

Detektion von Fahrradständern in Luftbildern mittels Deep Learning

**KARIN ERBE¹, MELANIE BRANDMEIER², MICHAEL SCHMITT³, ANDREAS DONAUBAUER¹,
JAN-ANDREAS LIEBSCHER⁴ & THOMAS H. KOLBE¹**

Zusammenfassung: Dieser Beitrag erörtert anhand der Detektion und Klassifikation von unüberdachten Fahrradständern aus Luftbildern und Oberflächenmodellen, inwiefern Deep Learning eingesetzt werden kann, um Stadtmöbel automatisch zu erfassen und für 3D-Stadtmodelle nutzbar zu machen. Dafür wird am Beispiel der Fahrradständer der Landeshauptstadt München ein Trainingsdatensatz erzeugt und ein Mask R-CNN Modell trainiert. Um die Ergebnisse zu evaluieren, wird eine umfangreiche quantitative und qualitative Fehleranalyse durchgeführt. Dabei zeigte sich, dass bei der Verwendung der RGB-Bilddaten in Verbindung mit dem nDOM-Raster die besten Detektionsergebnisse erzielt werden. Die Detektion erreicht eine Genauigkeit (Precision) von 83,2% und eine Trefferquote (Recall) von 77,3%. Als Einschränkungen erweisen sich der Schattenwurf durch Gebäude und Bäume. Zudem wird deutlich, dass die Zahl der abgestellten Fahrräder einen signifikanten Einfluss auf den Erfolg der Detektion hat.

1 Motivation

3D-Stadtmodelle spielen eine wichtige Rolle bei Planungs- und Verwaltungsaufgaben. Durch fortwährende Veränderungen im Stadtbild ergibt sich der Bedarf nach einer regelmäßigen Aktualisierung der 3D-Stadtmodelle. Insbesondere Digitale Zwillinge von Städten, die versuchen die physikalische Umwelt möglichst exakt abzubilden, benötigen eine zeitnahe Synchronisation der abgebildeten Objekte. Während für Gebäude, beispielsweise mit Hilfe von Laserscanning-Befliegungen, dem kontinuierlich aktualisierten Liegenschaftskataster und darauf aufsetzender Gebäuderekonstruktion bewährte Fortführungsprozesse genutzt werden, hat sich bei der automatisierten Erfassung von Stadtmöblierung noch keine geeignete Methode durchgesetzt. Deep Learning bietet sich in diesem Feld an, da es eine effiziente Möglichkeit bietet, aus Bilddaten automatisiert Objekte zu erfassen. Deep Learning ist ein Teilbereich des maschinellen Lernens, welcher die Lehre von tiefen künstlichen neuronalen Netzen umfasst. Einmal trainiert kann ein Deep Learning Netzwerk auf ähnliche, aber neue Bilder angewendet werden. Inwiefern Deep Learning eingesetzt werden kann, um Stadtmöbel automatisch zu erfassen und für 3D-Stadtmodelle nutzbar zu machen, wird in diesem Beitrag anhand der Detektion und Klassifikation von unüberdachten Fahrradständern aus Luftbildern und Oberflächenmodellen

¹ Technische Universität München, Lehrstuhl für Geoinformatik, Arcisstraße 21, D-80333 München, E-Mail: [karin.erbe, andreas.donaubauer, thomas.kolbe]@tum.de

² Hochschule Würzburg-Schweinfurt, Röntgenring 8, D-97070 Würzburg, E-Mail: melanie.brandmeier@fhws.de

³ Universität der Bundeswehr München, Institut für Raumfahrttechnik und Weltraumnutzung, Werner-Heisenberg-Weg 39, D-85577 Neubiberg, E-Mail: michael.schmitt@unibw.de

⁴ Landeshauptstadt München - Kommunalreferat - Geodatenservice, Denisstraße 2, D-80335 München, E-Mail: jan.liebscher@muenchen.de

erörtert. Für den Aufbau des Deep Learning Frameworks und das Training werden dabei die Deep Learning Funktionalitäten von ArcGIS Pro (ESRI 2022A) und der ArcGIS API für Python (ESRI 2022B) verwendet.

2 Verwandte Arbeiten

In der Literatur finden sich bisher keine einschlägigen Vorarbeiten zur automatischen Erfassung von Fahrradständern und Straßenmöbeln ähnlicher Art, wie Bänke und Mülleimer, mittels Deep Learning aus Luftbildern. Veröffentlichungen über die Erkennung von Fahrzeugen oder Flugzeugen (z.B. CHEN 2018) sind dagegen verbreiteter. Die Fernerkundung stellt besondere Anforderungen an die Objekterkennung und die Instanzsegmentierung. Dies liegt an der besonderen Beschaffenheit der Daten, die aus der Vogelperspektive und nicht wie die meisten Bilddaten im Profil erfasst werden. Daraus ergeben sich 360°-Objektdrehungen, komplexe Hintergründe (LI et al. 2018), unterschiedliche Aufnahmebedingungen wie Wetter, Jahreszeit und Auflösung (LI et al. 2020), aber auch Verdeckungen, Beleuchtungsänderungen und Schattenwurf (SU et al. 2019). Darüber hinaus können Fernerkundungsdaten aus mehr als den drei üblichen Farbbändern Rot, Grün und Blau bestehen.

