, it follows the margin-based parameter learning of the CRF.

The evaluation is performed in two steps. At the beginning, we examine the 2D human model and later the 3DPS model applied on upper body pose estimation. For all evaluations, we rely on the *strict* PCP evaluation metric both for 2D and 3D body pose estimation.

[1] We cordially thank Xinchao Wang for performing the calibration task and Kiyoshi Hashimoto for annotating the dataset.



2D Human Model 3D Human Model (3DPS Model)

Figure 6.3: **Human model**: On the left, the 2D human model is presented. It has 9 body joints which are regressed using a ConvNet. Moreover, a confidence value is obtained for each regressed joint using a second ConvNet for classification. The symmetric joints count for a single class and thus we have in total 6 classes $(1-6, 2-5, 3-4, 7, 8, 9)$. On the right, the 3D human model is presented. We model it using a CRF, where the blue edges correspond to pairwise potentials and the green one to ternary potentials. The pairwise potentials model the translation between the body parts, while the ternary the rotation.

6.3.1 2D Human Model

The 2D human pose estimation of the individuals is performed with a convolutional neural network (ConvNet), as it has been presented in Chapter 4. Using the training data of the OR dataset, we train a ConvNet for regressing the 2D body joints and another for classifying the regressed joints. The 2D model (see Figure 6.3) is composed of 9 body joints and 6 classes (the symmetric joints count for a single class). The training process takes place for each camera view separately. We do not train a single model for all camera views due to the significant viewpoint variation between the camera views (Figure 6.2). Furthermore, we apply the same data augmentation and normalization as in Chapter 4.

During prediction, the individuals are localized using tracking [24] and then the first ConvNet is used for obtaining 2D body pose candidates. Afterwards, the second ConvNet is employed for acquiring a confidence value for the regressed body joints of each individual. This step is applied on the training dataset to create training data for the training of the 3DPS model.

6.3.2 3D Human Model

Once the training of the 2D model is completed, it follows the parameter learning of the 3DPS model. The 3DPS model has 9 random variables for modelling the body parts (i.e. body joints). Furthermore, the upper body's prior models the translation and rotation between body parts using pairwise

and ternary potential functions (Figure 6.3). Note that we skip the collision potential functions (see Chapter 5) because we do not face the problem of false positive candidates between the symmetric body parts using the ConvNet regressor in the 2D space. In order to learn the parameters of the model, we employ an S-SVM, as in Chapter 5.

The inference task is one more time accomplished by belief propagation. During inference, we rely on the ConvNets for producing body part candidates in the 2D space across all views. Then, we create our discrete state space from which the inference is performed with the max product algorithm. Next, we evaluate first the 2D human model and afterwards the 3DPS model on the OR dataset.

6.3.3 Evaluation in 2D

In this part, we evaluate the performance of the ConvNets to regress the 2D body joints across each view. To that end, we estimate the PCP scores of all individuals jointly for each camera view. The results are summarized in Table 6.1. We do not distinguish the score of the different individuals, instead we consider them as a joint testing set, similar to the 2D human body pose evaluation in Chapter 3 and 4.

In Table 6.1, we observe similar performance of the body parts across the different camera views. The localization of the head and torso is quite precise for all cameras, while the lower arms are proven to be the most challenging body part to be correctly predicted. At the end, the full body localization is also similar for all cameras. The classification error of the second ConvNet is also similar for all camera views. In particular, it is 30.30% for Camera 1 and 28.73% for Camera 5. Camera 2, 3 and 4 have slightly higher classification error that is 35.26%, 33.84% and 35.22%. We provide visual results of the 2D body pose prediction in Figures 6.4 and 6.5.

Table 6.1: **2D Human Pose Evaluation**: The evaluation on 2D human pose estimation is presented for each camera view. We have used the *strict* PCP performance metric. The last row summarizes the global PCP score.

	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Head	97.90	91.67	98.15	93.85	97.66
Torso	81.76	92.89	91.46	92.08	91.66
Upper Arms	91.13	69.69	80.03	76.13	87.16
Lower Arms	50.59	48.73	46.56	42.79	51.55
Full Body	77.17	70.23	73.80	70.63	77.79
All individuals (global PCP)			73.70		

6.3.4 Evaluation in 3D

The second part of the evaluation is related to the 3D human pose estimation. Now, we move to the 3D space and examine the body pose of each individual independently. The results are summarized in Table 6.2.

Head and torso parts are the most correctly inferred body parts for the individuals. On the other hand, the PCP score is low for the lower arms. The lower arms remain the most difficult part to infer even with multiple camera views as input. The results on the upper arms are different between the individuals, with the Actor 4 having the best performance. In general, Actor 4 has the best performance among the others, stemming from his excellent position that is well captured by Camera 1 and 5. Comparing the global PCP score between the 2D and 3D human pose estimation, we note that the 3D results are around 10% lower due to the higher dimensional output space. Inference in the 3D space is a more difficult and demanding task than in 2D space, but it does not result in significant lower performance. We demonstrate our results in Figure 6.2, 6.6 and 6.7. The human localization results are presented in Appendix C.

Table 6.2: **3D Human Pose Evaluation**: The evaluation on 3D human pose estimation is presented for each individual. We have used the *strict* PCP performance metric. The last row summarizes the global PCP score.

	Actor 1	Actor 2	Actor 3	Actor 4	Actor 5
Head	93.66	80.00	83.80	99.25	78.27
Torso	83.29	98.25	64.01	97.75	94.24
Upper Arms	78.68	68.50	55.79	95.75	51.05
Lower Arms	34.01	20.88	21.11	77.00	24.61
Full Body	67.05	59.50	50.60	90.42	53.97
All individuals (global PCP)	64.41				

6.4 Conclusions

We have presented our models for 2D and 3D human pose estimation applied on the operating room (OR). To address the problem, we have introduced a unique dataset which has been captured in a real OR and it simulates a medical operation using a phantom as patient. Our results in the OR dataset demonstrate that our algorithms deliver discriminative body poses that can be potential used for modelling the surgical workflow task.

Next, we move away from the task of human pose estimation and present an object tracker. Since we have relied on tracking for localizing humans, we investigate the problem of 2D object tracking in the following chapter.

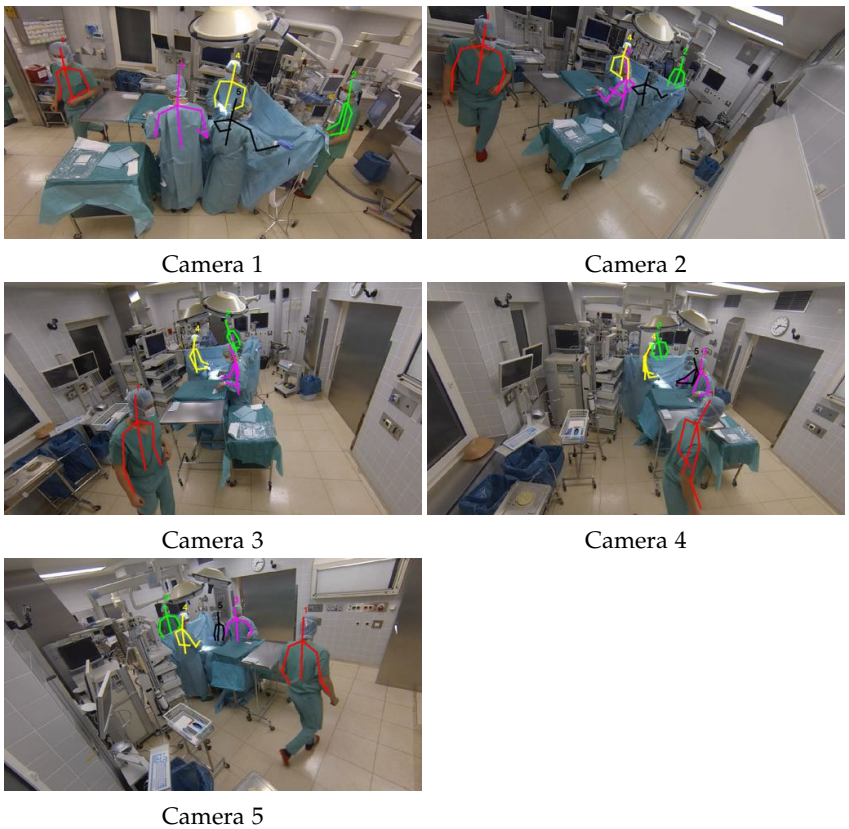


Figure 6.4: **Results on 2D Human Pose Estimation**: Visual results of the 2D human pose estimation task are presented. The presented results are from the same time step across all camera views.

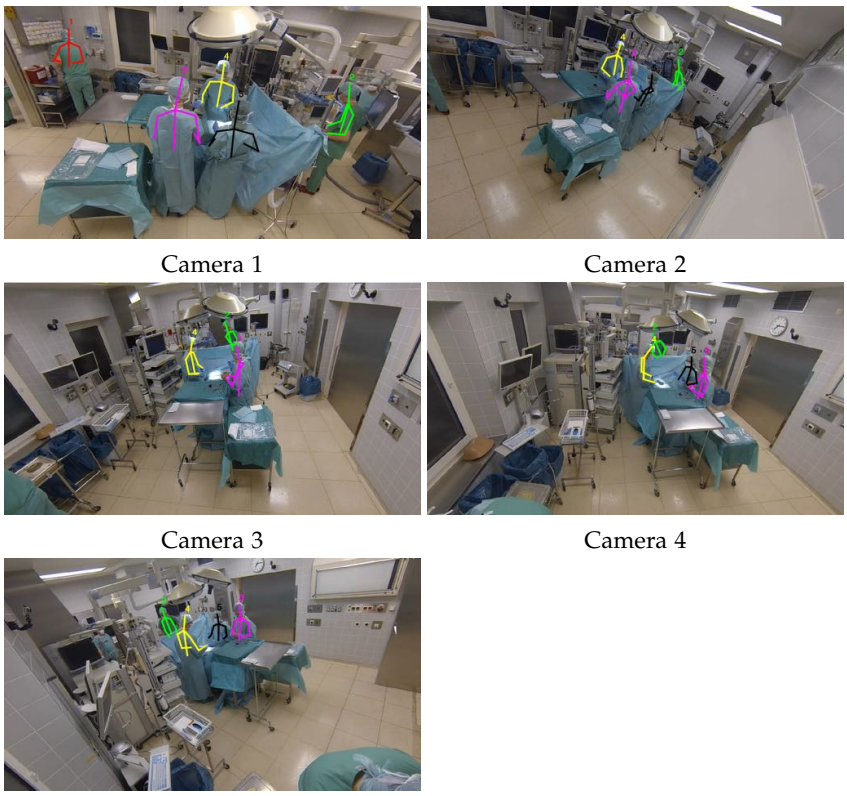


Figure 6.5: **More Results on 2D Human Pose Estimation**: Visual results of the 2D human pose estimation task are presented. The presented results are from the same time step across all camera views.

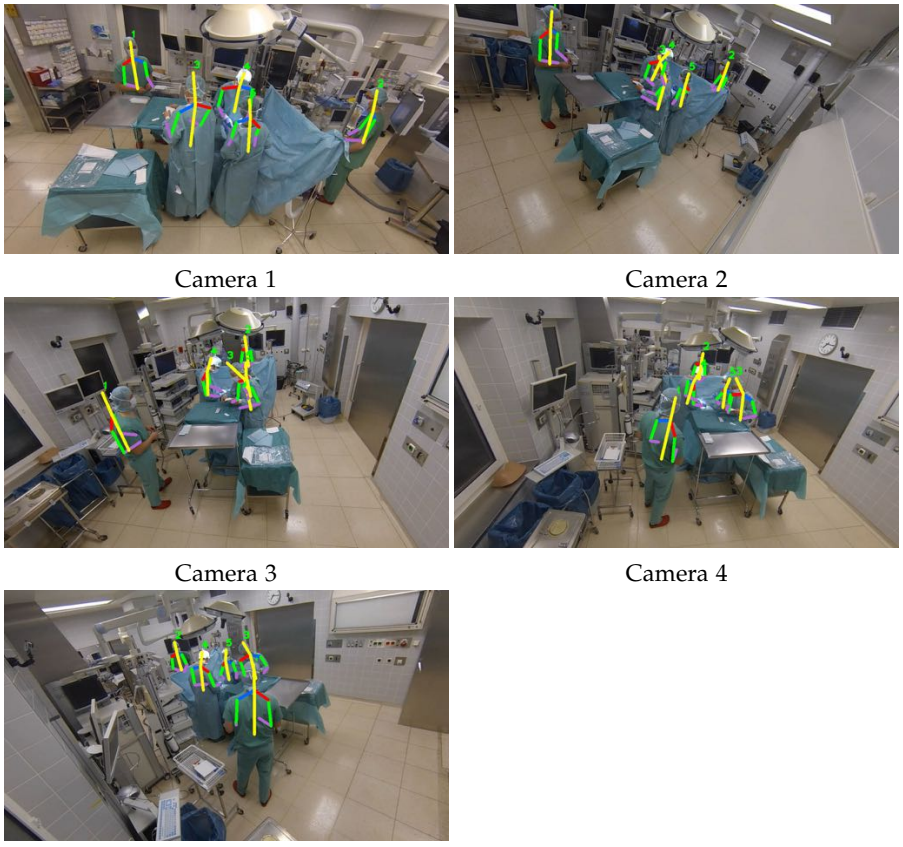


Figure 6.6: **Results on 3D Human Pose Estimation**: Visual results of the 3D human pose estimation task are presented. The inferred 3D body poses are projected across all camera views.

