

Über Baustellen Lernen – Wissensgewinn aus Luftbildern

Katrin Jahr

Lehrstuhl für Computergestützte Modellierung und Simulation · Technische Universität München ·
Arcisstraße 21 · 80333 München
E-Mail: katrin.jahr@tum.de

Baustellenplanung basiert sowohl auf explizitem Wissen, das in Vorschriften und Richtlinien vorliegt, als auf implizitem Wissen, das sich aus der Erfahrung des Planers ergibt. Um Regeln aus implizitem Wissen zu formalisieren, können abgeschlossene Bauprojekte analysiert werden. In diesem Beitrag stellen wir eine Bildanalyse-Pipeline vor, mit der Informationen über vergangene Baustellen aus Luftbildern erlangt werden können. Wir verbinden Machine-Learning-basierte Bildanalyse mit Georeferenzierung und freiverfügbaren Geodaten, um eine detaillierte Beschreibung und tatsächliche Abmessungen der jeweiligen Baustelle zu erhalten.

1 Einleitung

Trotz der fortschreitenden Digitalisierung im Bauwesen wird die Baustelleneinrichtung (BE) noch immer manuell, weitgehend auf Grundlage von Faustregeln und der Erfahrung der Planer, geplant. Der Einsatz wissenschaftlich erprobter digitaler Unterstützungsinstrumente kann erhebliches Verbesserungspotenzial in Bezug auf Planungsgenauigkeit und Zeiteffizienz bieten. Um eine effiziente BE-Planung zu ermöglichen, wurden in früheren Forschungsarbeiten mehrere halbautomatische Lösungen vorgeschlagen. Ein wichtiger Punkt bei der Formulierung dieser Planungsalgorithmen ist das umfassende Fachwissen über die Domäne.

Bei der BE-Planung muss sowohl explizites als auch implizites Wissen genutzt werden. Explizites Wissen kann aus Vorschriften, Richtlinien und lokalen Randbedingungen gewonnen werden. Allerdings sind in der Praxis häufig zahlreiche verschiedene Lösungen für jedes Bauvorhaben möglich. Um die Planer bei der Entscheidung für eine bestimmte Variante zu unterstützen, kann implizites Wissen verwendet werden. Implizites Wissen entsteht aus Expertenwissen und praktischer Erfahrung und ist schwer auf eine strenge, logische und maschinenlesbare Weise zu formulieren, kann jedoch durch die Analyse von vergangenen Bauprojekten gewonnen werden. Daher schlagen wir vor, Informationen zu sammeln, indem wir eine große Anzahl von Luftbildern mit Baustellen analysieren.

In diesem Beitrag stellen wir eine Bildanalyse-Pipeline vor, mit deren Hilfe Informationen über vergangene Baustellen aus Luftbildern erlangt werden können. Wir kombinieren Bildanalyse, Georeferenzierung und externe Datenquellen, um eine detaillierte Beschreibung der jeweiligen Baustelle zu erhalten. Im ersten Teil des Beitrags geben wir einen Überblick über den Stand der Technik in der Baustellenplanung mit Schwerpunkt Bildanalyse, wie sie heute auf Baustellen eingesetzt wird. Anschließend erläutern wir Schritt für Schritt die in der Pipeline verwendeten Methoden. Im dritten Teil des Papiers stellen wir eine Fallstudie vor. Schließlich folgt eine Zusammenfassung der Ergebnisse.

2 Stand der Technik

Bildanalysemethoden haben durch die jüngsten Fortschritte im Bereich des autonomen Fahrens und des maschinellen Lernens viel Aufmerksamkeit erregt. Für eine effektive und effiziente Bildanalyse und Objekterkennung wurden und werden in den letzten Jahrzehnten zunehmend maschinelle Lernalgorithmen eingesetzt. Im Jahr 2012 erreichte das Convolutional Neural Network (CNN) „AlexNet“ (Krizhevsky, Sutskever und Hinton, 2012) einen Top-5-Fehler von 15,3% bei der renommierten ImageNet Large Scale Visual Recognition Challenge (Russakovsky et al., 2015). Diese Ergebnisse waren überraschend genau und zeigten die Vorteile der Verwendung von CNN. Aus diesem Grund konzentriert sich die Softwareindustrie momentan auf CNNs für Machine-Learning-basierte Bildverarbeitungsaufgaben (LeCun, Bengio und Hinton, 2015).