Ein Beispiel für ein 3D-Stadtmodell, in dem Stadtmobiliar enthalten ist, wurde für die Stadt Rotterdam implementiert (GEMEENTE ROTTERDAM 2022). Es enthält beispielsweise Laternen, Mülleimer, Parkautomaten und auch Fahrradständer. Ob es möglich ist Objekte dieser Art automatisch in Luftbildern mit Hilfe von Deep Learning zu detektieren, wird im Folgenden anhand von Fahrradständern in München erörtert.

3 Deep Learning-basierte Erkennung von Fahrradständern

3.1 Anwendungsgebiet & Daten

Als Untersuchungsgebiet wird die Landeshauptstadt München gewählt, die an etwa 350 Orten mehr als 34.000 öffentliche Fahrradstellplätze bereitstellt (Stand: 2019). Eine Reihe von Fahrradstellplätzen, die baulich als zusammenhängend erkennbar sind, kann als ein Fahrradständer-Objekt bezeichnet werden. Die Landeshauptstadt München führt diese Fahrradständer-Objekte als linienhafte Geoobjekte in einem Geodatensatz, welcher als Grundlage für die Erstellung der notwendigen Trainingsdaten für den Deep Learning Ansatz dient. Jedem der Fahrradständer ist dabei ein Fahrradständer-Typ zugewiesen. Dabei handelt es sich beispielsweise um Anlehnbügel, Klemmbügel oder verschiedene Typen von Vorderradhaltern, die ein Anschließen des Rahmens ermöglichen.

Als Bildgrundlage dienen RGB-Orthophotos aus einer Winterbefliegung im Jahr 2019 mit einer Bodenauflösung von 8 cm. Zusätzliche Rasterdaten werden hinzugezogen, um zu testen, inwiefern CIR (color-infrared), nDOM (normalisiertes digitales Oberflächenmodell) und NDVI (Normalized Difference Vegetation Index) die Ergebnisse verbessern können. Dabei werden folgende fünf Datenkombinationen getestet: RGB, CIR, RGB+nDOM, RGB+NDVI bzw. RGB+nDOM+NDVI.

3.2 Erstellung der Trainingsdaten

Die Erstellung der Trainingsdaten erfolgt mit dem ArcGIS-Pro-Werkzeug *Beschriften von Objekten für Deep Learning*. Dafür werden alle Linienobjekte des Fahrradständer-Datensatzes einzeln betrachtet. Jeder im Luftbild erkennbare Fahrradständer wird entsprechend seines Typs manuell mit einem Rechteck eingerahmt. Von 1.999 Fahrradständer-Objekten im Geodatensatz sind 1.463 sichtbar und als Trainingsobjekte nutzbar. Die übrigen 536 Objekte können aufgrund der Überdachung, der Verdeckung durch Bäume oder aufgrund von sehr dunklem Schatten nicht verwendet werden. Darüber hinaus enthält der Datensatz Fahrradständer, die zum Zeitpunkt der Luftbildaufnahme nicht mehr oder noch nicht aufgestellt waren. Einige der Objekte im Datensatz werden durch mehrere Rechtecke repräsentiert, sodass ein Trainingsdatensatz entsteht, der mit ca. 1.800 markierten Fahrradständer-Objekten als Grundlage für das Deep Learning dient.

Bei der Erstellung der Trainingsdatensätze werden bereits Herausforderungen dieses Deep-Learning-Ansatzes deutlich (siehe Abb. 1): Schattenwurf durch Gebäude und Bäume erschweren das Erkennen von Fahrradständern ebenso wie Äste und leere Stellplätze. Hinzu kommt, dass einige der Linienobjekte verschoben oder mit falscher Länge hinterlegt sind und somit den Markierungsprozess (*Labeling*) erschweren.

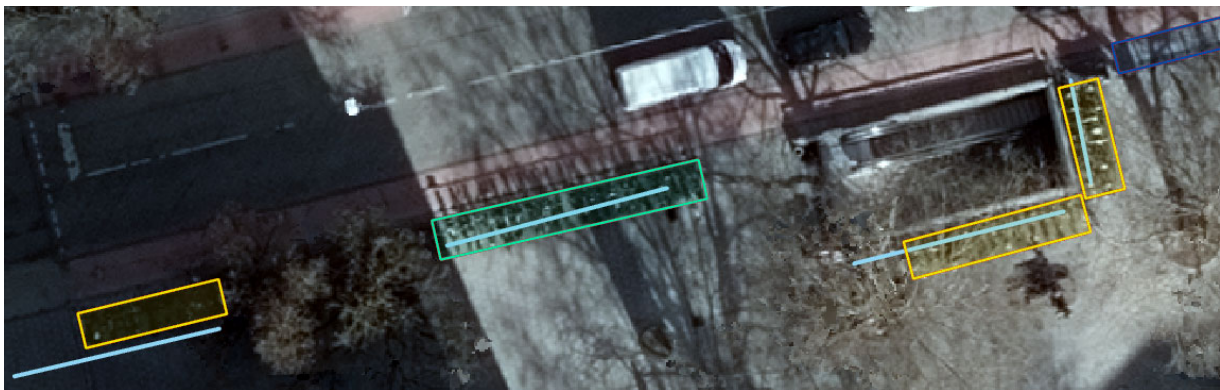


Abb. 1: Ausschnitt aus RGB-Orthophoto (Rotkreuzplatz) mit markierten Fahrradständern unterschiedlichen Typs; links: verschobene Liniengeometrie (Hellblau) und dunkler Schatten durch Gebäude, der ein Erkennen des Objekts erschwert; Mitte: Fahrradständer im Schatten eines Baumes; rechts: Äste verdecken Teile des Fahrradständers (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