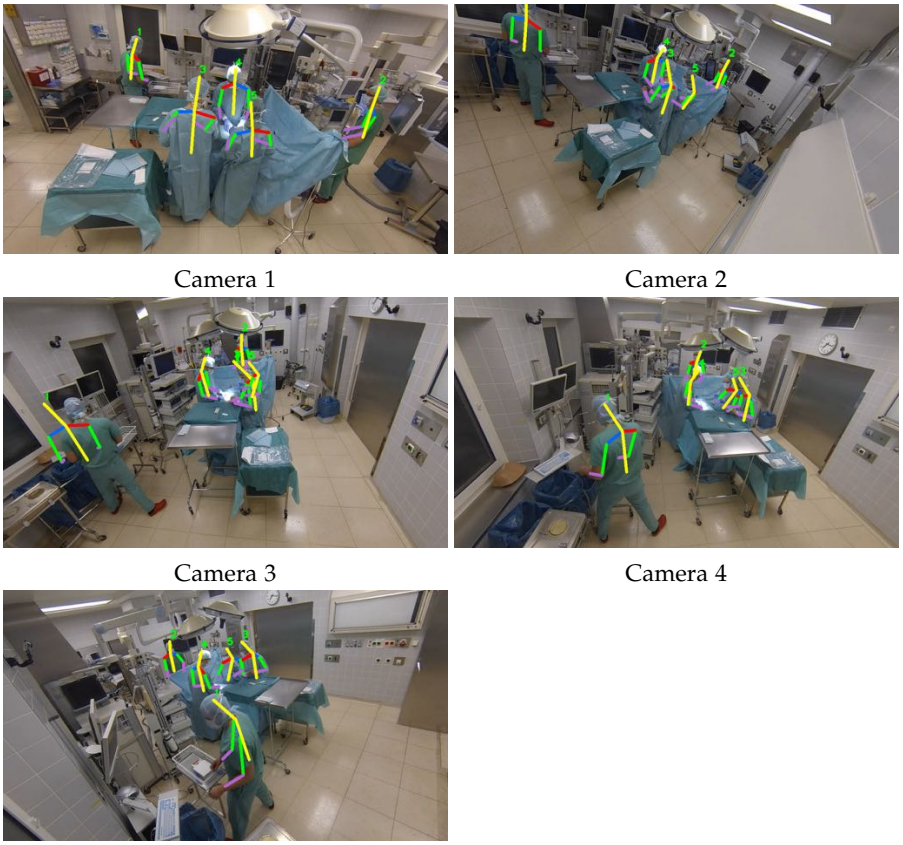


Figure 6.7: **More Results on 3D Human Pose Estimation**: Visual results of the 3D human pose estimation task are presented. The inferred 3D body poses are projected across all camera views.

7

Object Tracking by Segmentation

The main scope of the thesis is to address the problem of human pose estimation in complex environments. Up to this point, we have explored the problem from a single view, as well as from multiple views. In all cases, it has been assumed that the individuals have been localized in the image plane, using a bounding box. In this Chapter, we work on the problem of object localization by considering the temporal aspect. In particular, we address the problem of visual 2D tracking of arbitrary objects that undergo significant scale and appearance changes. Although, we focus on generic object tracking, we propose an algorithm that could be combined with the task of human pose estimation.

The classical tracking methods rely on the bounding box surrounding the target object. Regardless of the tracking approach, the use of bounding box quite often introduces background information. This information propagates in time and its accumulation quite often results in drift and tracking failure. This is particularly the case with the algorithm of particle filtering approach that is often used for visual tracking. However, it is always used a bounding box to sample observations by extracting image features for each particle sample. Since the sampling based on a bounding box can cause drifts, we propose to use segmentation for sampling. Relying on segmentation and computing the colour and gradient orientation histograms from these segmented particle samples allows the tracker to easily adapt to the object's deformations, occlusions, orientation, scale and appearance changes. We propose two particle sampling strategies based on segmentation. In the first, segmentation is done for every propagated particle sample, while in the second only the strongest particle sample is segmented. Depending on the strategy there is obviously a trade-off between speed and performance. In the experimental section, we apply the proposed tracker into different type of objects and scenarios in order to show that pixel-wise tracking is very beneficial for complex environments.

Figure 7.1: **Tracking Results**: From top to the bottom row respectively sequences are named: *Mountain-bike*, *Entrance*, *UAV*, *Cliff-dive 1*. The *Entrance* sequence has been captured with a stationary camera while in the other three sequences both the object and camera are moving

7.1 Introduction

Visual object tracking is a classic problem in the computer vision community. It is essential for numerous applications, such as surveillance [39], action recognition [84], augmented reality [181] as well as human pose estimation. In Chapter 5, we have relied on tracking for human localization across different views. One of the classical approaches for object tracking is particle filtering. It generalizes well to any kind of objects, models well non-Gaussian noise and is able to run in real-time. The observation models that have been used with particle filtering are mostly colour histograms [139] and histograms of oriented gradients [117], where the histograms are computed from a bounding box surrounding the target object. While using bounding boxes is fast and convenient, they often capture undesirable background information as most objects do not fit into a rectangle very well. This information is further propagated to all sample particles and often causes drift. This is particularly true for deformable objects, like humans, where the bounding box sometimes includes very large portions of the background.

The recent trend in visual tracking is related to learning the object's appearance. The tracking then becomes a classification problem where the goal is to discriminate the object of interest from the background [11]. The appearance of the object can be learned offline or online. These approaches are traditionally called *tracking-by-detection* or online learning approaches and have performed very well on demanding tracking scenarios, e.g. sport activities, pedestrian

tracking or vehicle tracking. Although they are often robust against occlusions, deformations, orientation, scale and appearance changes, their computational cost makes most of them inefficient for real-time applications. In addition, the presence of false positive detections causes drifting. The drifting is closely related to the area from which the object's features are extracted. It is usually determined from a rectangular bounding box. However, the object does not usually fit perfectly inside the box, so the additional background information is included in the extracted features. For instance, this results in learning background in the trackers based on the online learning of the object appearance. Again, the presence of the background information becomes more critical for deformable objects where the bounding box always includes that type of noise. To overcome this problem Godec et al. [74] recently proposed an approach that removes a bounding box constraint and combines segmentation and online learning. However, due to the very expensive learning procedure based on Hough forests the efficiency of this tracker is far from real time.

In this chapter, our objective is to overcome majority of the above limitations and propose a general purpose tracker that can track arbitrary objects whose initial shape is not a priori known in challenging sequences and real-time. These sequences contain clutter, partial occlusions, rapid motion, significant viewpoint and appearance changes (Figure 7.1). We propose to use the standard particle filter approach based on colour and gradient histograms and incorporate the object shape into the state vector. Since the classical particle filter based on bounding box surrounding particle samples drifts due to sometime abrupt amount of captured background, we propose to use segmentation at the particle sample locations propagated by a basic dynamic motion model. This allows having particle samples of arbitrary shapes and collecting more relevant regions features, in a pixel-wise level, than when the bounding box is used. Consequently, the object state vector strongly depends on the object's shape. Relying on segmentation allows the tracker to easily adapt to the object's deformations, occlusions, orientation, scale and appearance changes. We propose two particle sampling strategies based on segmentations. In one case the segmentation is done for every propagated particle sample and therefore is more robust to large displacements, scales and deformations, but it is more time consuming. The other strategy is to do the segmentation on the particle sample with the highest importance weight and propagate its shape to all other samples. This is definitely less robust and more critical in difficult sequences where object shape and position change dramatically from frame to frame, but in all other sequences, where this is not the case, is sufficient and comes with the great computational complexity reduction leading to very fast runtime of up to 50 fps. Depending on this decision, there is obviously a trade-off between speed and performance.

For the rest of the chapter, we first focus on the related work for object tracking and afterwards introduce our pixel-wise tracker that builds on segmentation. Finally, we test the proposed algorithm with related approaches in the evaluation section and highlight the advantages of performing tracking without the limitations of a bounding box.

7.1.1 Related Work

Object tracking in 2D is well studied problem with vast amount of literature [112, 190]. In this section, we focus on approaches related to particle filtering, learning-based and methods that do not rely on rectangular bounding box localization. Starting from the probabilistic methods, Isard and Blake [86] introduced the particle filter, namely condensation algorithm, for tracking curves. Later on, the method was also applied to colour based tracking [131]. Similarly, Pérez et al. [139] proposed a colour histogram based particle filtering approach. However, the colour distribution fails to describe an object in situations where the object is of a similar colour as the background. For that reason, Lu et al. [117] incorporated a gradient orientation histogram in the particle filter. The most common particle filtering algorithm, the bootstrap filter [50], has been combined with a classifier [132] in order to be created an advanced motion model. All these methods rely on bounding boxes for sampling and therefore are sensible to the particle samples erroneously taken from the background. A more recent approach combines an off-line and an online classifier in the bootstrap filter's importance weight estimation [39]. In all cases, incorporating a classifier into the particle filter has an important impact on the runtime.

In the domain of a unified tracking and segmentation, the object is presented from a segmented area instead of a bounding box. Particularly impressive is the probabilistic approach of Bibby and Reid [29]. They have combined the bag-of-pixels image representation with a level-set framework, where the likelihood term has been replaced from the posterior term. Even though this approach adapts the model online and is not based on the bounding box, it is susceptible to the background clutter and occlusions. Chockalingam et al. [43] divided the object into fragments based on level-sets as well. Recently, Tsai et al. [175] have proposed a multi-label Markov Random Field framework for segmenting the image data by minimizing an energy function, but the method works only offline. The complexity of all these methods increases their computational cost significantly. In addition to the object segmentation, Nejhum et al. [154] have used a block configuration for describing the object. Each block corresponds to an intensity histogram and all together share a common configuration. This representation forms the searching window which is iteratively updated. Nevertheless, the bounding box representation is still present but in a small scale. Finally, Duffner and Garcia [51] have presented a tracking-by-segmentation approach that relies on pixel-based descriptors and object segmentation.

The first work on learning-based approaches was published by Avidan [11] and Javed et al. [88], where tracking is defined as a binary classification problem. A set of weak classifiers is trained online and afterwards boosted to discriminate the foreground object from the background. The idea of online training has been continued by Grabner et al. [75] for achieving a real-time performance in a semi-supervised learning framework. In this approach, the samples from the initialization frame are considered as positive for online training and during the runtime the classifier is updated with unlabelled data. Babenko et. al [13] have proposed a multiple instance learning (MIL) approach for dealing with

the incorrectly labelled data during the training process. The MIL classifier is trained with bags of positive and negative data, where a positive bag contains at least one positive instance. More recently, Kalal et al. [93] have combined the KLT tracker [118] with an online updated randomized forest classifier for learning the appearance of the foreground object. The tracker updates the classifier and the classifier reinitializes it in case of a drift. Similarly in [102], the appearance model of the tracker evolves during time. All the above approaches present mechanisms for preventing the drifting effect in some form. However, they are all trained with data extracted from a bounding box. As a result, background information is highly probable to penetrate into the training process which will eventually lead to drift assuming arbitrarily shaped objects. Another learning-based tracker with promising results [185] is STRUCK [77] which relies on structured prediction. Sparse coding has been also used for learning offline and online the object appearance in order to perform robust tracking [90, 113].

Godec et al. [74] have gone a step further into online learning by removing the rectangular bounding box representation. They have employed the Hough Forests [66] classification framework for online learning. In this approach, the classification output initializes a segmentation algorithm for getting a more accurate shape of the object. The approach is relatively slow, but it delivers promising results on demanding tracking sequences. In our work, we similarly make use of the segmentation concept as well but we incorporate this into a much faster particle filter tracker instead of using a non-bounding box classification approach.

7.2 Particle Filter Based Visual Object Tracking

The particle filter has shown to be a robust tracking algorithm for deformable objects with non-linear motion [86]. The tracking problem is defined as a Bayesian filter that recursively calculates the probability of the state $\mathbf{x}_t$ at time t, given the observations $\mathbf{z}_{1:t}$ up to time t. This requires the computation of the (probability density function) pdf $p(\mathbf{x}_t \mid \mathbf{z}_{1:t})$. It is assumed that the initial pdf $p(\mathbf{x}_0 \mid \mathbf{z}_0) = p(\mathbf{x}_0)$ of the state vector, also known as the prior, is available. $\mathbf{z}_0$ is an empty set indicating that there is no observation. In our problem the state consists of the object's shape S and 2D position of the shape's centre of mass x_c, y_c and is defined as $\mathbf{x}_t = [x_c, y_c, S]^T$. The prior distribution is estimated from the initial object shape. The initial shape can be either manually drawn or estimated from segmenting a bounding box which surrounds the object. Finally, the pdf $p(\mathbf{x}_t \mid \mathbf{z}_{1:t})$ can be computed from the Bayesian recursion, consisting of two phases called prediction and update. Assuming that the pdf $p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1})$ is available and the object state evolves from a transition model $\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{v})$, where $\mathbf{v}$ is a noise model, then in the prediction phase the prior pdf $p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1})$ at time t can be computed using the Chapman-Kolmogorov equation:

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}) d\mathbf{x}_{t-1} \tag{7.1}$$

The probabilistic model of the state evolution $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ is defined by the transition model. When at time t an observation $\mathbf{z}_t$ becomes available, the prior can be updated via Bayes' rule:

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t \mid \mathbf{x}_t)p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1})}{p(\mathbf{z}_t \mid \mathbf{z}_{1:t-1})} = \tag{7.2}$$
$$= \frac{p(\mathbf{z}_t \mid \mathbf{x}_t)\int p(\mathbf{x}_t \mid \mathbf{x}_{t-1})p(\mathbf{x}_{t-1} \mid \mathbf{z}_{t-1})\mathrm{d}\mathbf{x}_{t-1}}{\int p(\mathbf{z}_t \mid \mathbf{x}_t)p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1})\mathrm{d}\mathbf{x}_t}$$

where the likelihood $p(\mathbf{z}_t \mid \mathbf{x}_t)$ is defined by the observation model $\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{n}_t)$ with known statistics $\mathbf{n}_t$. In the update phase, the observation $\mathbf{z}_t$ is used to update the prior density in order to obtain the desirable posterior of the current state. The observation in our method comes from colour $p(\mathbf{z}_t^{col} \mid \mathbf{x}_t)$ and gradient orientation $p(\mathbf{z}_t^{or} \mid \mathbf{x}_t)$ histograms.

Since posterior density cannot be computed analytically, it is represented by a set of random particle samples $\left\{\mathbf{x}_i^t\right\}_{i=1\cdots N_s}$ with associated weights $\left\{\mathbf{w}_i^t\right\}_{i=1\cdots N_s}$. The most standard particle filter algorithm is Sequential Importance Sampling (SIS). Theoretically, when the number of samples becomes very large, this so called Monte Carlo sampling becomes an equivalent representation to the usual analytical description of the posterior pdf. Each particle sample represents a hypothetical object state and it is associated with an importance weight. The calculation of the weight is based on the observation likelihood and weight from the previous time step.