Im Bausektor ist Bildanalyse ein ziemlich neues Thema. Da einer der Hauptaspekte des maschinellen Lernens die Erfassung großer Datensätze ist, konzentrieren sich die aktuellen Ansätze auf die Datenerfassung. Tajeen und Zhu (2014) präsentieren einen Bilddatensatz mit zahlreichen kommentierten Bildern von Baumaschinen, wobei der Schwerpunkt auf Baumaschinen der Aushubphase liegt. Die analysierten Bilder wurden vom Boden aufgenommen. Kropp, Koch und König (2018) erkennen Innenbauteile wie Heizkörper anhand von Ähnlichkeiten. Im Rahmen der automatisierten Baufortschrittskontrolle veröffentlichten Han et al. einen Ansatz zur Amazon Turk-basierten Kennzeichnung (Han und Golparvar-Fard, 2017). Bügler et al. (2017) kombinierten photogrammetrische Verfahren und Videoanalyse, um den Fortschritt der Erdarbeiten zu bewerten. Zu diesem Zweck erstellten sie Punktwolken, um das Volumen des ausgehobenen Bodens zu messen, und erkannten LKW-Kipper auf Bildern durch Foreground Detection. Jahr, Braun und Borrmann (2018) verwenden künstliche Intelligenz zur Erkennung von Schalungselementen auf Luftbildern von Baustellen, die aus UAVs stammen.

3 Methodik

Um implizites Wissen aus Luftbildern zu verallgemeinern, schlagen wir eine Bildanalysepipeline vor, die maschinelle Lernalgorithmen, Georeferenzierung und die Abfrage zusätzlicher Daten nutzt (siehe Abbildung 1). Für die Georeferenzierung benötigt jedes Luftbild zusätzliche Vermessungsinformationen. Im ersten Schritt verwenden wir einen Objekterkennungsalgorithmus, um Baustellen auf den Luftbildern zu erkennen. Wird mindestens eine Baustelle erkannt, wird ein Algorithmus zu Instanzsegmentierung verwendet, um einzelne Elemente der Baustelle zu erkennen. In diesem Beitrag konzentrieren wir uns auf die Erkennung von Turmdrehkränen, da diese einen besonders starken Einfluss auf den Baufortschritt haben. Um die Elementabmessungen abzuschätzen, werden die Bilder orthorektifiziert. Zusätzliche baustellenrelevante Informationen wie Gebäudemaße, Grundstückslinien oder angrenzende Gebäude können durch Georeferenzierung des Bildes und dessen Verknüpfung mit räumlichen Informationen aus der Katasterkarte oder anderen Geoinformationsdiensten, wie OpenStreetMap oder Stadtmodellen, abgerufen werden.

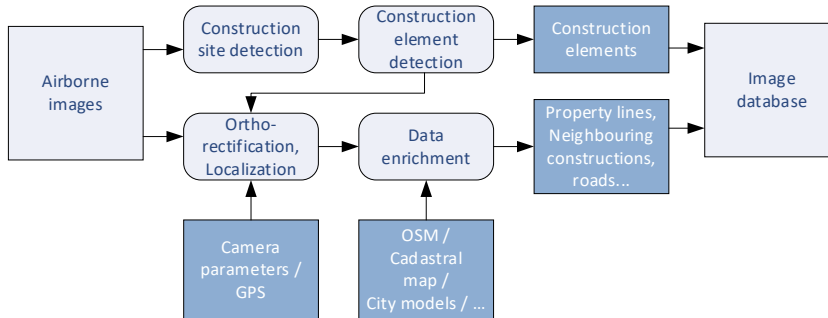


Abbildung 1: Bildanalyse-Pipeline: Im ersten Schritt werden Luftbilder nach Baustellen durchsucht. Auf diesen Baustellen werden einzelne BE-Gegenstände erkannt. Um die korrekten Abmessungen zu erhalten, werden die Bilder orthorektifiziert und georeferenziert. Zusätzliche Datenquellen werden verwendet, um Kontext hinzuzufügen.