Anschließend werden die Bildkacheln mit den zugehörigen Labeln für das Training exportiert. Dazu werden im selben ArcGIS-Pro-Werkzeug TIFF-Dateien der Größe 256×256 Pixel exportiert. Zu jedem Bild wird eine binäre Maske der gleichen Größe ausgegeben, welche die Position der Fahrradständer einer Klasse ausgibt. Für die gesamte Fläche Münchens ergeben sich damit ca. 7500 Bildkacheln.

3.3 Training

ArcGIS Pro stellt für die Detektion von Objekten verschiedene Deep-Learning-Frameworks zur Verfügung. Einer der Netzentwürfe, Mask R-CNN (HE et al. 2017), bietet über die Objekterkennung hinaus zusätzlich eine Instanzsegmentierung, sodass nicht nur das Objekt durch eine rechteckige Maske markiert, sondern pixelgenau lokalisiert wird. Dabei handelt es sich um die Weiterentwicklung der R-CNN Familie: Convolutional Neural Networks, die zusätzlich Region Proposal Networks implementieren, d.h. Layer, die in einem zusätzlichen Schritt dafür zuständig sind, Regionen von Interesse zu identifizieren. Mask R-CNN implementiert zwei Head Netzwerke, die parallel arbeiten: eines für die Klassifikation und die Bounding Box Regression und das andere für die exakte Berechnung der Segmentierungsmaske. Die Verlustfunktion ergibt sich dabei aus der Summe der Verlustfunktionen für Klassifikation (log loss), Bounding Box Regression (L_1 loss) und Maskenberechnung (average binary cross-entropy loss) (HE et al. 2017 & GIRSHICK 2015).

Für die Detektion der Fahrradständer-Objekte wurde ein Mask R-CNN Deep-Learning-Modell mit vortrainiertem ResNet-50 Backbone mit Hilfe der ArcGIS API für Python über 100 Epochen trainiert.

Als ein wichtiger Teil der Preprocessing-Aufgaben stellt sich die Data Augmentation, also die Erweiterung der Daten, heraus. Die Bilder im Trainingsdatensatz werden durch Drehungen, Helligkeits- und Kontraständerungen, sowie Zoomen verändert und verwendet, um den Trainingsdatensatz künstlich zu vergrößern (FAST.AI 2021). In diesem Projekt wurden 50% der Bilder um einen Winkel zwischen -30° und 30° gedreht. Diese Werte werden gewählt, um den Schattenwurf durch Bäume und Gebäude weiterhin realistisch zwischen West und Ost zu erhalten. Zudem werden 50% der Bilder horizontal an der vertikalen Bildmittelachse gespiegelt. Der Kontrast wird zufällig mit Werten zwischen 1 (gleichbleibend) und 1,5 erhöht und die Helligkeit leicht verändert (Werte zwischen 0,4 und 0,6 (0,5 ist gleichbleibend)). Zudem wird jedes Bild zufällig auf Zoomwerte zwischen 100% und 120% vergrößert. Mit Hilfe dieses Vorgangs wird das Training verbessert.

3.4 Nachbereitung

Die Anwendung der trainierten Modelle erfolgt anschließend in ArcGIS Pro mit dem Werkzeug *Objekte mit Deep Learning erkennen*. Detektierte Objekte werden durch die Segmentierungsmaske eingerahmt und durch diese in Position und Ausdehnung markiert. Im Rahmen des Projekts wurde zudem eine entsprechende Toolbox realisiert, die es Sachbearbeitern des 3D-Stadtmodells ermöglicht, auf Basis der trainierten Netze die Erkennung auf einfache und wiederholte Weise durchzuführen. Die Toolbox berechnet aus den Segmentierungsmasken (Abb. 2: Lila) Rechtecke (Abb. 2: Orange) und Linien (Abb. 2: Grün), welche nun für den gewünschten Zweck weiterverarbeitet werden können.



Abb. 2: Ausschnitt aus RGB-Orthophoto (Weißenburger Straße) mit gelabelten Fahrradständern in Rot, detektierten Fahrradständern in Lila, daraus berechneten Rechtecken in Orange und Linien in Grün (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

4 Evaluation der Ergebnisse

4.1 Vorgehen

Für die Evaluation des Trainingserfolgs müssen Bereiche verwendet werden, die einerseits nicht in den Trainingsdaten enthalten sind, andererseits aber Ground-Truth-Daten aufweisen. Dafür werden vier Gebiete aus dem Trainingsdatensatz entfernt (Abb. 3): die Gegenden um die S- und U-Bahn-Station Harras, um den Rotkreuzplatz, die U-Bahn-Station Messestadt West und das Gebiet zwischen Rosenheimer Platz und Ostbahnhof. Diese werden gewählt, da diese unterschiedliche Stadtstrukturen repräsentieren. Während Harras und Rotkreuzplatz belebte Plätze mit unterschiedlicher Baumdichte in der Stadt vertreten, liegt die Messestadt weiter außerhalb und ist mit vergleichsweise neuen Fahrradständern ausgestattet. Die Gegend nahe dem Ostbahnhof ist hingegen durch mehrstöckige Gebäude und damit viel Schatten und Hinterhöfe gekennzeichnet. Insgesamt liegen in diesen Stadtgebieten 128 gelabelte Fahrradständer.