However, a common problem with the SIS particle filter algorithm is the degeneracy phenomenon. This means that after a few iterations the majority of particles will have negligible weight. To overcome this problem the bootstrap filter, which is based on the Sampling-Importance-Resampling (SIR) technique, aims to remove low importance samples from the posterior distribution. When the number of particle samples with high importance weight drops under a constant threshold, the resampling step is executed. There, every sample contributes to the posterior with proportion to its importance weight. The weight estimation is given by:

$$\mathbf{w}_t^{(i)} = \mathbf{w}_{t-1}^{(i)} \cdot p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)}) = \mathbf{w}_{t-1}^{(i)} \cdot p(\mathbf{z}_t^{col} \mid \mathbf{x}_t^{(i)})p(\mathbf{z}_t^{or} \mid \mathbf{x}_t^{(i)}), \sum_{i=1}^{N_s} \mathbf{w}_t^{(i)} = 1 \tag{7.3}$$

After the resampling step, the samples are equally weighted with $\mathbf{w}_{t-1}^{(i)}$ being constant (i.e. $1/N_s$). The importance weight calculation cost is increased linearly with the number of the the particle samples. Detailed description and discussion of particle filtering can be found in [50]. Next, we detail elements of our particle filtering approach including observation and transition model as well as the segmentation of the particle samples.

7.2.1 Observation Model

Our observation model relies on two components, the colour and gradient orientation histograms. Concerning the colour information, we use the HSV space similar to [139] since it is less sensitive to illumination changes. The

colour distribution is invariant to rotation, scale changes and partial occlusion. For the gradient orientation histogram, we compute the histogram of oriented gradients (HOG) descriptor [47]. The strong normalization of the descriptor makes it invariant to illumination changes.

The likelihood of the observation model $p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)})$ for each particle sample $i = 1 \dots N_s$ is calculated from the similarity between the current $\mathbf{q}(\mathbf{x}_{t-1}) = \{q_n(\mathbf{x}_{t-1})\}_{n=1,\dots,N_c}$ and the predicted state $\mathbf{q}(\mathbf{x}_t) = \{q_n(\mathbf{x}_t)\}_{n=1,\dots,N_c}$ distributions represented by colour histograms, where N_c is the number of colour bins. The state distribution of the gradient orientation histogram is formulated in the same way. We use the Bhattacharyya coefficient $\rho[\mathbf{q}(\mathbf{x}_{t-1}), \mathbf{q}(\mathbf{x}_t)] = \sum_{i=1}^{N_c} \sqrt{q_i(\mathbf{x}_{t-1}) q_i(\mathbf{x}_t)}$ for measuring the similarity of two distributions. As a result, the distance measure is equal to $d = \sqrt{1 - \rho[\mathbf{q}(\mathbf{x}_{t-1}), \mathbf{q}(\mathbf{x}_t)]}$. In the proposed method, likelihoods of both colour and gradient orientation histograms are estimated using the Bhattacharyya coefficient and an exponential distribution, resulting in $p(\mathbf{z}_t^{col} \mid \mathbf{x}_t^{(i)}) = e^{-\lambda d_{col}}$ being the colour likelihood and $p(\mathbf{z}_t^{or} \mid \mathbf{x}_t^{(i)}) = e^{-\lambda d_{or}}$ being the gradient orientation likelihood. The final importance weight is consequently given by:

$$\mathbf{w}_t^{(i)} = p(\mathbf{z}_t^{col} \mid \mathbf{x}_t^{(i)}) p(\mathbf{z}_t^{or} \mid \mathbf{x}_t^{(i)}) = e^{-\lambda d_{col}} e^{-\lambda d_{or}} \tag{7.4}$$

where λ is a scaling factor. While d_{col} and d_{or} are the distances of the colour and orientation histogram respectively.

7.2.2 Transition Model

The transition model of the particle filter has the same importance as the observation model for achieving an accurate forward inference. The variance and/or non-linearity of the motion of different objects do not allow to use a simplified motion model, like the constant velocity in [39]. In our work, the transition model of the particle filter is based on a learnt second order autoregressive model. The Burg method [165] is used for deriving two second order autoregressive functions, independently for the x and y direction. The last term of the object's state, the shape, is represented by a constant term in state space, which is estimated from the segmentation.

7.2.3 Segmentation of the Particle Samples

The particle filter algorithm treats the uncertainty of the object's state by estimating the state's distribution. In the state model we introduce the shape term S for discriminating the foreground object from the background information during sampling. The shape term is assumed to be known while a segmentation algorithm is employed for estimating it. Finally, the sample's observation is free of background during the likelihood $p(\mathbf{z}_t \mid \mathbf{x}_t^{(i)})$ estimation.

In the current work, the choice of the segmentation algorithm is important. We require that the segmentation algorithm is fast, generic and provides two-class segmentation output. Therefore, we chose the *GrabCut* algorithm, a

graph-cut segmentation approach [147]. The algorithm is incorporated with the particle filter for refining the shape of the particle samples.

The area to be segmented is always slightly larger than the area of the sample's shape. Based on the current shape, an initial bounding box is specified where everything outside of it is considered as background and the interior area is considered uncertain. With such input, *GrabCut* segments the foreground object inside the rectangular area occupied by the particle.

The computational cost of the *GrabCut* algorithm scales with the size of the area which has to be segmented. Even though the speed of the GrabCut is appropriate for small regions of interests like our particle samples, the overall computational complexity grows with the number of particle samples. For that reason we have implemented two different sampling strategies.

7.2.4 Sampling Strategies

To investigate the approximation of the state distribution, we propose two sampling strategies based on the segmentation output. In the first strategy each particle sample is segmented in every iteration in order to refine its shape. We name this sampling strategy a multiple particle filter samples segmentation (*Multi-PaFiSS*) strategy and use this name in our experimental evaluation. The second sampling strategy that we call the single particle filter samples segmentation (*Single-PaFiSS*) strategy is based on segmentation of the sample with the highest importance weight and then propagating its shape to the rest particle samples.

The first sampling strategy is more robust and better adapts to the object's large deformations and scale from frame to frame. However, it comes at the price of increased computational complexity. On the contrary, the second strategy is not that robust to large appearance and scale changes, but it is extremely fast and in many situations also performs well as our experimental validation indicates.

7.2.5 Segmentation Artifacts and Failure

The proposed algorithm is dependent on the segmentation output for refining the shape of the particle samples. Subsequently, a segmentation failure could obstruct the algorithm's pipeline. We identify two possible failure modes. In one case, the segmentation delivers more than one segmented areas of the same class (Figure 7.2). In the second case, the segmentation explodes by including almost the whole area to a single class or segments everything as background. These two common problems can occur when the *GrabCut* algorithm is used.

The first failure mode provides a successful segmentation output. However, there are some small isolated areas, which we call artifacts, that are often present in the output (Figure 7.2). In our experiments, it never happened to have artifacts with an area larger than 5% of the segmented area. By applying a two-pass connected component labelling, we locate the shape with the largest area and exclude the smaller artifacts.

The second failure mode is more critical because we cannot recover a

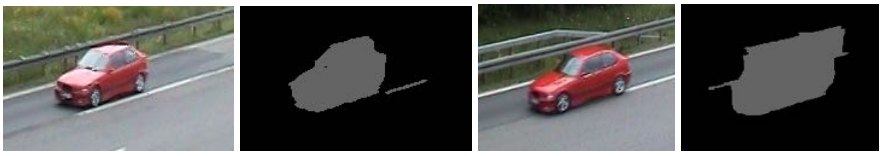


Figure 7.2: **Segmentation Artifacts and Failure**: The figures (a) and (c) show input images. (b) The red car is correctly segmented, but there are two connected components. One is a car and the other is a line marking that is an artifact. We eliminate it by keeping the largest connected component. (d) The segmentation algorithm failed to segment (c) and labeled background as foreground object. In this case the shape of the particle samples becomes rectangular until a new shape is estimated

meaningful segmentation (Figure 7.2). The reason for the failure of the *GrabCut* algorithm is poor quality of the image, failure of the edge extraction and when the colour of the object is not discriminative enough from the background color. Hopefully, this type of failure is easily identified in our algorithm by comparing the current output with the segmentation of the particle sample in the previous time instant based on a threshold. The overlap of the two areas is being compared. In the case of a segmentation failure, the shape of the particle samples becomes rectangular until a new shape is estimated. Thereby, the algorithm continues the tracking task without segmentation refining.

7.3 Experiments

In order to demonstrate the advantages of the proposed algorithm, we evaluate it on standard tracking sequences used in other related work and we also offer five new challenging sequences[1]. For evaluating our algorithm, we have implemented two versions of our method according to the sampling strategy. The evaluation dataset includes videos with objects of different classes that undergo deformations, occlusions, scale and appearance changes. The test video sequences come from the following datasets: *ETH Walking Pedestrians (EWAP)* [138], *Pedestrian dataset* [110], *Comets project* [133] and the *Aerial Action Dataset* [115]. In total, we used 13 sequences for evaluation. The comparison is done with the standard particle filter and three recent approaches. We compare the two versions of our method with the *TLD* [93], *MIL* [13] and *HoughTrack* [74] algorithms.

The evaluation dataset includes the ground-truth annotations in which the target object is outlined by a bounding box in every frame. We use this type of annotation for all test sequences. This type of annotation is not the appropriate way to describe complex objects (e.g. articulated), but it is the standard annotation method. Therefore, our ground-truth are bounding box representations centered on the centre of mass of the segmented area. *HoughTrack* [74] segmentation based tracking algorithm produces bounding boxes for evaluation in the same way. *TLD* [93] and *MIL* [13] have already a bounding box output and they do not require any modification. Then

[1]The evaluation dataset can be found at http://campar.in.tum.de/Chair/PaFiSS

the overlap between the tracker's bounding box and the annotated one is calculated, based on the *PASCAL VOC* challenge [56] overlap criterion. In all experiments, we set the overlap threshold to 50%. Additionally, we evaluate the computational cost of each method by estimating the average number of tracked frames per second (fps) for every sequence.

7.3.1 System Setup

Both versions of our method have fixed parameters for all sequences. There are two parameters which affect the performance of the system: the number of particle samples and the threshold indicating the segmentation failure. Since we do not depend on the bounding box, we found out experimentally that the performance of our method does not increase with the number of the samples. Hence, the number of samples is set to 50 and the segmentation failure threshold to the 40% overlap between two successful consecutive segmentation. All methods have been downloaded from the web and executed with their default settings. All experiments are carried out on a standard Intel i7 3.20 GHz desktop machine.

7.3.2 Comparison to the Standard Particle Filter

The proposed method is compared to the standard particle filter (*SPF*) to prove the superiority of the non-rectangular sampling. For comparison, we implemented the standard bootstrap particle filter [50]. We tested it on all of our sequences but choose the *Entrance* sequence for comparison, since it nicely demonstrates that the amount of background, captured with the bounding box, causes drift. Based on the 50% overlap criterion of the *PASCAL VOC* challenge [56], the standard way of sampling totally fails (Fig 7.3). Since we also noticed that the increase of the number of samples does not increase the performance of *SPF*, we also set it to 50. In contrast, the proposed method excludes the background information from the likelihood estimation and keeps tracking the object until the end of the sequence.

Figure 7.3: **Failure of the Standard Particle Filter**: (a): The overlap over time plot, based on the *PASCAL VOC* challenge [56] criterion, shows the performance of the *SPF* and the two versions of our method. Other images: *SPF* tracker gradually drifts due to collecting background information.

7.3.3 Comparison to the state-of-the-art

The comparison to the latest online learning methods aims to show the outstanding performance of the computationally inexpensive single sampling *Single-PaFiSS* strategy and the more accurate multiple segmentation *Multi-PaFiSS* strategy of our method. Table 7.1 shows that both strategies of our method outperform the other approaches. While Table 7.2 shows that Single-PaFiSS is considerably faster than the other approaches.

We introduce the sequences *Entrance*, *Exit 1*, *Exit 2* and *Bridge* for evaluation of occlusions, scale and appearance changes. All of them come from outdoor and dynamic environments where the illumination varies. Furthermore, the main characteristic of the sequences is the simultaneous motion and deformations of the target objects.

There is a number of sequences where we have achieved better results than the other methods. For instance, in *Actions 2* sequence both of our sampling versions outperform the other methods because of the adaption to the scale changes.

In *Exit 1* and *Exit 2* sequences, both versions of our method and *HoughTrack* give similar results, while *TLD* partially drifts. *MIL* succeeds in *Exit 2* but it does not scale in *Exit 1* sequence. Next, in the *Skiing* sequence the abrupt motion leads *TLD* and *MIL* to complete failure while only *HoughTrack* tracks partially the object until the end. In our algorithm, the segmentation fails to refine the object's shape after some time and the algorithm completely drifts.

In general, we face the segmentation failure problem when the quality of the image data is low, like in the *Pedestrian 1* sequence. As long as the tracker is dependent on the segmentation output for getting the object's shape, a possible failure can cause drift. However, our algorithm continues tracking the object by fitting a bounding box to the most recent object shape and sampling using the bounding box, up to small scale changes. This behavior can be observed in the *Single-PaFiSS* sampling strategy while in *Multi-PaFiSS*, it rarely occurs.

Another segmentation failure can be observed in *Cliff-dive 1* sequence where there is an articulated object in low qualitative image data. Consequently there is high probability that the segmentation can provide incorrect information about the shape of the object. For that reason *Single-PaFiSS* performs better than *Multi-PaFiSS* where there are multiple segmentations per frame. In *Bridge* sequence, our algorithm failed to track the object because there is full occlusion. It is a situation which we do not treat with the current framework. The same failure result occurred with the other approaches.

Taking into consideration the evaluation results, one can conclude that the idea of using a probabilist searching method with the combination of shape based sampling produces a robust tracker. The two evaluated implementations of our method give similar results but *Single-PaFiSS* is considerably faster than all the other methods. Figure 7.1 and 7.4 show some of our results for selected frames.

Table 7.1: **Results for** 13 **sequences**: Percentage of correct tracked frames based on the overlap criterion ($> 50\%$) of the *PASCAL VOC* challenge [56]. The average perentage follows in the end.

Sequence	Frames	Single-PaFiSS	Multi-PaFiSS	TLD [93]	MIL [13]	HT [74]
Actions 2 [115]	2113	82.30	**89.87**	8.18	8.42	8.61
Entrance	196	96.42	**98.46**	35.20	35.20	64.79
Exit 1	186	**100**	**100**	74.19	17.74	**100**
Exit 2	172	96.51	98.83	59.88	95.93	**100**
Skiing [74]	81	13.50	**48.14**	6.17	8.64	46.91
UAV [133]	716	64.26	**88.68**	47.90	58.10	73.46
Bridge	55	10.90	10.90	10.9	**12.72**	12.65
Pedestrian 1 [110]	379	1.84	11.60	**66.22**	56.20	12.40
Pedestrian 2 [138]	352	83.23	94.73	**98.57**	89.20	96.30
Cliff-dive 1 [74]	76	**100**	94.73	55.26	63.15	56.57
Mountain-bike [74]	228	18.85	40.35	36.84	**82.89**	39.03
Motocross 2 [74]	23	**95.65**	69.56	73.91	60.86	91.65
Head	231	82.68	**84.41**	77.05	33.34	61.47
Average		65.53	**70.92**	49.88	47.87	58.73

Table 7.2: **Speed results for** 13 **sequences**: Average frames per second (fps) for every sequence. The total average fps follows in the end.