3.1 Luftbildanalyse mit Convolutional Neural Networks

Um Informationen über Baustellen zu sammeln, müssen diese zunächst im Bild lokalisiert werden. Es sollen also Objekte auf Bildern lokalisiert und klassifiziert werden, wobei mehrere Objekte vorhanden sein können („Objekterkennung“) (Rusk, 2015). Dazu sagen Algorithmen in der Regel rechteckige Flächen (Bounding Boxes) vorher, innerhalb derer sich mit hoher Wahrscheinlichkeit ein Objekt befindet, sowie die entsprechende Objektklasse.

In unserer Pipeline verwenden wir ein "You only look once"-Netz (YOLOv3) im Darknet-Framework (Redmon & Farhadi, 2018). YOLOv3 ist ein Single-Shot-Detector, der im Vergleich zu anderen führenden Algorithmen eine akzeptable Vorhersagerate mit sehr schnellen Trainings- und Vorhersagezeiten ermöglicht. YOLOv3 teilt das Eingebild in ein Raster, wobei jede Zelle nur ein Objekt voraussagt. Vorhersagen von Objekten unterschiedlicher Größe werden durch ein Feature-Pyramiden-Netz ermöglicht – YOLOv3 erstellt Vorhersagen in drei verschiedenen Größen für jeden Bildbereich. Um die Bounding Box des erkannten Objekts vorherzusagen, verwendet YOLOv3 Ankerboxen, die auf die spezifische Problem-domäne zugeschnitten sind.

Für die BE Gegenstände sollen genaue Informationen über den Ort und die Abmessungen ermittelt werden. Daher muss jeder Gegenstand so genau wie möglich erfasst werden – nicht nur seine Bounding Box. Dazu verwenden wir einen Instanzsegmentierungsalgorithmus, der Bilder pixelweise untersucht, und nicht nur zwischen verschiedenen Klassen, sondern auch zwischen mehreren Objekten derselben Klasse unterscheiden kann.

Ein sehr leistungsfähiger Algorithmus ist Mask R-CNN (He, Gkioxari, Dollár und Girshick, 2017). Mask R-CNN prognostiziert Instanzmasken für erkannte Objekte in zwei Schritten: Zunächst verwendet es ein RoI (Region of Interest) Align-Netz, um Bounding Boxen und Klassen für mögliche Objekte zu finden. Danach wird ein semantisches Segmentierungsmodell verwendet, um die genauen Objektkonturen innerhalb des Begrenzungsrahmens zu bestimmen. Da in jedem Begrenzungsrahmen nur ein Objekt enthalten sein soll, genügt ein binärer Klassifikator, der die Pixel 1/0 abbildet, wobei „1“ Anwesenheit, „0“ das Fehlen eines Objekts repräsentiert.

3.2 Georeferenzierung und Orthorektifizierung

Um Objekte in Luftbildern messen und global lokalisieren zu können, müssen die Bilder georeferenziert und orthorektifiziert werden. Zur Georeferenzierung können Bodenkontrollpunkte verwendet werden, oder die Position der Kamera über die GNSS-Position (Global Navigation Satellite System) berechnet werden. Die Genauigkeit der Georeferenzierung hängt von der Qualität des GNSS-Signals ab und kann von wenigen Dezimetern bis zu wenigen Metern variieren (Pfeifer, Glira und Briese, 2012).