Für jede der fünf Datenkombinationen werden alle detektierten Objekte einzeln betrachtet und in richtig positiv (RP) bzw. falsch positiv (FP) eingeordnet. Bei richtig positiven Detektionen wird dabei zusätzlich unterschieden, ob es sich um Fahrradständer handelt, welche auf Privatgrund stehen und daher nicht in den Ground-Truth-Daten enthalten sind (RP_{sonstige}), oder ob es Fahrradständer sind, die im kommunalen Datensatz enthalten sind (RP). Weitere werden ferner darin unterschieden, ob sie korrekt oder falsch klassifiziert werden. Darüber hinaus fallen in die Gruppe der RP_{sonstige} einzeln detektierte Fahrräder und öffentliche Fahrradständer, die fälschlicherweise nicht im Geodatensatz der Stadt München enthalten sind, aber korrekt detektiert werden. Falsch positive Detektionen werden anhand des fälschlich markierten Objektes gruppiert. Beispiele hierfür sind Autos, Dächer oder Vegetation. Umgekehrt werden auch die Ground-Truth-Daten betrachtet und alle nicht detektierten, also falsch negativen (FN), Objekte gezählt. Daraus ergibt sich nun folgende Tabelle (Tab. 1), die einen Vergleich der Datenkombinationen für die gewählten Stadtgebiete ermöglicht.