Sequence	Frames	Single-PaFiSS	Multi-PaFiSS	TLD [93]	MIL [13]	HT [74]
Actions 2 [115]	2113	6.07	0.50	3.76	**19.09**	1.35
Entrance	196	**51.17**	5.79	5.44	20.60	1.75
Exit 1	186	**39.73**	4.17	5.29	21.10	1.83
Exit 2	172	**21.07**	1.92	4.57	17.79	1.57
Skiing [74]	81	**83.67**	4.71	4.25	24.65	2.93
UAV [133]	716	**36.50**	4.30	6.50	27.3	4.58
Bridge	55	**22.17**	1.46	4.38	19.4	1.67
Pedestrian 1 [110]	379	18.82	2.51	5.87	**24.43**	1.56
Pedestrian 2 [138]	352	**29.46**	3.14	2.73	18.72	1.73
Cliff-dive 1 [74]	76	6.46	0.55	8.97	**30.24**	2.48
Mountain-bike [74]	228	**37.79**	3.22	4.53	26.53	2.81
Motocross 2 [74]	23	10.05	1.45	3.95	**23.28**	1.78
Head	231	7.50	0.76	9.74	**34.40**	7.51
Average		**28.23**	2.65	5.38	23.64	2.20

7.4 Conclusion

We have presented a simple yet effective method for tracking deformable generic objects that undergo a wide range of transformations. The proposed method relies on tracking using a non-rectangular object description. This is achieved by integrating a segmentation step into a bootstrap particle filter for sampling based on shapes. We investigated two sampling strategies which

allow a great trade-off between performance and speed. In the first version, we have reached a better performance by segmenting every particle sample while in the second, we have a less accurate but significantly faster algorithm. The proposed algorithm of this chapter has a wide range of applications, including human tracking. In this chapter, we have evaluated it in different video sequences and have observed encouraging results. As a result, it could be coupled with our human pose estimation methodology in an end-to-end system.

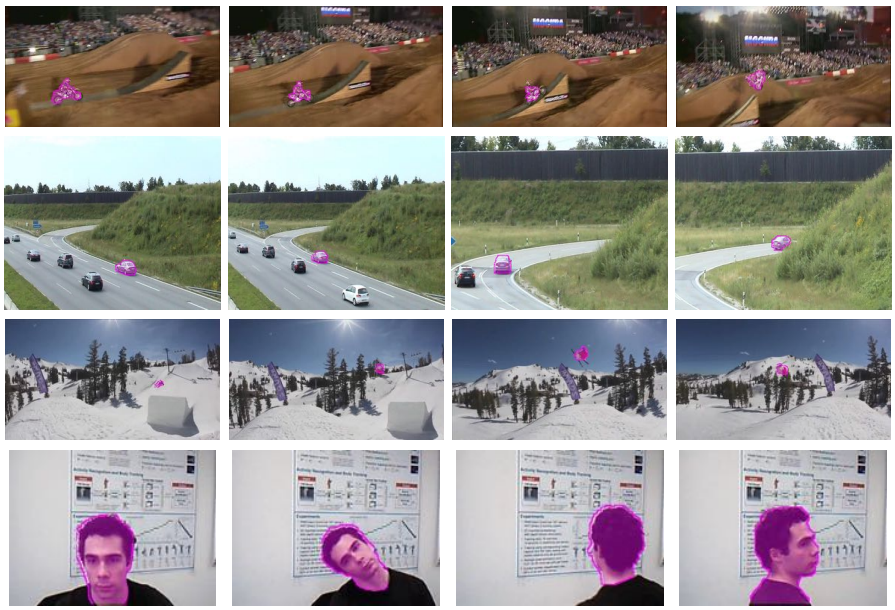


Figure 7.4: **Additional Tracking Results**: (first row: *Motocross 2*, second row: *Exit 2*, third row: *Skiing*, fourth row: *Head*). The *Exit 2* and *Head* sequences have been captured with a stationary camera while in the other two sequences both the object and camera are moving.

8 Conclusion and Outlook

8.1 Summary and Findings

This thesis is devoted to the study of human pose estimation in complex environments. We investigate the problem of single human pose estimation from a single view, as well as, from multiple views. In multiple view camera systems, we also approach multiple human pose estimation. For all these problems, we propose novel algorithms and also demonstrate the algorithms applicability in demanding scenarios. To evaluate our algorithms, we rely on standard datasets, but we also introduce a number of challenging datasets, including the operating room (OR) dataset. Furthermore, we dedicate a chapter to study the absorbing problem of 2D object tracking, which could be eventually used by the human pose estimation algorithms.

Initially, we tackle the problem of 2D human pose estimation from a single view. In Chapter 3, we work with random forests, introducing a novel discriminative model for 2D human pose estimation. Our regression forest relies on engineered features (i.e. HOG) for learning a direct mapping between 2D body poses and image features. In Chapter 4, we argue that feature and model learning have to be performed simultaneously. For that reason, we propose a deep model that is based on convolutional neural networks (ConvNets) and a robust loss function. We show empirically that a robust loss function is valuable for better performance and faster convergence in regression tasks such as 2D human pose estimation. Furthermore, we demonstrate that deep learning achieves very promising results in comparison to random forests and the classic part-based methods.

In Chapter 5, we orient our research towards human pose estimation from multiple views. We propose a 3D pictorial structures (3DPS) model for single and multiple human pose estimation. 3D pictorial structures (3DPS) is a part-based model that copes with 2D body part detectors to sample the observation. Since, the model is composed of several potential functions, we learn the model parameters in order to balance the influence of each potential function over the inference task. The 3DPS model is exhaustively evaluated in several single

and multiple human pose estimation scenarios. In Chapter 6, we adapt the 3DPS model for human pose estimation in the OR. In addition, we introduce a unique dataset that was captured in a real OR and simulates a medical operation. To reach our goal, we use ConvNets, as they are presented in Chapter 4, to generate body pose candidates for the individuals in the 2D space, across all views. Theses candidates contribute to the generation of the state space and, in this way, they are incorporated in the 3DPS model. This combination between a 2D discriminative model and 3D generative model demonstrates promising results in the OR scenario.

In Chapter 7, we move on to 2D object tracking in the image plane and propose a segmentation by tracking algorithm. We strongly believe that object tracking should be accomplished in pixel level, which is more accurate than the commonly used bounding box localization. Our method delivers reliable tracking results for deformable and rigid objects in challenging sequences.

In the thesis, all proposed models offer an algorithmic solution to a certain problem under some assumptions. Next, we discuss the limitations of our methods and afterwards we will propose a number of directions for future work.

8.2 Limitations

We work on the problem of human pose estimation in complex environments and we deliver a number of methods for addressing it. In Chapter 3 and 4, we work with discriminative models for 2D human pose estimation from a single image. Estimating the body pose with engineered features (Chapter 3) is limited to one kind of features, which is probably not equally discriminative for all body regions. To overcome this limitation, we propose to jointly learn the features and human model (Chapter 4). This is definitely a better way to tackle the problem of human pose estimation, but our learnt features are generic up to a particular scale. Our coarse-to-fine approach helps to extract features in higher resolution, but it does not actually models multiple scales. As a result, we find cases where the predicted 2D pose looks fine, but is in fact not correct due to scale issues. However, using a tuple of deep models would not necessarily solve the problem, instead it would add more computational effort.

The 3D pictorial structures model (3DPS) of Chapter 5 and 6 is an effective human model that has demonstrated promising applicability in different scenarios of single and multiple human pose estimation. However, the body prior of the model is very simple and might be less accurate in cases where humans closely interact with each other (e.g. hugging or dancing tango). However, this is a common problem of part based models such as pictorial structures. Moreover, the model relies on the triangulation of body part detections for generating the state space. Consequently, we are bound to a calibrated camera system, where the calibration error is propagated by our model to the inferred 3D body poses. Moreover, geometric ambiguities (e.g. opposite cameras) between different cameras directly affect the performance of our model. These are some limitations of our models that we would like to address in our future

work.

8.3 Future Work

To address the aforementioned limitations, we propose a number of ideas for future research. In 2D human pose estimation, we have experienced that deep learning is an successful direction for future work. However, we believe that we need to move away from standardized input images and build models that cope with input data of different dimensions and modalities as well. Moreover, learning deep models with different type of signals (e.g. image, audio and inertial sensors) can advance the problem of human pose estimation.

We interact with probabilistic graphical models and convolutional neural networks. Both models are based on graphs and share common characteristics. We believe in their unification in the near future. This would mean that we could use learning and inference algorithms from both sides under a common framework. We think that such a powerful model would be able to directly learn geometrical principles and infer 3D body poses without the necessity of calibrating a system. This would be a verification that we can learn geometrical properties using machine learning.

8.4 Epilogue

It has been a long, exciting and laborious way to reach these last lines of the thesis. We have not solved the problem of human pose estimation, but we have set the course for the future research on this topic and substantially contributed to the computer vision community. We focused on multiple human pose estimation from multiple views by combining machine learning models with multiple view geometry. In this thesis, we present models that can be applied in real-world environments and support the human factor by automatizing processes.

A

Deep Regression

We provide additional numerical results as well as more visual results of our method applied on 2D human pose estimation. In particular, an additional comparison between *L*2 and *Tukey's biweight* loss functions follows in Sec. A.1 and more visual results are presented in Sec. A.2.

A.1 Additional Comparisons

In the comparison of *L*2 and *Tukey's biweight* loss functions, we have presented the final result of the full body using the mean squared error (MSE) and PCP (percentage of correctly estimated parts) performance metrics. In this section, we report, in addition, the scores for each body part for the PCP metric.The results are summarized in Figure A.1.

A.2 More Results

In this section, additional visual results using our coarse-to-fine model are presented for the task of 2D human pose estimation. The results are illustrated in Figure A.2 - A.5. Since the biggest improvement using the coarse-to-fine model has been achieved in PARSE [187] and LSP [91] datasets, we provide additional visual results of the refinement in Figure A.6 and A.7.

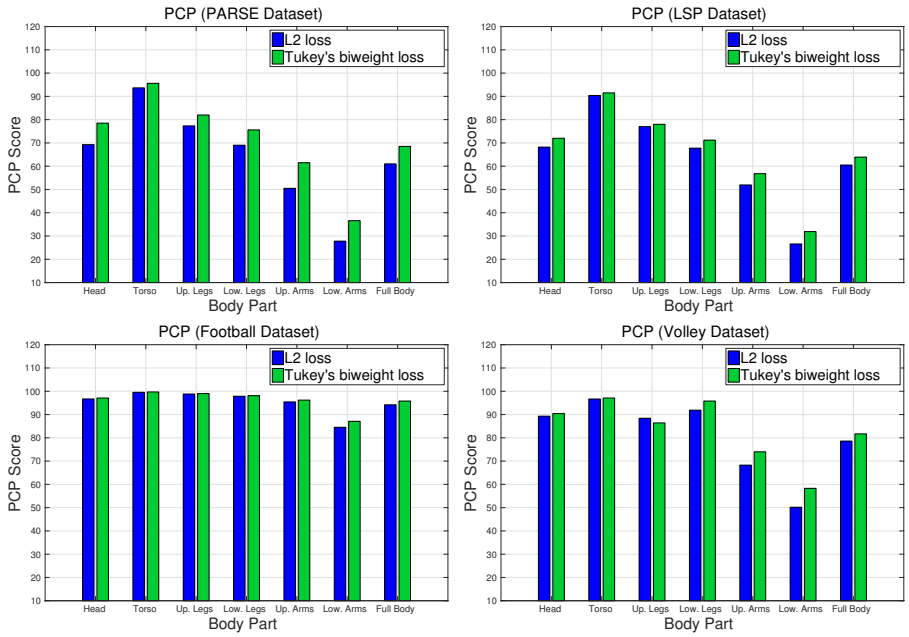


Figure A.1: **Further Comparison of** *L*2 **and** *Tukey's biweight* **loss functions**: We compare our results (*Tukey's biweight* loss) with *L*2 loss for each body part on PARSE [187], LSP [91], Football [95] and Volleyball [15] datasets. In PARSE [187] and LSP [91], the evaluation has been performed using the *strict* PCP, while in Football [95] and Volleyball [15] using the *loose* PCP.

Figure A.2: **Additional results on PARSE [187]**: Samples of our results on 2D human pose estimation are presented.

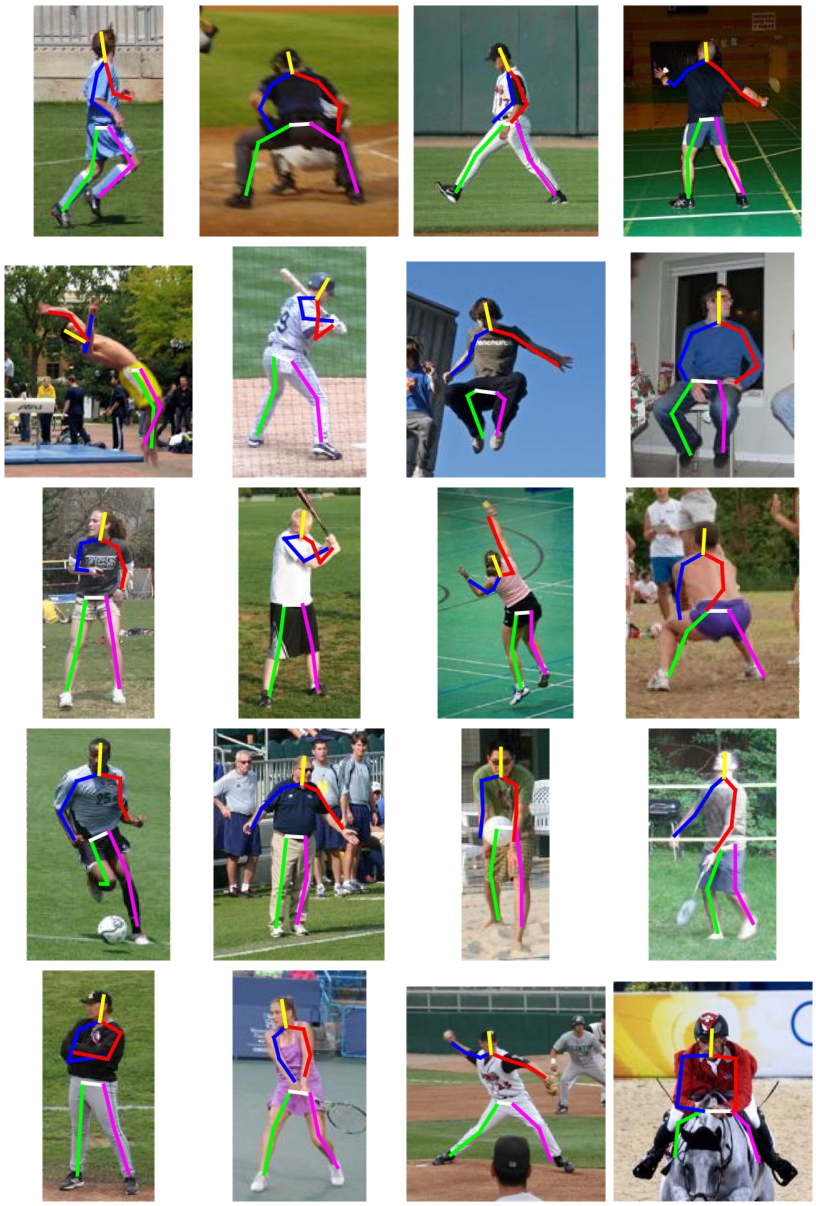


Figure A.3: **Additional results on LSP [91]**: Samples of our results on 2D human pose estimation are presented.