Um die realen Dimensionen von Objekten in Luftbildern abzuschätzen, muss der Abstand zwischen zwei Pixeln am Boden (Ground Sample Distance, GSD) bekannt sein. In der traditionellen Luftphotogrammetrie werden Bilder in Nadirsicht aufgenommen. Unter der Annahme einer lokal flachen Erdoberfläche und unter Kenntnis von Flughöhe, Sensorgröße und Kamerakonstante (Brennweite) kann die GSD angenähert werden. Änderungen der Geländehöhe oder störende Objekte (z.B. hohe Gebäude) führen dabei zu abweichenden GSD.

Bei der Orthorektifizierung werden Luftbilder in der Zentralprojektion in eine orthogonale Projektion umgewandelt, um die GSD zu vereinheitlichen und direkte Messungen in den Bildern zu ermöglichen. Für jede Zelle des DSM-Netzes wird der entsprechende Teil des Bildes identifiziert und auf das DSM transformiert. Das resultierende Netz wird dann auf ein regelmäßiges Gitter ortho-projiziert, das auf der Referenzfläche mit definierter GSD erstellt wird.

3.3 Verknüpfung mit zusätzlichen Datenquellen

Die Umgebung der Baustelle hat einen erheblichen Einfluss auf die Baustellenausstattung. So beeinflussen benachbarte Gebäude die Position und Mindesthöhe von Kranen; Zufahrtsstraßen beeinflussen die Lage von Baustraßen und Lagerflächen. Wurde der Standort einer Baustelle durch die Georeferenzierung des Bildes bestimmt, kann diese Baustelle mit räumlichen Informationen über die umliegenden Einrichtungen angereichert werden.

Geodaten sind aus verschiedenen Quellen verfügbar. Eine große Auswahl an digitalen Geodaten steht auf OpenStreetMap (OSM) zur Verfügung. Auf OSM werden Daten unter offener Lizenz gesammelt und bereit gestellt, unter anderem Straßen, Parks, Gebäudeumrisse und Einrichtungen wie Feuerlöschhydranten und Briefkästen. Der Datenzugriff erfolgt über eine Kartenansicht oder verschiedene APIs. In diesem Papier haben wir die Overpass API mit Overpass QL verwendet, um Informationen über benachbarte Gebäude und Straßen abzurufen.

4 Fallstudie

4.1 Trainieren der CNNs

Damit die verwendeten CNNs Baustellen und Baustelleneinrichtung erkennen können, müssen sie zunächst trainiert werden. Dafür wird eine große Menge von Beispieldaten (Ground Truth) verwendet, welche jeweils aus einem Bild und der dazugehörigen, erwarteten Ausgabe bestehen.

Für die Ground Truth beider CNNs verwenden wir Luftbilder des Deutschen Zentrums für Luft- und Raumfahrt (DLR) (Kurz et al., 2012). Die Bilder wurden in Deutschland aus

Flugzeugen und Hubschraubern aufgenommen. Insgesamt stehen ca. 4.500 hochauflösende Bilder zur Verfügung. Im ersten Schritt haben wir alle Bilder, die Baustellen enthalten, ausgewählt (siehe Abbildung 2). Um Daten für die Objekterkennung zu generieren, wurden anschließend Bounding Boxes für alle Baustellen hinzugefügt und die Daten in Trainings-, Test- und Validierungsdatensätze aufgeteilt (70%, 20%, 10%).

Für die Segmentierung werden diese Bilder entlang der Baustellengrenze zugeschnitten. Um den Datensatz zu diversifizieren, nutzen wir zusätzlich Drohnen-Fotos und Bodenaufnahmen von Turmdrehkränen (Abbildung 3). Alle Kranumrisse werden pixelgetreu mit polygonalen Labels markiert. Auch in diesem Fall teilen wir die Daten in 70% Trainingsdaten, 20% Testdaten und 10% Validierungsdaten auf.