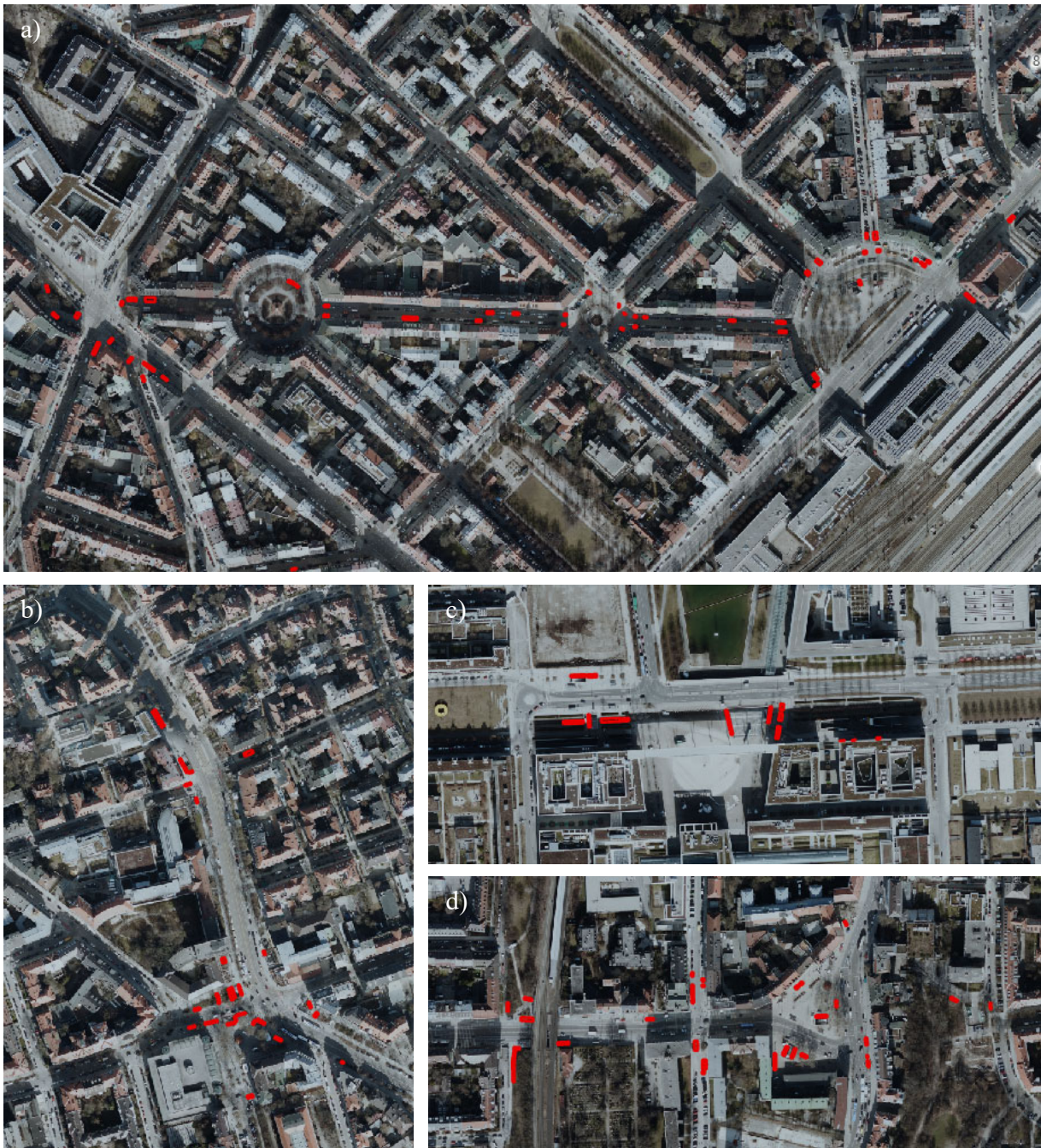


Abb. 3: Aus Trainingsdaten herausgenommene Gebiete: gelabelte Fahrradständer-Objekte in Rot; a) Rosenheimer Platz bis Orleansplatz/Ostbahnhof (50 Objekte); b) Rotkreuzplatz (31 Objekte); c) Harras (30 Objekte); d) Messestadt West (17 Objekte) (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

Tab. 1: Detaillierte Auflistung aller RP, FP und FN Detektionen in allen Datenkombinationen zusammengefasst über alle Gebiete

		RGB	Anteil an detektierten Objekten	CIR	Anteil an detektierten Objekten	+nDOM	Anteil an detektierten Objekten	+NDVI	Anteil an detektierten Objekten	+nDOM +NDVI	Anteil an detektierten Objekten
RP + RP_sonstige	korrekt detektiert	170	86,3%	187	82,4%	188	90,4%	174	89,7%	166	85,1%
	privat	71	36,0%	78	34,4%	69	33,2%	70	36,1%	71	36,4%
	korrekt klassifiziert	67	34,0%	79	34,8%	80	38,5%	77	39,7%	75	38,5%
	falsch klassifiziert	20	10,2%	17	7,5%	19	9,1%	16	8,2%	17	8,7%
	einzelne Fahrräder ungelabelt & öfftl.	11 1	5,6% 0,5%	10 3	4,4% 1,3%	17 3	8,2% 1,4%	6 5	3,1% 2,6%	1 2	0,5% 1,0%
FP	falsch detektiert	27	13,7%	40	17,6%	20	9,6%	20	10,3%	29	14,9%
	Dach	6	3,0%	5	2,2%	0	0,0%	1	0,5%	2	1,0%
	Bodenmuster	8	4,1%	7	3,1%	4	2,4%	3	1,5%	8	4,1%
	Friedhof	4	2,0%	3	1,3%	0	0,0%	2	1,0%	3	1,5%
	Auto	2	1,0%	2	0,9%	0	0,0%	0	0,0%	4	2,1%
	Busch, Baum, Gras	6	3,0%	9	4,0%	8	3,8%	4	2,1%	8	4,1%
	Schatten	0	0,0%	4	1,8%	0	0,0%	2	1,0%	1	0,5%
	rechteckiges Objekt Motorrad, o.ä.	0 1	0,0% 0,5%	3 7	1,3% 3,1%	2 5	1,0% 2,4%	4 4	2,1% 2,1%	1 2	0,5% 1,0%
Σ detektiert	197	100,0%	227	100,0%	208	100,0%	194	100,0%	195	100,0%	

	zu detektieren	128		128		128		128		128	
RP	detektiert & gelabelt	87	68,0%	96	75,0%	99	77,3%	93	72,7%	92	71,9%
	korrekt klassifiziert	67	52,3%	79	61,7%	80	62,5%	77	60,2%	75	58,6%
	falsch klassifiziert	20	15,6%	17	13,3%	19	14,8%	16	12,5%	17	13,3%
FN	gelabelt & undetektiert	41	32,0%	32	25,0%	29	22,7%	35	27,3%	36	28,1%

Aus diesen Werten können nun Genauigkeit (engl.: precision), Falscherkennungsrate (engl.: false discovery rate, FDR), Trefferquote (engl.: recall) und Falsch-negativ-Rate (engl.: false negative rate, FNR) berechnet werden (Tab. 2). Die Unterscheidung zwischen zwei RP-Werten spiegelt sich dabei in der Berechnung der Werte wider. Bei Precision₂ und FDR₂ werden die RP_{sonstige} miteinbezogen und RP₂ stellt die Summe aus RP und RP_{sonstige} dar. Da für private Fahrradständer keine Ground-Truth-Daten vorhanden sind, können hier keine FN-Werte in die Berechnung miteinfließen.

Tab. 2: Precision, FDR, Recall und FNR für alle Datenkombinationen

		RGB	CIR	+nDOM	+NDVI	+nDOM +NDVI
Precision₂	$\frac{RP_2}{RP_2 + FP}$	86,3%	82,4%	90,4%	89,7%	85,1%
FDR₂	$\frac{FP}{RP_2 + FP}$	13,7%	17,6%	9,6%	10,3%	14,9%
Recall	$\frac{RP}{RP + FN}$	68,0%	75,0%	77,3%	72,7%	71,9%
FNR	$\frac{FN}{FN + RP}$	32,0%	25,0%	22,7%	27,3%	28,1%
Precision	$\frac{RP}{RP + FP}$	76,3%	70,6%	83,2%	82,3%	76,0%
FDR	$\frac{FP}{RP + FP}$	23,7%	29,4%	16,8%	17,7%	24,0%