Figure A.4: **Additional results on Football [95]**: Samples of our results on 2D human pose estimation are presented.

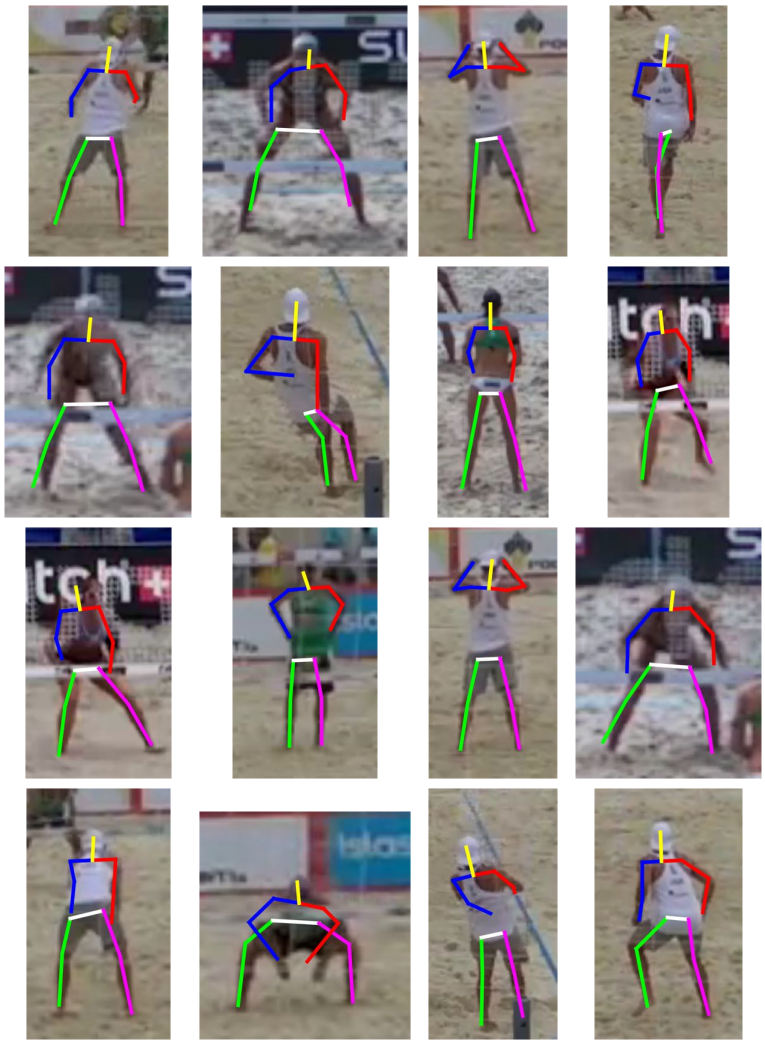


Figure A.5: **Additional results on Volleyball [15]**: Samples of our results on 2D human pose estimation are presented.

Figure A.6: **Additional results on model refinement - PARSE**: Additional results for the coarse-to-fine model using the cascade of ConvNets are presented from the PARSE dataset [187]. On the top row the result of a single ConvNet is presented, while on the bottom row the refined result using the cascade of ConvNets.

Figure A.7: **Additional results on model refinement - LSP**: Additional results for the coarse-to-fine model using the cascade of ConvNets are presented from the LSP dataset [91]. On the top row the result of a single ConvNet is presented, while on the bottom row the refined result using the cascade of ConvNets.

B

3D Pictorial Structures

We present additional results for all sequences. The Figures from B.1 until 5.9 present the results for the single human evaluation. We demonstrate that we can robustly estimate the pose of humans performing different actions in static and dynamic environments.

The Figures from B.3 until B.7 present the results for the multiple human evaluation. We show cases in which we parse the 3D body skeleton of multiple humans in dynamic enviroments. In Figure B.5 and mainly in Figure B.6, there are missing human skeletons due to detection failures (high occlusion) in most of the views.This problem occurs because there are body parts, which are observed only from one view and as a result the body part triangulation fails.

B.1 Part-detector Evaluation

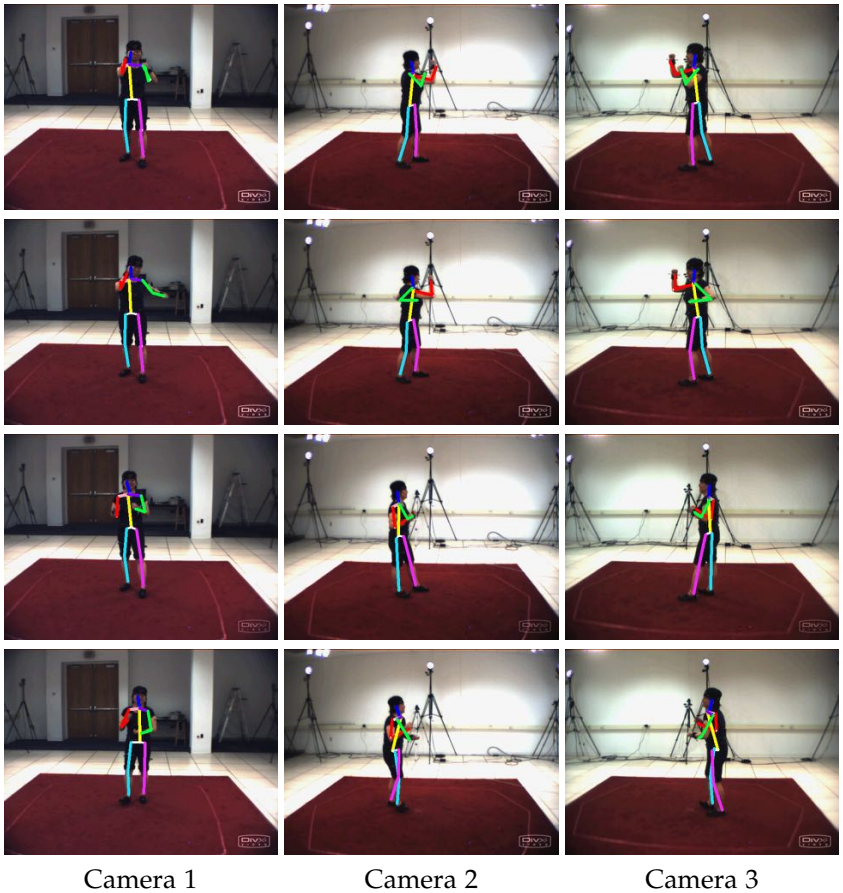
We have evaluated the part-detectors within our framework. As our final goal is to recover the poses in 3D, we evaluated the 2D part detectors by testing generated 3D samples. To that end, we have performed the inference step using only the confidence of the part detector. We empirically have fixed the number of samples to 10 for all experiments because it gave us a reasonable trade-off between accuracy and computational cost. The results are summarised in the Table B.1 and B.2. For the datasets with more that one sequence and/or more than one instance, we report the average estimate.

Dataset	Performance
KTH Football II [40]	53.8
Campus [24]	61.2
Shelf	60.5

Table B.1: **Evaluation PCP**: We have evaluated the part detectors by running our framework using only the detection confidence unary potential function. The evaluation metric is the PCP score.

Dataset	Performance
HumanEva-I [159]	130.6

Table B.2: **Evaluation 3D Error**: We have evaluated the part detectors by running our framework using only the detection confidence unary potential function. The results present the average 3D joint error in millimetres (mm).



Camera 1 Camera 2 Camera 3

Figure B.1: **HumanEva-I 1**: The 3D estimated body pose is projected across each view for the Box sequence, in different time instances.

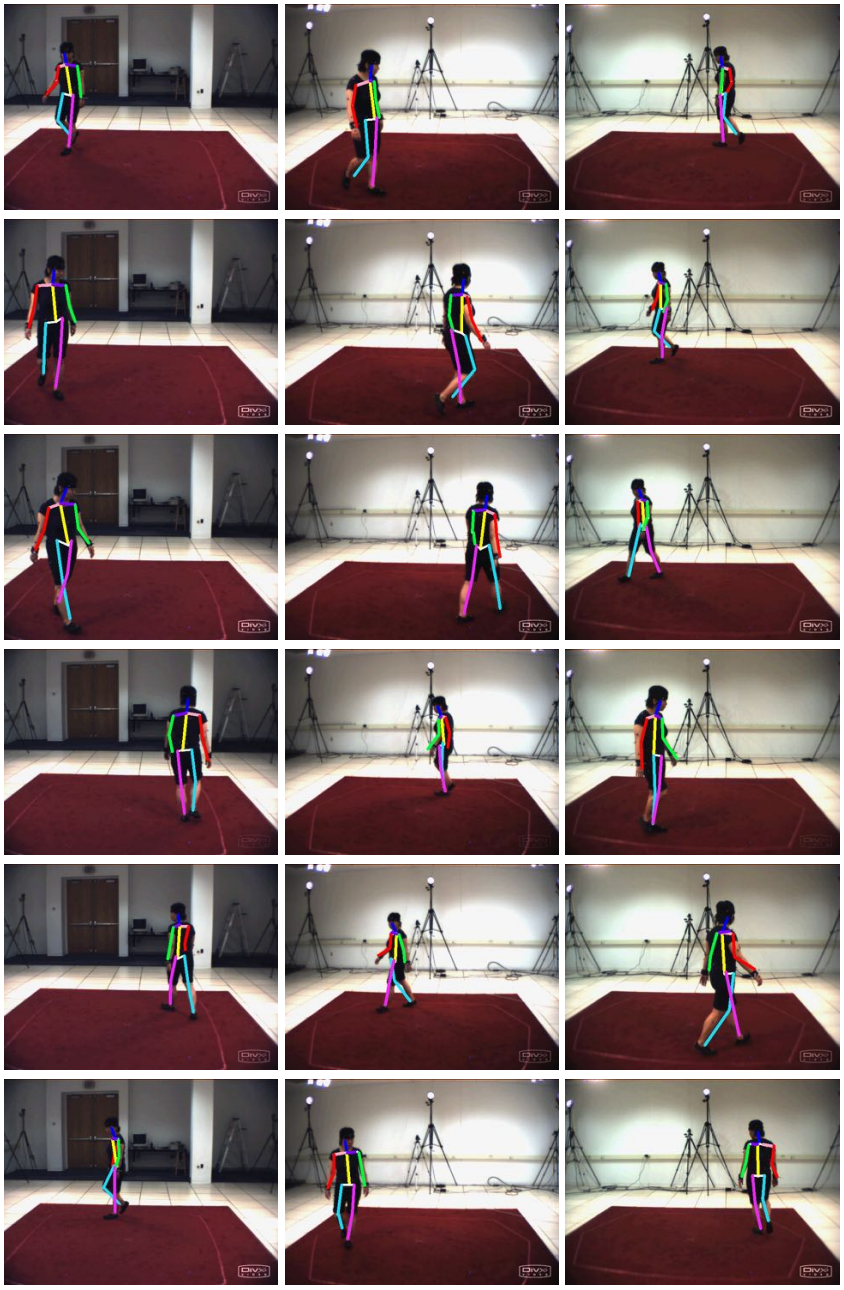


Figure B.2: **HumanEva-I 2**: The 3D estimated body pose is projected across each view for the Walking sequence, in different time instances.