Abbildung 2: Luftbildaufnahme, zugeschnitten auf Baustelle



Abbildung 3: Bodenaufnahme und UAV-Aufnahme von Turmdrehkränen

4.1.1 Baustellenerkennung mit YOLOv3

Für die Objekterkennung verwenden wir eine YOLOv3-Architektur in Darknet (Redmon and Farhadi, 2018). Um das Netz besser an den Baustellendatensatz anzupassen, haben wir die Ankergrößen auf die Baustellengrößen angepasst, wobei Länge und Breite der Ankerboxen mit Länge und Breite der Bounding Boxes, normiert auf die jeweiligen Bildmaße, korrelieren. Um 9 Anker zu erhalten, die den Datensatz möglichst gut abbilden, haben wir ein k-means Clustering auf den Maßen der Bounding Boxes verwendet.

Um die Trainingszeit zu verkürzen und mit unserem verhältnismäßig kleinen Datensatz bessere Ergebnisse zu erzielen, verwenden wir vortrainierten Gewichte von Darknet53. Das Training für 1000 Epochen dauerte ca. 10 Stunden auf einer Nvidia DGX-1. Abbildung 4 zeigt Beispiele für resultierende Vorhersagen. Während Begrenzungsrahmen für kleinere Baustellen sehr gut vorhergesagt werden, ist das CNN für größere Baustellen noch nicht ausreichend angepasst. Bei der Betrachtung des Datensatzes wird deutlich, dass kleinere Baustellen in Wohngebieten dominieren und einen Großteil des Datensatzes ausmachen. Für eine größere Vielfalt von Baustellen wird der Trainingsdatensatz derzeit noch erweitert. Zur weiteren Verbesserung der Objekterkennung könnten die Daten vorverarbeitet werden.



Abbildung 4: Beispiele für resultierende Bounding Boxes in der Baustellenerkennung

4.1.2 Segmentierung von Turmdrehkränen mit Mask R-CNN.

Für die Bildsegmentierung verwenden wir Mask R-CNN in Keras mit TensorFlow Backend (He et al., 2017). Wir verwenden vortrainierte Gewichte aus dem COCO-Datensatz (Lin et al., 2014). Beispiele für die vorhergesagten Masken und Bounding Boxes zeigt Abbildung 5. Die Bounding Boxes werden für die meisten Krane zuverlässig vorhergesagt, die Masken sind leicht lückenhaft. Einige Krane im Randbereich von Bildern werden übersehen.



Abbildung 5: Beispiele für resultierende Bounding Boxes und Masken in der Kransegmentierung.

4.2 Georeferenzierung

Um die Bilder zu georeferenzieren, generieren wir zunächst ein DSM und berechnen das Orthophoto. Bereiche, die von zwei Bildern abgedeckt werden, werden dabei unterbrochen dargestellt, was es schwierig macht, Turmdrehkrane direkt im Orthophoto zu erkennen. Daher wurden die Krane in den Originalbildern markiert und die Label auf das DSM orthoprojiziert.

4.3 Datenanreicherung mit Overpass API

Um Objekte in direkter Umgebung der betreffenden Baustelle von OSM abzurufen, verwenden wir Overpass API. Anhand der GPS-Koordinaten der Außenecken des Gebäudes haben wir angrenzende Straßen und Gehwege sowie benachbarte Gebäude abgefragt. Overpass API gibt die abgefragten Daten textbasiert (z.B. als json oder xml) zurück. Zur visuellen Prüfung der Ergebnisse haben wir Overpass turbo verwendet (Abbildung 6). Overpass turbo ist eine webbasierte Anbindung an Overpass API. Die Ergebnisse werden als interaktive Karte dargestellt, auf der weitere Informationen zu den abgefragten Knoten abgerufen werden können. Die von OSM gesammelten zusätzlichen Daten werden zu den aus den Bildern gewonnenen Informationen hinzugefügt.

```

4 <meta osm_base="2019-04-12T11:05:02Z"/>
5 <way id="26775855"/>
6   <nd ref="293694475"/>
7   <nd ref="293694476"/>
8   <nd ref="293694477"/>
9   <nd ref="2368200281"/>
10  <nd ref="293694475"/>
11  <tag k="addr:city" v="Braunschweig"/>
12  <tag k="addr:country" v="DE"/>
13  <tag k="addr:housenumber" v="35"/>
14  <tag k="addr:postcode" v="38106"/>
15  <tag k="addr:street" v="Franz-Liszt-Straße"/>
16  <tag k="building" v="university"/>
17  <tag k="building:levels" v="3"/>
18  <tag k="operator" v="TU Braunschweig"/>
19 </way>
20 <way id="26776014">
21   <nd ref="429587899"/>
22   <nd ref="293696352"/>
23   <tag k="access" v="permissive"/>