Wenn RP_{sonstige} einbezogen wird, erzielt RGB+nDOM mit einer Precision₂ von 90,4 % die besten Ergebnisse. CIR schneidet mit 82,4 % am schlechtesten ab. Wenn RP_{sonstige} außer Acht gelassen wird, liegen die Recall-Werte zwischen 68,0 % (RGB) und 77,3 % (RGB+nDOM). Die Precision erreicht Werte zwischen 70,6% (CIR) und 83,2% (RGB+nDOM). In allen Fällen erreicht RGB+nDOM die besten Werte für die Erkennung von Fahrradständern (Recall: 77,3%, Precision: 83,2%). Wenn es um den Anteil der RP im Vergleich zu allen erkannten Objekten (Precision) geht, schneidet CIR (70,6%) weiterhin am schlechtesten ab. Beim Recall schneidet RGB (68,0 %) am schlechtesten ab. RGB+NDVI kommt RGB+nDOM in Bezug auf die Precision am nächsten. Beim Recall ist es CIR.

4.2 Richtig positive Detektionen

In der Analyse zeigt sich, dass die Segmentierungsmasken die Fahrradständer sehr genau erfassen und einrahmen können. Abb. 4a) zeigt korrekt detektierte Fahrradständer mit den roten Rechtecken als Ground-Truth. Zu richtig positiven Detektionen zählen auch Fahrradständer, die nicht von der Stadt München verwaltet werden, da sie beispielsweise auf Privatgrund stehen (Abb. 4b).

Gelegentlich werden auch einzelne Fahrräder als Fahrradständer-Objekt erkannt und markiert (Abb. 4c).



Abb. 4: Beispiele richtig positiver Detektionen: a) korrekt markierte Fahrradständer-Objekte in Lila mit gelabelten Fahrradständern in Rot (Rosenheimer Platz); b) private Fahrradstellplätze (Spicherenstraße); c) einzelne Fahrräder angelehnt an ein Geländer (Weißenburger Platz) (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

4.3 Falsch positive Detektionen

Für fälschlicherweise detektierte Objekte sind Bildstrukturen verantwortlich, die denen eines Fahrradständer-Objektes gleichen. Abb. 5 zeigt Beispiele für die in Tab. 1 aufgeführten Objekte. Bodennahe Strukturen beispielsweise wie Büsche (Abb. 5a), Gräber auf Friedhöfen (Abb. 5b) und Schatten (Abb. 5c) sorgen für falsch positive Detektionen. Auch Gleisbetten (Abb. 5d) ähneln in ihrem Muster denen eines Fahrradständers und können vom Modell falsch erkannt werden. Diese lassen sich aber beispielsweise durch die Verwendung einer Maske eliminieren, die nur Flächen enthält, an denen Fahrradständer aufgestellt werden. Bahnstrecken können dafür ausgeschlossen werden. Falsch detektierte Dachstrukturen (Abb. 5e) können genauso wie fälschlich detektierte Autos (Abb. 5f) durch die Verwendung des nDOM-Rasters vermieden werden.

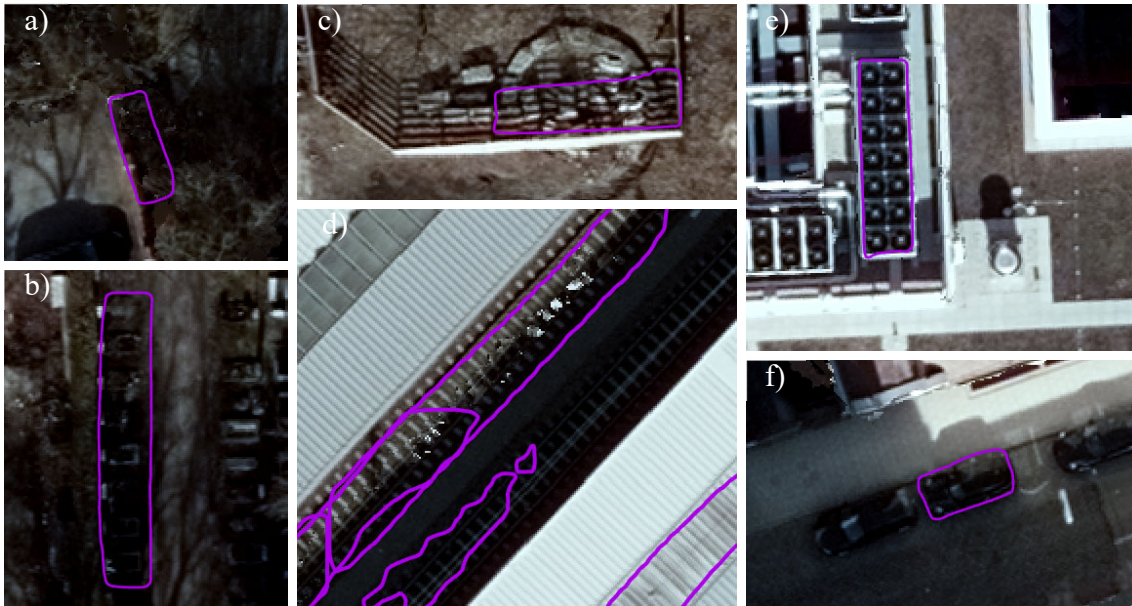


Abb. 5: Beispiele falsch positiver Detektionen: a) Busch (Kidlerplatz); b) Friedhof (Friedhof Sendling); c) Schatten (Kidlerplatz); d) Gleisbett und Dach (Ostbahnhof); e) Dach (Willy-Brandt-Platz); f) Auto (Volkartstraße) (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