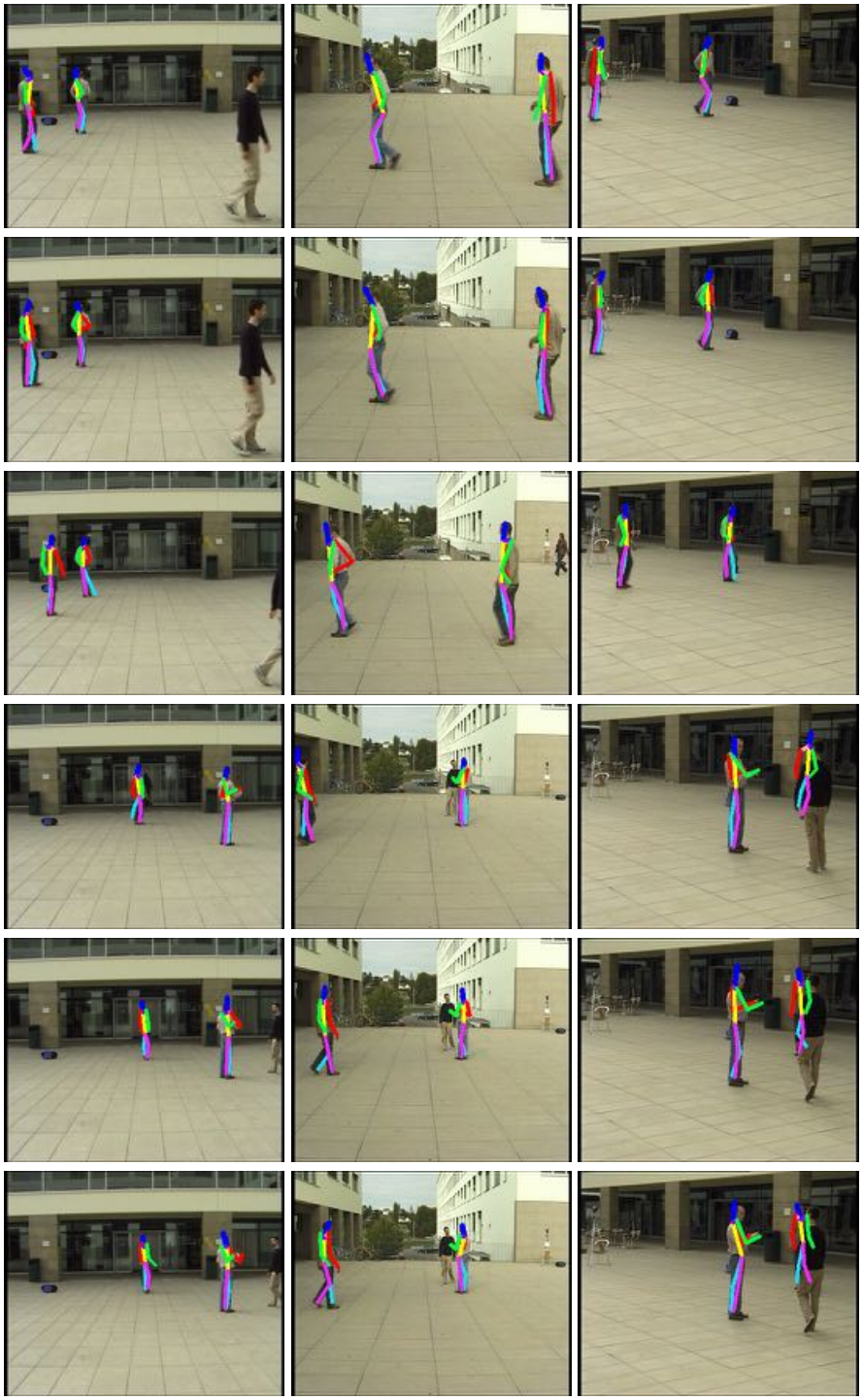


Figure B.3: **Campus 1**: The 3D estimated body poses are projected across each view, in different time instances.

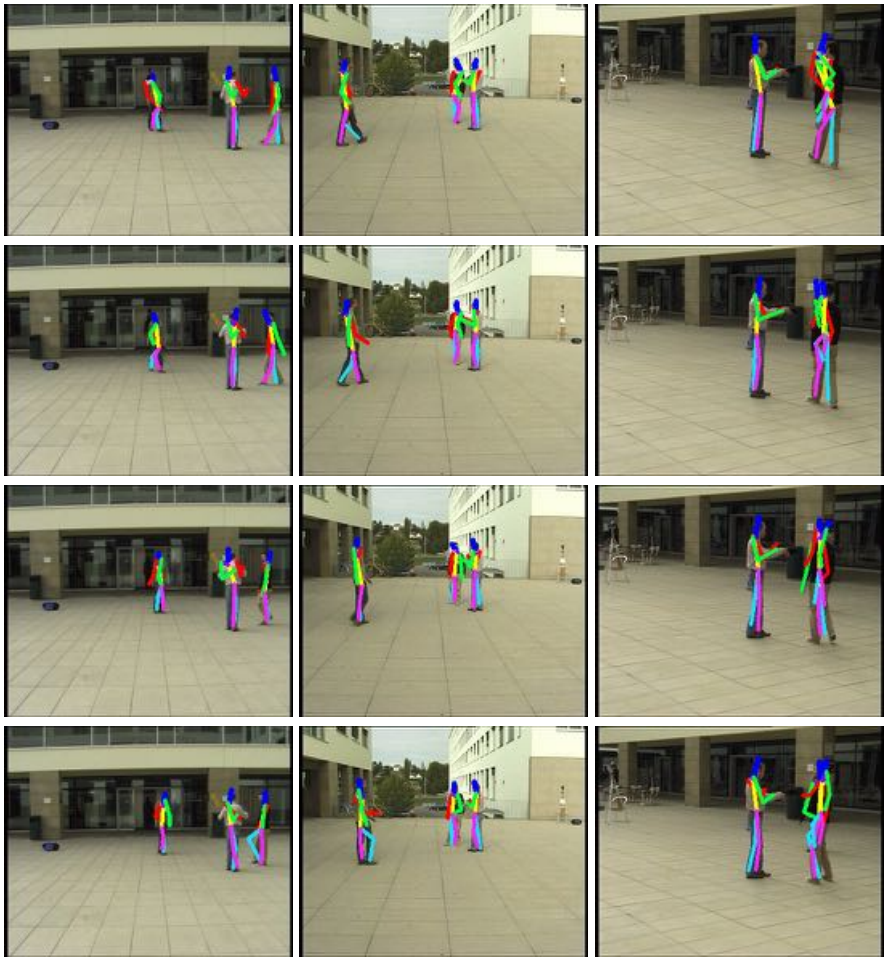


Figure B.4: **Campus 2**: The 3D estimated body poses are projected across each view, in different time instances. In Camera 3, two humans occlude each other.

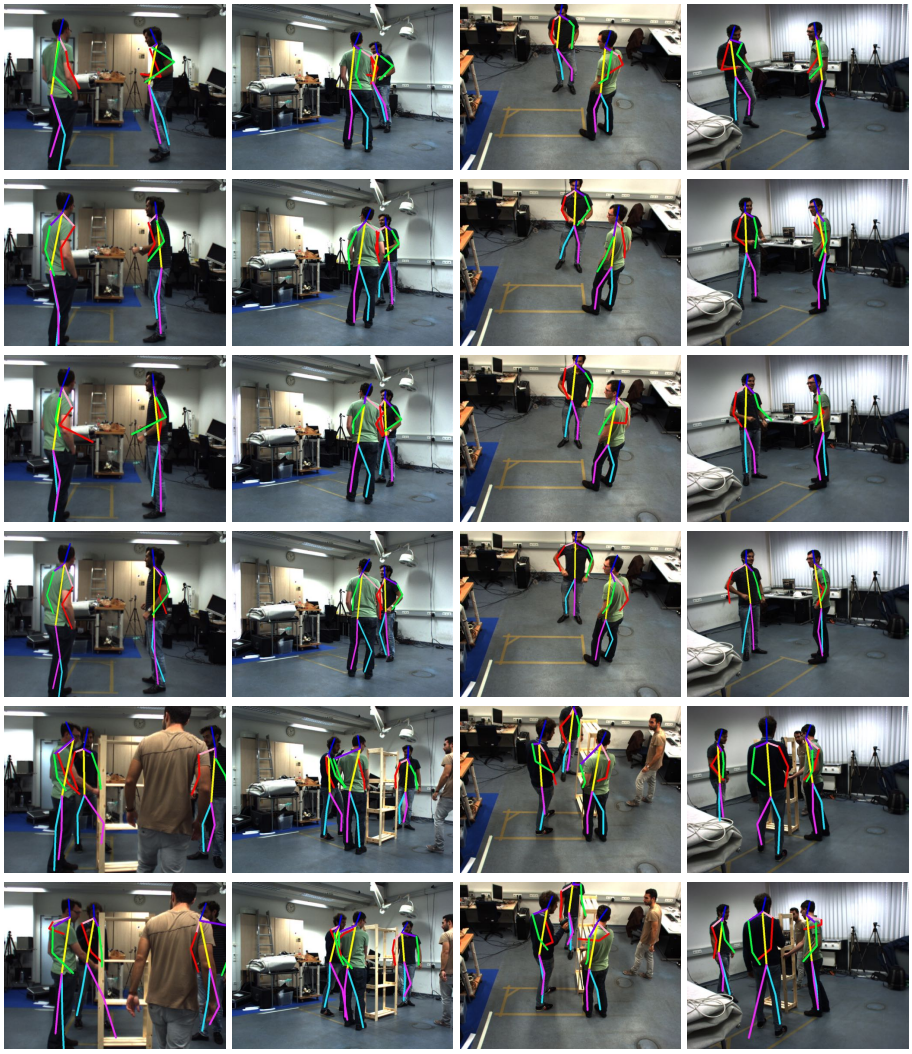


Figure B.5: **Shelf 1**: The 3D estimated body poses are projected across each view, in different time instances. On the last two rows, there is a missing human skeleton due to detection failures (high occlusion) in most of the views.

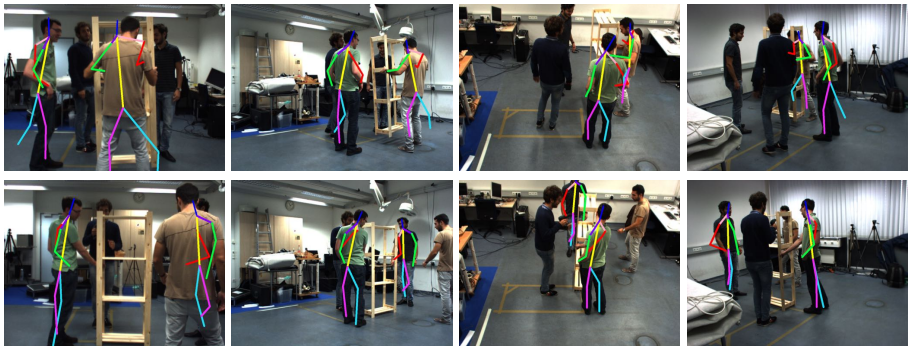


Figure B.6: **Shelf 2**: The 3D estimated body poses are projected across each view, in different time instances. There are missing human skeletons due to detection failures (high occlusion) in most of the views.

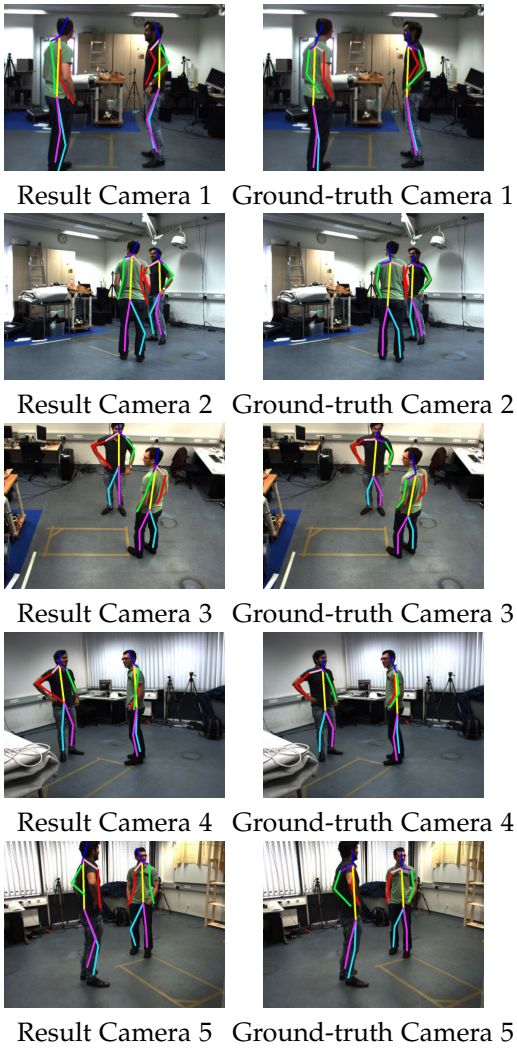


Figure B.7: **Shelf 3**: In the left column, the 3D estimated body poses are projected across each view. For better clarity, we provide the ground-truth pose separately, in the right column. The whole Shelf dataset has been annotated and will be made publicly available upon publication.

C

Human Localisation

The localization of the individuals is performed using tracking [24] in Chapters 5 and 6. We have evaluated the performance of the localization and we present it below for the datasets: Shelf (Chapter 5), Campus (Chapter 5) and OR (Chapter 6). To evaluate the people localization, we employ the *loose* PCP evaluation metric. For each individual, we define a line in the centre of the ground-truth cube which is perpendicular to the ground. It corresponds to the height of each individual in the 3D space. In the evaluation stage, we derive the same line from the inferred pose by fitting a cube and estimate the PCP score. From our experiments, we have found that the results of this mean of evaluation is equivalent to cube intersection but the computation of the PCP score is way faster. The results are summarised in Table C.1 and C.2.

	Campus		**Shelf**	
	Belagiannis et al. [16]	Our method	Belagiannis et al. [16]	Our method
Recall	98.05	99.30	90.50	97.82

Table C.1: **Human Localization Results**:The localization recall is estimated using the PCP score for the Campus and Shelf datasets. Note that the localization in [16] is done using a human detector [57] that is refined on the 3D inferred body poses.

	OR
	Our method
Recall	99.80

Table C.2: **Human Localization Results 2**:The localization recall is estimated using the PCP score for the OR dataset.