```



Abbildung 6: Angrenzende Objekte (Fußwege, Straßen, Gebäude) der Baustelle aus Overpass API. Links: XML, rechts: Visualisierung von Overpass turbo, gefundene Objekte sind in blau markiert.

5 Zusammenfassung

In diesem Beitrag stellen wir eine Bildanalysepipeline vor, die sowohl Baustellen als auch BE-Gegenstände auf Luftbildern erkennen kann. Um weitere Informationen über die Situation vor Ort zu erhalten, ermitteln wir die realen Abmessungen des Baufeldes und der Turmdrehkrane durch Orthorektifizierung der Bilder. Zur Anreicherung der Informationen können weitere Datenquellen genutzt werden. Mit Hilfe der Overpass API erhalten wir über OpenStreetMap Informationen über die Umgebung der Baustelle. Nächste Schritte sind die Erweiterung der Algorithmen zur Erkennung weiterer Baustellenelemente wie Container oder Fahrzeuge, sowie die Anbindung an Stadtmodelle zur Gewinnung von 3D-Informationen.

Literatur

- Bügler, M., Borrmann, A., Ogunmakin, G., Vela, P. A., & Teizer, J. (2017). Fusion of Photogrammetry and Video Analysis for Productivity Assessment of Earthwork Processes. *Computer-Aided Civil and Infrastructure Engineering*, 32(2), 107–123. <https://doi.org/10.1111/mice.12235>
- Han, K. K., & Golparvar-Fard, M. (2017). Potential of big visual data and building information modeling for construction performance analytics: An exploratory study.

- Automation in Construction, 73, 184–198.
<https://doi.org/10.1016/j.autcon.2016.11.004>
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2961–2969.
- Jahr, K., Braun, A., & Borrmann, A. (2018). Formwork detection in UAV pictures of construction sites. Proc. of the 12th European Conference on Product and Process Modelling, Copenhagen, Denmark.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 1097–1105.
- Kropp, C., Koch, C., & König, M. (2018). Interior construction state recognition with 4D BIM registered image sequences. Automation in Construction, 86, 11–32.
<https://doi.org/10.1016/J.AUTCON.2017.10.027>
- Kurz, F., Meynberg, O., Rosenbaum, D., Türmer, S., Reinartz, P., & Schroeder, M. (2012). Low-cost optical camera system for disaster monitoring. Int. Archives of the Photogrammetry, Remote Sens. and Spatial Information Sci, 39, B8.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444.
<https://doi.org/10.1038/nature14539>
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. European Conference on Computer Vision, 740–755.
- Pfeifer, N., Glira, P., & Briese, C. (2012). Direct georeferencing with on board navigation components of light weight UAV platforms. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 39(B7), 487–492.
- Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. ArXiv Preprint ArXiv:1804.02767.
- Rusk, N. (2015). Deep learning. Nature Methods, Vol. 13, p. 35.
<https://doi.org/10.1038/nmeth.3707>
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>
- Tajeen, H., & Zhu, Z. (2014). Image dataset development for measuring construction equipment recognition performance. Automation in Construction, 48, 1–10.
<https://doi.org/10.1016/J.AUTCON.2014.07.006>