4.4 Falsch negative Detektionen

Werden bei der Anwendung von Deep Learning Objekte nicht erkannt, liegt dies zum einen daran, dass nicht genug Beispiele in den Trainingsdaten für ein spezielles Pixelmuster vorliegen. Darüber hinaus bestätigt die Analyse der falsch negativ detektierten Fahrradständer die bereits bei der Trainingsdatensatzerstellung beobachteten Herausforderungen: Verdeckungen durch dichte Äste und sehr dunkle Schatten können eine Detektion verhindern (Abb. 6a, 6b und 6d). Des Weiteren zeigt sich, dass die Anzahl abgestellter Fahrräder von Bedeutung ist: sind diese chaotisch geparkt, beispielsweise weil zu viele Räder auf zu wenig Stellplätze kommen (Abb. 6c) oder ist der Fahrradständer leer (Abb. 6e), werden die Fahrradständer-Objekte unter Umständen ebenfalls nicht erkannt.



Abb. 6: Beispiele falsch negativer Detektionen (Label in Rot): a) dichte Äste (Orleansplatz); b) dunkler Schatten (Willy-Brandt-Allee); c) große Anzahl chaotisch abgestellter Fahrräder (Albert-Roßhaupter-Straße); d) Baumschatten und leerer Fahrradständer (Rotkreuzplatz); e) (fast) leerer Fahrradständer (Plinganserstraße) (Bildquelle: LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT – GEODATENSERVICE 2020)

4.5 Klassifizierung

Um neben der reinen Detektion auch eine Klassifizierung der Fahrradständertypen zu untersuchen, wird ein Trainingsdatensatz mit allen 17 enthaltenen Fahrradständertypen erstellt. Da zehn der Fahrradständertypen aber durch weniger als 50 manuell markierte Objekte repräsentiert sind, werden anschließend ähnlich aussehende Typen in einer Oberklasse gruppiert. Dadurch entstehen sieben Klassen, die für einen zweiten Trainingsdatensatz verwendet werden. Ebenso wird ein Trainingsdatensatz erstellt, bei dem keine Unterscheidung der Klassen gemacht wird.

Auf der einen Seite kommt es beim Training mit allen Klassen durch die geringe Anzahl an Trainingsbeispielen für einzelne Klassen zu Unter- und Überanpassungen. Auf der anderen Seite zeigt sich keine Verbesserung mehr, wenn keine Gruppierung vorgenommen wird, sodass sich ein Gruppieren als sinnvoll erweist.

Dabei können die Objekte mit einer Rate von 62,5% korrekt klassifiziert werden. Es zeigt sich, dass eine Unterscheidung zwischen Anlehnbügeln und Fahrradständern, bei denen das Vorderrad gehalten wird, möglich ist. Durch die begrenzte Größe des Datensatzes gibt es allerdings nicht für alle Fahrradständertypen genügend Trainingsbeispiele.

5 Diskussion

Insgesamt zeigen alle Datenkombinationen gute Ergebnisse, RGB+nDOM kann aber durch das Hinzuziehen der Objekthöhen zum Beispiel Gebäude und Autos ausschließen und die besten Ergebnisse erzielen. Vegetationssensitive Raster wie Nahinfrarot und NDVI können wenig dazu beitragen, dass Vegetationsobjekte wie Büsche und Bäume nicht fälschlicherweise detektiert werden, da es sich um unbelaubte Winterbilder handelt.

Bei der Betrachtung der Ergebnisse zeigt sich, dass der Schattenwurf insgesamt eine große Rolle spielt. Die Lichtverhältnisse zur Zeit der Bildaufnahme beeinflussen maßgeblich die Trainingsdaten. Zum einen kommt es durch die Aufnahme in Wintermonaten zu ausgeprägtem Schattenwurf durch Gebäude und Bäume, sodass 74% der Fahrradständer in den Trainingsdaten im Schatten stehen. Durchgängige Schatten verändern den Kontrast zwischen Fahrradständer-Objekten und Hintergrund, während unbelaubte Laubbäume ein Schattenmuster werfen, welches die Detektion erschweren kann. Zudem werden Fahrräder detektiert, die einen Schatten werfen, der vom Deep Learning Netzwerk als Fahrradständer interpretiert wird, da in den Trainingsdaten einige Fahrräder in Fahrradständern ebenso diese Fahrradschatten werfen.

Diese Beobachtungen zeigen, dass die Größe und der Inhalt der Trainingsdaten für diese Anwendung ausreichend sind, aber eine Erweiterung des Datensatzes von Vorteil wäre.

Auch die geringe Zahl leerer Fahrradständer in den Trainingsdaten führt dazu, dass diese seltener erkannt werden. Das Vorhandensein abgestellter Fahrräder und deren regelmäßige Anordnung in Fahrradständern sind somit für die Detektion förderlich.