D

Authored and Co-authored Publications

Authored:

1. Belagiannis, V., Rupprecht, C., Carneiro, G., Navab, N.: Robust optimization for deep regression. In: Computer Vision (ICCV), 2015 IEEE International Conference on. IEEE (2015)

2. Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures revisited: Multiple human pose estimation. Pattern Analysis and Machine Intelligence, IEEE Transactions on (revised)

3. Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures for multiple human pose estimation. In: CVPR 2014-IEEE International Conference on Computer Vision and Pattern Recognition (2014) (**Oral Presentation**)

4. Belagiannis, V., Wang, X., Schiele, B., Fua, P., Ilic, S., Navab, N.: Multiple human pose estimation with temporally consistent 3D pictorial structures. In: Computer Vision–ECCV 2014, ChaLearn Looking at People Workshop. Springer (2014)

5. Belagiannis, V., Amann, C., Navab, N., Ilic, S.: Holistic human pose estimation with regression forests. In: Articulated Motion and Deformable Objects, pp. 20–30. Springer (2014)

6. Belagiannis, V., Schubert, F., Navab, N., Ilic, S.: Segmentation based particle filtering for real-time 2d object tracking. In: Computer Vision–ECCV 2012, pp. 842–855. Springer (2012)

Co-authored:

1. Rieke, N., Tan, D.J., Alsheakhali, M., Tombari, F., di San Filippo, C.A., Belagiannis, V., Eslami, A., Navab, N.: Surgical tool tracking with pose

estimation in retinal microsurgery. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. Springer (2015)

2. Schubert, F., Belagiannis, V., Casaburo, D.: Revisiting robust visual tracking using pixel-wise posteriors. In: International Conference on Computer Vision Systems (ICVS) (2015)

3. Baur, C., Milletari, F., Belagiannis, V., Navab, N., Fallavollita, P.: Automatic 3D reconstruction of electrophysiology catheters from two-view monoplane c-arm image sequences. In: International Journal of Computer Assisted Radiology and Surgery (IJCARS) (2015)

4. Nissler, C., Mouriki, N., Castellini, C., Belagiannis, V., Navab, N.: Omg: Introducing optical myography as a new human machine interface for hand amputees. In: International Conference on Rehabilitation Robotics - ICORR 2015. IEEE/RAS-EMBS (2015)

5. Wang, L., Belagiannis, V., Marr, C., Theis, F., Yang, G.Z., Navab, N.: Anatomic-landmark detection using graphical context modelling. In: Biomedical Imaging (ISBI), 2015 IEEE International Symposium on (2015)

6. Yigitsoy, M., Belagiannis, V., Djurka, A., Katouzian, A., Ilic, S., Pernus, F., Eslami, A., Navab, N.: Random ferns for multiple target tracking in microscopic retina image sequences. In: Biomedical Imaging (ISBI), 2015 IEEE International Symposium on (2015)

7. Milletari, F., Belagiannis, V., Navab, N., Fallavollita, P.: Fully automatic catheter localization in c-arm images using $\ell 1$- sparse coding. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014. Springer (2014)

8. Stauder, R., Belagiannis, V., Schwarz, L., Bigdelou, A., Soehngen, E., Ilic, S., Navab, N.: A user-centered and workflow-aware unified display for the operating room. In: MICCAI Workshop on Modeling and Monitoring of Computer Assisted Interventions (M2CAI), (2012)

Bibliography

[1] A.Dosovitskiy, J.T.Springenberg, T.Brox: Learning to generate chairs with convolutional neural networks. Tech. rep., arXiv:1411.5928 (2014). URL `http://lmb.informatik.uni-freiburg.de//Publications/2014/DB14a`

[2] Agarwal, A., Triggs, B.: Recovering 3d human pose from monocular images. Pattern Analysis and Machine Intelligence, IEEE Transactions on **28**(1), 44–58 (2006)

[3] Ahmadi, S.A., Sielhorst, T., Stauder, R., Horn, M., Feussner, H., Navab, N.: Recovery of surgical workflow without explicit models. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006, pp. 420–428. Springer (2006)

[4] Alahari, K., Seguin, G., Sivic, J., Laptev, I.: Pose estimation and segmentation of people in 3d movies. In: Computer Vision (ICCV), 2013 IEEE International Conference on, pp. 2112–2119. IEEE (2013)

[5] Amin, S., Andriluka, M., Rohrbach, M., Schiele, B.: Multi-view pictorial structures for 3d human pose estimation. In: British Machine Vision Conference, vol. 2 (2013)

[6] Amin, S., Müller, P., Bulling, A., Andriluka, M.: Test-time adaptation for 3d human pose estimation. In: German Conference on Pattern Recognition (GCPR/DAGM). Münster, Germany (2014)

[7] Andriluka, M., Roth, S., Schiele, B.: People-tracking-by-detection and people-detection-by-tracking. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)

[8] Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: CVPR, pp. 1014–1021. IEEE (2009)

[9] Andriluka, M., Roth, S., Schiele, B.: Monocular 3d pose estimation and tracking by detection. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pp. 623–630. IEEE (2010)

[10] Andriluka, M., Roth, S., Schiele, B.: Discriminative appearance models for pictorial structures. IJCV **99**(3), 259–280 (2012)

[11] Avidan, S.: Ensemble tracking. In: CVPR (2005)

[12] Avidan, S.: Ensemble tracking. Pattern Analysis and Machine Intelligence, IEEE Transactions on **29**(2), 261–271 (2007)

[13] Babenko, B., Yang, M., Belongie, S.: Visual tracking with online multiple instance learning. In: CVPR (2009)

[14] Baur, C., Milletari, F., Belagiannis, V., Navab, N., Fallavollita, P.: Automatic 3D reconstruction of electrophysiology catheters from two-view monoplane c-arm image sequences. In: International Journal of Computer Assisted Radiology and Surgery (IJCARS) (2015)

[15] Belagiannis, V., Amann, C., Navab, N., Ilic, S.: Holistic human pose estimation with regression forests. In: Articulated Motion and Deformable Objects, pp. 20–30. Springer (2014)

[16] Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures for multiple human pose estimation. In: CVPR 2014-IEEE International Conference on Computer Vision and Pattern Recognition (2014)

[17] Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3D pictorial structures revisited: Multiple human pose estimation. Pattern Analysis and Machine Intelligence, IEEE Transactions on (revised)

[18] Belagiannis, V., Rupprecht, C., Carneiro, G., Navab, N.: Robust optimization for deep regression. In: Computer Vision (ICCV), 2015 IEEE International Conference on. IEEE (2015)

[19] Belagiannis, V., Schubert, F., Navab, N., Ilic, S.: Segmentation based particle filtering for real-time 2d object tracking. In: Computer Vision–ECCV 2012, pp. 842–855. Springer (2012)

[20] Belagiannis, V., Wang, X., Beny Ben Shitrit, H., Hashimoto, K., Stauder, R., Aoki, Y., Kranzfelder, M., Schneider, A., Fua, P., Ilic, S., Feussner, H., Navab, N.: Parsing human skeletons in the operating room. Machine Vision and Applications (submitted)

[21] Belagiannis, V., Wang, X., Schiele, B., Fua, P., Ilic, S., Navab, N.: Multiple human pose estimation with temporally consistent 3D pictorial structures. In: Computer Vision–ECCV 2014, ChaLearn Looking at People Workshop. Springer (2014)

[22] Bengio, Y.: Learning deep architectures for ai. Foundations and trends® in Machine Learning **2**(1), 1–127 (2009)

[23] Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning, pp. 41–48. ACM (2009)

[24] Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. Pattern Analysis and Machine Intelligence, IEEE Transactions on **33**(9), 1806–1819 (2011)

[25] Bergtholdt, M., Kappes, J., Schmidt, S., Schnörr, C.: A study of parts-based object class detection using complete graphs. International journal of computer vision **87**(1-2), 93–117 (2010)

[26] Bertsekas, D., Scientific, A.: Convex optimization algorithms. Athena Scientific, United States (2015)

[27] Bertsekas, D.P.: Nonlinear programming. Athena scientific Belmont (1999)

[28] Bertsekas, D.P.: Convex optimization theory. Athena Scientific Belmont, MA (2009)

[29] Bibby, C., Reid, I.: Robust real-time visual tracking using pixel-wise posteriors. ECCV (2008)

[30] Bishop, C.M., et al.: Pattern recognition and machine learning, vol. 1. Springer New York (2006)

[31] Black, M.J., Rangarajan, A.: On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. International Journal of Computer Vision **19**(1), 57–91 (1996)

[32] Blaschko, M.B., Lampert, C.H.: Learning to localize objects with structured output regression. In: Computer Vision–ECCV 2008, pp. 2–15. Springer (2008)

[33] Bomze, I.M., Budinich, M., Pardalos, P.M., Pelillo, M.: The maximum clique problem. In: Handbook of combinatorial optimization, pp. 1–74. Springer (1999)

[34] Bosch, A., Zisserman, A., Muoz, X.: Image classification using random forests and ferns. In: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, pp. 1–8. IEEE (2007)

[35] Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010, pp. 177–186. Springer (2010)

[36] Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: ICCV, pp. 1365–1372. IEEE (2009)

[37] Bregler, C., Malik, J.: Tracking people with twists and exponential maps. In: Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on, pp. 8–15. IEEE (1998)

[38] Breiman, L.: Random forests. Machine learning **45**(1), 5–32 (2001)

[39] Breitenstein, M., Reichlin, F., Leibe, B., Koller-Meier, E., Van Gool, L.: Online multiperson tracking-by-detection from a single, uncalibrated camera. IEEE Trans on PAMI (2011)

[40] Burenius, M., Sullivan, J., Carlsson, S.: 3D pictorial structures for multiple view articulated pose estimation. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pp. 3618–3625. IEEE (2013)

[41] Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: Delving deep into convolutional nets. In: British Machine Vision Conference (2014)

[42] Chen, X., Yuille, A.L.: Articulated pose estimation by a graphical model with image dependent pairwise relations. In: Advances in Neural Information Processing Systems, pp. 1736–1744 (2014)

[43] Chockalingam, P., Pradeep, N., Birchfield, S.: Adaptive fragments-based tracking of non-rigid objects using level sets. In: ICCV (2009)

[44] Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. TPAMI **24**(5), 603–619 (2002)

[45] Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20**(3), 273–297 (1995)

[46] Criminisi, A., Shotton, J.: Decision forests for computer vision and medical image analysis. Springer (2013)

[47] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 886–893. IEEE (2005)

[48] Dantone, M., Gall, J., Leistner, C., Van Gool, L.: Human pose estimation using body parts dependent joint regressors. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pp. 3041–3048. IEEE (2013)

[49] Deutscher, J., Reid, I.: Articulated body motion capture by stochastic search. International Journal of Computer Vision **61**(2), 185–205 (2005)

[50] Doucet, A., De Freitas, N., Gordon, N.: Sequential Monte Carlo methods in practice. Springer Verlag (2001)

[51] Duffner, S., Garcia, C.: Pixeltrack: a fast adaptive algorithm for tracking non-rigid objects. In: Computer Vision (ICCV), 2013 IEEE International Conference on, pp. 2480–2487. IEEE (2013)

[52] Eichner, M., Ferrari, V.: Better appearance models for pictorial structures. In: British Machine Vision Conference, BMVC 2009, London, UK, September 7-10, 2009. Proceedings, pp. 1–11 (2009). doi: 10.5244/C.23.3. URL `http://dx.doi.org/10.5244/C.23.3`

[53] Eichner, M., Ferrari, V.: We are family: Joint pose estimation of multiple persons. In: Computer Vision–ECCV 2010, pp. 228–242. Springer (2010)

[54] Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in Neural Information Processing Systems, pp. 2366–2374 (2014)

[55] Elhayek, A., Stoll, C., Kim, K.I., Theobalt, C.: Outdoor human motion capture by simultaneous optimization of pose and camera parameters. In: Computer Graphics Forum. Wiley Online Library (2014)

[56] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV (2010)

[57] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. Pattern Analysis and Machine Intelligence, IEEE Transactions on **32**(9), 1627–1645 (2010)

[58] Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient matching of pictorial structures. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on, vol. 2, pp. 66–73. IEEE (2000)

[59] Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. International Journal of Computer Vision **61**(1), 55–79 (2005)

[60] Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)

[61] Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)

[62] Finley, T., Joachims, T.: Training structural svms when exact inference is intractable. In: Proceedings of the 25th international conference on Machine learning, pp. 304–311. ACM (2008)

[63] Fischler, M.A., Elschlager, R.A.: The representation and matching of pictorial structures. IEEE Transactions on Computers **22**(1), 67–92 (1973)

[64] Forney Jr, G.D.: The viterbi algorithm. Proceedings of the IEEE **61**(3), 268–278 (1973)

[65] Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological cybernetics **36**(4), 193–202 (1980)

[66] Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: CVPR (2009)

[67] Gall, J., Rosenhahn, B., Brox, T., Seidel, H.P.: Optimization and filtering for human motion capture. International journal of computer vision **87**(1-2), 75–92 (2010)

[68] Gavrila, D.M.: The visual analysis of human movement: A survey. Computer vision and image understanding **73**(1), 82–98 (1999)

[69] Gavrila, D.M.: A bayesian, exemplar-based approach to hierarchical shape matching. Pattern Analysis and Machine Intelligence, IEEE Transactions on **29**(8), 1408–1421 (2007)

[70] Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: Computer Vision, 2009 IEEE 12th International Conference on, pp. 221–228. IEEE (2009)

[71] Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 580–587. IEEE (2014)

[72] Girshick, R., Shotton, J., Kohli, P., Criminisi, A., Fitzgibbon, A.: Efficient regression of general-activity human poses from depth images. In: ICCV, pp. 415–422. IEEE (2011)

[73] Gkioxari, G., Hariharan, B., Girshick, R., Malik, J.: R-cnns for pose estimation and action detection. arXiv preprint arXiv:1406.5212 (2014)

[74] Godec, M., Roth, P., Bischof, H.: Hough-based tracking of non-rigid objects. In: ICCV (2011)

[75] Grabner, H., Leistner, C., Bischof, H.: Semi-supervised on-line boosting for robust tracking. ECCV (2008)

[76] Grauman, K., Shakhnarovich, G., Darrell, T.: Inferring 3d structure with a statistical image-based shape model. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, pp. 641–647. IEEE (2003)

[77] Hare, S., Saffari, A., Torr, P.H.: Struck: Structured output tracking with kernels. In: Computer Vision (ICCV), 2011 IEEE International Conference on, pp. 263–270. IEEE (2011)

[78] Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Computer Vision–ECCV 2014, pp. 297–312. Springer International Publishing (2014)

[79] Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)

[80] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., Tibshirani, R.: The elements of statistical learning, vol. 2. Springer (2009)

[81] Hofmann, M., Gavrila, D.M.: Multi-view 3d human pose estimation in complex environment. International journal of computer vision **96**(1), 103–124 (2012)

[82] Hubel, D.H., Wiesel, T.N.: Receptive fields and functional architecture of monkey striate cortex. The Journal of physiology **195**(1), 215–243 (1968)

[83] Huber, P.J.: Robust statistics. Springer (2011)

[84] Ikizler, N., Forsyth, D.: Searching video for complex activities with finite state models. In: CVPR (2007)

[85] Ionescu, C., Li, F., Sminchisescu, C.: Latent structured models for human pose estimation. In: ICCV, pp. 2220–2227. IEEE (2011)

[86] Isard, M., Blake, A.: Condensation-conditional density propagation for visual tracking. IJCV (1998)

[87] Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. arXiv preprint arXiv:1412.1842 (2014)

[88] Javed, O., Ali, S., Shah, M.: Online detection and classification of moving objects using progressively improving detectors. In: CVPR (2005)

[89] Jhuang, H., Gall, J., Zuffi, S., Schmid, C., Black, M.J.: Towards understanding action recognition. In: Computer Vision (ICCV), 2013 IEEE International Conference on, pp. 3192–3199. IEEE (2013)

[90] Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on, pp. 1822–1829. IEEE (2012)

[91] Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: Proceedings of the British Machine Vision Conference (2010). Doi:10.5244/C.24.12

[92] Johnson, S., Everingham, M.: Learning effective human pose estimation from inaccurate annotation. In: CVPR, pp. 1465–1472. IEEE (2011)

[93] Kalal, Z., Matas, J., Mikolajczyk, K.: Pn learning: Bootstrapping binary classifiers by structural constraints. In: CVPR (2010)

[94] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1725–1732. IEEE (2014)