Auch die Genauigkeit der Klassifizierung ist von der Menge und den Inhalten der Trainingsdaten abhängig. Insgesamt lässt sich aber die Aussage treffen, dass anhand der Anordnung der Fahrräder in den Fahrradständern (sehr geordnet oder chaotisch und mit welchem Abstand) eine erste Klassifizierung möglich ist.

6 Fazit & Ausblick

Es zeigt sich, dass bei der Verwendung der RGB-Bilddaten in Verbindung mit dem nDOM Raster die besten Detektionsergebnisse erzielt werden. Die Detektion erreicht eine Genauigkeit von 83,2% und eine Trefferquote von 77,3%. Eine Einschränkung stellt hierbei insbesondere der Schattenwurf durch Gebäude und Bäume dar. Da 74% der Fahrradständer in den Trainingsdaten im Schatten liegen, werden zum einen Fahrradständer, die in der Sonne stehen, schlechter detektiert. Zum anderen verhindern sehr dunkle Schatten ebenfalls eine Detektion. Die Analyse der detektierten Objekte zeigt zudem, dass Fahrradständer besser erkannt werden, wenn dort viele Fahrräder abgestellt sind. Der Grund ist auch hier in den Trainingsdaten zu finden, da diese nur wenige leere Fahrradständer enthalten. Eine Klassifizierung der Fahrradständertypen ist nur bedingt möglich: Aus den 17 unterschiedlichen Typen wurden zunächst sieben Kategorien eingeteilt, welche dann mit einer Rate von 62,5% korrekt klassifiziert werden können. Für eine erste grobe Einteilung der Objekte ist die Klassifizierung allerdings ausreichend.

Aus den detektierten und klassifizierten polygonförmigen Objekten können nun Informationen zur Größe und Ausrichtung der Fahrradständer abgeleitet werden, um diese im 3D-Stadtmodell durch standardisierte 3D-Modelle aus einer Modellbibliothek ersetzen zu können.

Um die Detektion und insbesondere die Klassifizierung noch weiter zu verbessern, ist eine größere Zahl von Trainingsdaten notwendig. Auch um die Wahrscheinlichkeit zu erhöhen, dass das Modell auf andere Städte angewendet werden kann, ist die Erweiterung des Trainingsdatensatzes wünschenswert. Dies adressiert die Problematik, dass im Bereich der Fernerkundung für viele Objekte bisher nicht ausreichend große Datensätze und Normen vorhanden sind, um auf passende Trainingsdaten zugreifen zu können. Dies muss durch Bereitstellung und Katalogisierung der vorhandenen Datensätze vorangetrieben werden.

7 Literaturverzeichnis

- CHEN, Z., ZHANG, T. & OUYANG, C., 2018: End-to-End Airplane Detection Using Transfer Learning in Remote Sensing Images. *Remote Sensing*, 10:139. <https://doi.org/10.3390/rs10010139>.
- ESRI, 2022A: Überblick über das Toolset „Deep Learning“. ArcGIS Pro Dokumentation. <https://pro.arcgis.com/de/pro-app/latest/tool-reference/image-analyst/an-overview-of-the-deep-learning-toolset-in-image-analyst.htm>, letzter Zugriff am 07.01.2022.
- ESRI, 2022B: ArcGIS API for Python. ArcGIS Developer. <https://developers.arcgis.com/python/>, letzter Zugriff am 07.01.2022.
- FAST.AI., 2021: vision.transform. List of transforms for data augmentation in CV. <https://fastai1.fast.ai/vision.transform.html>, letzter Zugriff am 07.01.2022.
- GEMEENTE ROTTERDAM, 2022: Rotterdam 3D. <https://www.3drotterdam.nl>, letzter Zugriff am 07.01.2022.
- GIRSHICK, R., 2015: Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV), 1440-1448, <https://doi.org/10.1109/iccv.2015.169>.
- HE, K., GKIOXARI, G., DOLLÁR, P. & GIRSHICK, R., 2017: Mask R-CNN. 2017 IEEE International Conference on Computer Vision (ICCV), 2980-2988, <https://doi.org/10.1109/iccv.2017.322>.

- LANDESHAUPTSTADT MÜNCHEN – KOMMUNALREFERAT - GEODATENSERVICE, 2020: RGB, CIR, nDOM, DOM, DGM, NDVI aus Winterbefliegung 2019.
- LI, K., CHENG, G., BU, S. & YOU, X., 2018: Rotation-Insensitive and Context-Augmented Object Detection in Remote Sensing Images. IEEE Transactions on Geoscience and Remote Sensing, **56**(4), 2337-2348, <https://doi.org/10.1109/TGRS.2017.2778300>.
- LI, K., WAN, G., CHENG, G., MENG, L. & HAN, J., 2020: Object detection in optical remote sensing images: A survey and a new benchmark. ISPRS Journal of Photogrammetry and Remote Sensing, **159**, 296-307, <https://doi.org/10.1016/j.isprsjprs.2019.11.023>.
- SU, H., WEI, S., YAN, M., WANG, C., SHI, J. & ZHANG, X., 2019: Object Detection and Instance Segmentation in Remote Sensing Imagery Based on Precise Mask R-CNN. In IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 1454-1457, <https://doi.org/10.1109/IGARSS.2019.8898573>.