[95] Kazemi, V., Burenius, M., Azizpour, H., Sullivan, J.: Multi-view body part recognition with random forests. In: 2013 24th British Machine Vision Conference, BMVC 2013; Bristol; United Kingdom; 9 September 2013 through 13 September 2013. British Machine Vision Association (2013)

[96] Kelley Jr, J.E.: The cutting-plane method for solving convex programs. Journal of the Society for Industrial & Applied Mathematics **8**(4), 703–712 (1960)

[97] Kiefel, M., Gehler, P.V.: Human pose estimation with fields of parts. In: Computer Vision–ECCV 2014, pp. 331–346. Springer (2014)

[98] Koller, D., Friedman, N.: Probabilistic graphical models: principles and techniques. MIT press (2009)

[99] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)

[100] Kullback, S., Leibler, R.A.: On information and sufficiency. The annals of mathematical statistics pp. 79–86 (1951)

[101] Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: Advances in Neural Information Processing Systems, pp. 1189–1197 (2010)

[102] Kwon, J., Lee, K.: Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling. In: CVPR (2009)

[103] Ladicky, L., Russell, C., Kohli, P., Torr, P.H.: Associative hierarchical crfs for object class image segmentation. In: Computer Vision, 2009 IEEE 12th International Conference on, pp. 739–746. IEEE (2009)

[104] Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001, pp. 282–289 (2001)

[105] Lallemand, J., Pauly, O., Schwarz, L., Tan, D., Ilic, S.: Multi-task forest for human pose estimation in depth images. In: 3DTV-Conference, 2013 International Conference on, pp. 271–278. IEEE (2013)

[106] Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society. Series B (Methodological) pp. 157–224 (1988)

[107] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation **1**(4), 541–551 (1989)

[108] LeCun, Y., Bottou, L., Orr, G.B., Muller, K.R.: Neural networks-tricks of the trade. Springer Lecture Notes in Computer Sciences **1524**, 5–50 (1998)

[109] Lee, M.W., Nevatia, R.: Human pose tracking using multi-level structured models. In: Computer Vision–ECCV 2006, pp. 368–381. Springer (2006)

[110] Leibe, B., Schindler, K., Van Gool, L.: Coupled detection and trajectory estimation for multi-object tracking. In: ICCV (2007)

[111] Li, S., Chan, A.B.: 3d human pose estimation from monocular images with deep convolutional neural network. In: Asian Conference on Computer Vision–ACCV 2014 (2014)

[112] Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., Hengel, A.V.D.: A survey of appearance models in visual object tracking. ACM Transactions on Intelligent Systems and Technology (TIST) **4**(4), 58 (2013)

[113] Liu, B., Huang, J., Yang, L., Kulikowsk, C.: Robust tracking using local sparse appearance model and k-selection. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 1313–1320. IEEE (2011)

[114] Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical programming **45**(1-3), 503–528 (1989)

[115] Lockheed-Martin: Ucf lockheed-martin uav dataset. `http://vision.eecs.ucf.edu/aerial/index.html` (2009)

[116] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on. IEEE (2015)

[117] Lu, W., Okuma, K., Little, J.: Tracking and recognizing actions of multiple hockey players using the boosted particle filter. Image and Vision Computing (2009)

[118] Lucas, B., Kanade, T.: with an application to stereo vision. Proceedings DARPA Image Understanding Workrhop (1998)

[119] Lucchi, A., Li, Y., Smith, K., Fua, P.: Structured image segmentation using kernelized features. In: Computer Vision–ECCV 2012, pp. 400–413. Springer (2012)

[120] Milletari, F., Belagiannis, V., Navab, N., Fallavollita, P.: Fully automatic catheter localization in c-arm images using $\ell 1$- sparse coding. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014. Springer (2014)

[121] Mitchelson, J.R., Hilton, A.: Simultaneous pose estimation of multiple people using multiple-view cues with hierarchical sampling. In: BMVC, pp. 1–10 (2003)

[122] Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Computer vision and image understanding **104**(2), 90–126 (2006)

[123] Moeslund, T.B., Hilton, A., Krüger, V., Sigal, L.: Visual Analysis of Humans. Springer (2011)

[124] Moore, D.S., McCabe, G.P.: Introduction to the Practice of Statistics. WH Freeman/Times Books/Henry Holt & Co (1989)

[125] Mori, G., Malik, J.: Estimating human body configurations using shape context matching. In: ECCV, pp. 666–680. Springer (2002)

[126] Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 807–814 (2010)

[127] Ng, A., Jordan, M.I.: On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. Advances in Neural Information Processing Systems **14**, 841 (2002)

[128] Nissler, C., Mouriki, N., Castellini, C., Belagiannis, V., Navab, N.: Omg: Introducing optical myography as a new human machine interface for hand amputees. In: International Conference on Rehabilitation Robotics - ICORR 2015. IEEE/RAS-EMBS (2015)

[129] Nowozin, S., Gehler, P.V., Lampert, C.H.: On parameter learning in crf-based approaches to object class image segmentation. In: Computer Vision–ECCV 2010, pp. 98–111. Springer (2010)

[130] Nowozin, S., Lampert, C.H.: Structured learning and prediction in computer vision. Foundations and Trends® in Computer Graphics and Vision **6**(3–4), 185–365 (2011)

[131] Nummiaro, K., Koller-Meier, E., Van Gool, L.: An adaptive color-based particle filter. Image and Vision Computing (2003)

[132] Okuma, K., Taleghani, A., Freitas, N., Little, J., Lowe, D.: A boosted particle filter: Multitarget detection and tracking. ECCV (2004)

[133] Ollero, A., Lacroix, S., Merino, L., Gancet, J., Wiklund, J., Remuss, V., Perez, I., Gutierrez, L., Viegas, D., Benitez, M., et al.: Multiple eyes in the skies: architecture and perception issues in the comets unmanned air vehicles project. Robotics & Automation Magazine, IEEE (2005)

[134] Ouyang, W., Chu, X., Wang, X.: Multi-source deep learning for human pose estimation. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 2337–2344. IEEE (2014)

[135] Padoy, N., Blum, T., Ahmadi, S.A., Feussner, H., Berger, M.O., Navab, N.: Statistical modeling and recognition of surgical workflow. Medical Image Analysis **16**(3), 632–641 (2012)

[136] Padoy, N., Blum, T., Feussner, H., Berger, M.O., Navab, N.: On-line recognition of surgical activity for monitoring in the operating room. In: AAAI, pp. 1718–1724 (2008)

[137] Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann (2014)

[138] Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: ICCV (2009)

[139] Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. ECCV (2002)

[140] Pfister, T., Simonyan, K., Charles, J., Zisserman, A.: Deep convolutional neural networks for efficient pose estimation in gesture videos. In: Asian Conference on Computer Vision (2014)

[141] Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Poselet conditioned pictorial structures. In: CVPR, pp. 588–595. IEEE (2013)

[142] Pishchulin, L., Jain, A., Andriluka, M., Thormahlen, T., Schiele, B.: Articulated people detection and pose estimation: Reshaping the future. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 3178–3185. IEEE (2012)

[143] Plankers, R., Fua, P.: Articulated soft objects for multi-view shape and motion capture. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(10) (2003)

[144] Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)

[145] Rieke, N., Tan, D.J., Alsheakhali, M., Tombari, F., di San Filippo, C.A., Belagiannis, V., Eslami, A., Navab, N.: Surgical tool tracking with pose estimation in retinal microsurgery. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. Springer (2015)

[146] Rogez, G., Rihan, J., Ramalingam, S., Orrite, C., Torr, P.H.: Randomized trees for human pose detection. In: CVPR, pp. 1–8. IEEE (2008)

[147] Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (TOG) (2004)

[148] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Cognitive modeling **5** (1988)

[149] Sapp, B., Toshev, A., Taskar, B.: Cascaded models for articulated pose estimation. In: Computer Vision–ECCV 2010, pp. 406–420. Springer (2010)

[150] Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pp. 1–8. IEEE (2007)

[151] Schmidhuber, J.: Deep learning in neural networks: An overview. Neural Networks **61**, 85–117 (2015)

[152] Schubert, F., Belagiannis, V., Casaburo, D.: Revisiting robust visual tracking using pixel-wise posteriors. In: International Conference on Computer Vision Systems (ICVS) (2015)

[153] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)

[154] Shahed Nejhum, S., Ho, J., Yang, M.: Visual tracking with histograms and articulating blocks. In: CVPR (2008)

[155] Shakhnarovich, G., Viola, P., Darrell, T.: Fast pose estimation with parameter-sensitive hashing. In: ICCV, pp. 750–757. IEEE (2003)

[156] Shimony, S.E.: Finding maps for belief networks is np-hard. Artificial Intelligence **68**(2), 399–410 (1994)

[157] Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Communications of the ACM **56**(1), 116–124 (2013)

[158] Sidenbladh, H., Black, M.J., Fleet, D.J.: Stochastic tracking of 3d human figures using 2d image motion. In: Computer Visionâ€"ECCV 2000, pp. 702–718. Springer (2000)

[159] Sigal, L., Balan, A.O., Black, M.J.: Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. International journal of computer vision **87**(1-2), 4–27 (2010)

[160] Sigal, L., Black, M.J.: Guest editorial: state of the art in image-and video-based human pose and motion estimation. International Journal of Computer Vision **87**(1), 1–3 (2010)

[161] Sigal, L., Isard, M., Haussecker, H., Black, M.J.: Loose-limbed people: Estimating 3d human pose and motion using non-parametric belief propagation. International journal of computer vision **98**(1), 15–48 (2012)

[162] Sminchisescu, C., Kanaujia, A., Li, Z., Metaxas, D.: Discriminative density propagation for 3d human motion estimation. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 390–397. IEEE (2005)

[163] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research **15**(1), 1929–1958 (2014)

[164] Stauder, R., Belagiannis, V., Schwarz, L., Bigdelou, A., Soehngen, E., Ilic, S., Navab, N.: A user-centered and workflow-aware unified display for the operating room. In: MICCAI Workshop on Modeling and Monitoring of Computer Assisted Interventions (M2CAI), (2012)

[165] Stoica, P., Moses, R.: Introduction to spectral analysis, vol. 51. Prentice Hall Upper Saddle River, NJ (1997)

[166] Sudderth, E.B., Ihler, A.T., Isard, M., Freeman, W.T., Willsky, A.S.: Nonparametric belief propagation. Communications of the ACM **53**(10), 95–103 (2010)

[167] Sun, D., Roth, S., Lewis, J., Black, M.J.: Learning optical flow. In: Computer Vision–ECCV 2008, pp. 83–97. Springer (2008)

[168] Sun, M., Savarese, S.: Articulated part-based model for joint object detection and pose estimation. In: ICCV, pp. 723–730. IEEE (2011)

[169] Sun, Y., Wang, X., Tang, X.: Deep convolutional network cascade for facial point detection. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pp. 3476–3483. IEEE (2013)

[170] Sutton, C., McCallum, A.: An introduction to conditional random fields. Machine Learning **4**(4), 267–373 (2011)

[171] Szegedy, C., Toshev, A., Erhan, D.: Deep neural networks for object detection. In: Advances in Neural Information Processing Systems, pp. 2553–2561 (2013)

[172] Taylor, G.W., Sigal, L., Fleet, D.J., Hinton, G.E.: Dynamical binary latent variable models for 3d human pose tracking. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pp. 631–638. IEEE (2010)

[173] Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in Neural Information Processing Systems, pp. 1799–1807 (2014)

[174] Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1653–1660. IEEE (2014)

[175] Tsai, D., Flagg, M., Rehg, J.: Motion coherent tracking with multi-label mrf optimization. Algorithms (2010)

[176] Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: Proceedings of the twenty-first international conference on Machine learning, p. 104. ACM (2004)

[177] Urtasun, R., Darrell, T.: Sparse probabilistic regression for activity-independent human pose inference. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)

[178] Vapnik, V.N., Vapnik, V.: Statistical learning theory, vol. 1. Wiley New York (1998)

[179] Vedaldi, A., Lenc, K.: Matconvnet – convolutional neural networks for matlab. CoRR **abs/1412.4564** (2014)

[180] Venables, W.N., Ripley, B.D.: Modern applied statistics with S. Springer Science & Business Media (2002)

[181] Wagner, D., Langlotz, T., Schmalstieg, D.: Robust and unobtrusive marker tracking on mobile phones. In: ISMAR (2008)

[182] Wang, L., Belagiannis, V., Marr, C., Theis, F., Yang, G.Z., Navab, N.: Anatomic-landmark detection using graphical context modelling. In: Biomedical Imaging (ISBI), 2015 IEEE International Symposium on (2015)

[183] Wang, X., Zhang, L., Lin, L., Liang, Z., Zuo, W.: Deep joint task learning for generic object extraction. In: Advances in Neural Information Processing Systems, pp. 523–531 (2014)

[184] Wang, Y., Tran, D., Liao, Z.: Learning hierarchical poselets for human parsing. In: CVPR, pp. 1705–1712. IEEE (2011)

[185] Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: Computer vision and pattern recognition (CVPR), 2013 IEEE Conference on, pp. 2411–2418. IEEE (2013)

[186] Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 1385–1392. IEEE (2011)

[187] Yang, Y., Ramanan, D.: Articulated human detection with flexible mixtures of parts. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(12), 2878–2890 (2013)

[188] Yao, A., Gall, J., Gool, L.V., Urtasun, R.: Learning probabilistic non-linear latent variable models for tracking complex activities. In: Advances in Neural Information Processing Systems, pp. 1359–1367 (2011)

[189] Yigitsoy, M., Belagiannis, V., Djurka, A., Katouzian, A., Ilic, S., Pernus, F., Eslami, A., Navab, N.: Random ferns for multiple target tracking in microscopic retina image sequences. In: Biomedical Imaging (ISBI), 2015 IEEE International Symposium on (2015)

[190] Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. Acm computing surveys (CSUR) **38**(4), 13 (2006)

[191] Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based r-cnns for fine-grained category detection. In: Computer Vision–ECCV 2014, pp. 834–849. Springer (2014)

[192] Zhang, Z., Luo, P., Loy, C.C., Tang, X.: Facial landmark detection by deep multi-task learning. In: Computer Vision–ECCV 2014, pp. 94–108. Springer (2014)

[193] Zhao, T., Nevatia, R.: Tracking multiple humans in complex situations. Pattern Analysis and Machine Intelligence, IEEE Transactions on **26**(9), 1208–1221 (2004)

[194] Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2879–2886. IEEE (2012)

[195] Zuffi, S., Freifeld, O., Black, M.J.: From pictorial structures to deformable structures. In: CVPR, pp. 3546–3553. IEEE (2012)

Index