Seminararbeit des Textes: A Brief Review of Recent Developments in the Integration of Deep Learning with GIS

Ricardo Altweck
(ricardo1.altweck@st.oth-regensburg.de)

Wissenschaftliches Seminar,
OTH Regensburg

WS 2022/2023 (Version vom 28. Januar 2023)

Zusammenfassung

Dieser Text behandelt die Erkenntnisse aus dem Paper **A Brief Review of Recent Developments in the Integration of Deep Learning with GIS** und geht dabei insbesondere auf die darin referenzierte Arbeit **Semantic Segmentation-Based Building Footprint Extraction using Very High-Resolution Satellite Images and Multi-Source GIS Data** ein. Darin wird eine Vorgehensweise beschrieben, wie Gebäude in einem Satellitenbild automatisiert erkannt werden können. Dabei wird das verwendete Verfahren des Papers beschrieben und die notwendigen Grundbegriffe bezüglich der Satellitendaten und dem verwendeten Neuronalen Netz erklärt, um Nachvollziehbarkeit zu gewährleisten.

1 Einleitung

Mit den Fortschritten im Bereich Machine Learning geht ebenso die Frage der Einsatzgebiete einher. Ein noch relativ unbekannter Bereich davon ist die Auswertung von Satellitenbildern für Kartographierung und Datensammlung. Um diesem Problem entgegenzuwirken, wurde die DeepGlobe Challenge abgehalten, welche die automatisierte Verarbeitung von großflächigen Satellitenbildern in den Vordergrund rücken soll. In diesem Paper soll die Vorgehensweise und Methodik eines Teilnehmerteams näher beschrieben werden. Um Nachvollziehbarkeit der Methodik zu gewährleisten, wird zunächst der Stand der Technik, sowie relevante Begriffe und Konzepte bezüglich Machine Learning und Geographischen Informationssystemen erläutert. Daraufhin wird das in der Challenge verwendete Neuronale Netz und seine Architektur, sowie die Vorgehensweise der gewählten Methodik näher beschrieben. Anschließend erfolgt eine Diskussion der Ergebnisse, gefolgt von einer Zusammenfassung.

2 Stand der Technik

2.1 Neuronale Netze

Ein Problem bezüglich des Trainings von neuronalen Netzen ist die benötigte Menge an Trainingsdaten. Diese sollten qualitativ vielfältig sein, um die Verhältnisse der zu lernenden Konzepte auch wirklich lernen zu können. Ein möglicher Ansatz dafür ist die *Data Augmentation*, mit deren Hilfe ein kleiner Datensatz so verändert und vergrößert werden kann, dass die geforderte Qualität damit erreicht werden kann.

Data Augmentation ermöglicht durch Bildbearbeitung wie *Colour Augmentation*, *Slicing* oder *Rotation* eine einfache Veränderungsmöglichkeit für einen vorhandenen Datensatz, um Probleme wie *Overfitting*, bei dem die Trainingsdaten vom Netz lediglich „auswendig“ gelernt werden [4], zu vermeiden[5]. So wird beispielsweise durch Rotation des Bildes sichergestellt, dass mehrere Betrachtungswinkel eines Objektes miteinbezogen werden, und dadurch einem falschen Lernvorgang, welcher Objekte in einem gewissen Winkel bevorzugen würde, entgegengewirkt wird [10].

Ein Netz, welches für wenige Trainingsdaten konzipiert wurde, ist das populäre *U-Net*, welches für die Aufgabe der binären Segmentierung zur Zellen- und Gewebserkennung entwickelt wurde [9]. Im Medizinbereich sind tausende Trainingsbilder unrealistisch, für den spezifischen Anwendungsfall von U-Net standen nur 30 Bilder á 512 x 512 Pixel zur Verfügung, wodurch Data Augmentation für ein Sicherstellen der Robustheit und Invarianz des Netzes unabdingbar war [9].

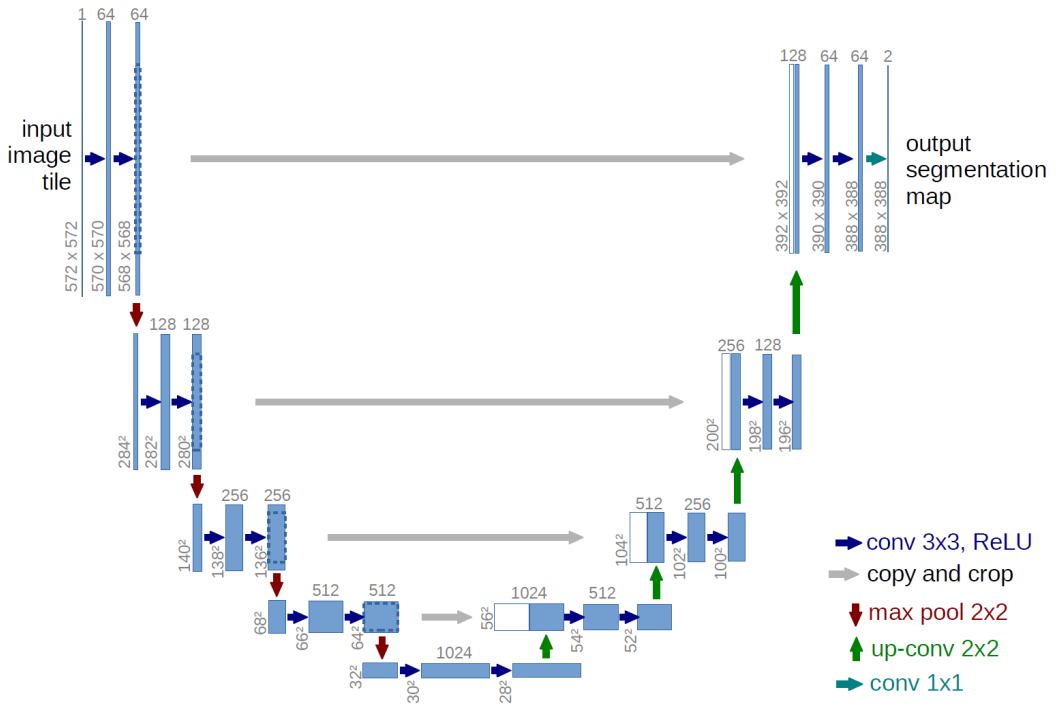

Abbildung 1: U-Net Architektur. Jede blaue Box entspricht einer Feature Map. Die diversen Operationen werden durch die Pfeile abgebildet [9].

Das U-Net ist, wie in Abb. 1 in „U-Form“ aufgebaut und setzt sich dabei aus dem *Contracting Path* und dem *Expansive Path* zusammen. Im ersten Teil erfolgen jeweils zwei 3 x 3 Faltungen gefolgt von einer ReLU und einer Max-Pool-Operation mit einem *Stride* von zwei für das Downsampling (dargestellt durch die roten Pfeile in Abb. 1). Im Expansive Path findet das Upsampling mit einer 2 x 2 Dekonvolution statt, welche die Feature Channels halbieren soll, gefolgt von einer Konkatenation mit der zugeschnittenen Feature Map aus der zugehörigen Schicht im Contracting Path (graue Pfeile; die gelernten Feature Maps werden benutzt, um die Rekonstruktion des Bildes zu untersützen und Verzerrungen zu vermeiden; Cropping ist notwendig, da durch die Faltungsoperationen die Randpixel verloren gehen) und erneut zwei 3 x 3 Faltungen mit einer ReLU. Am Ende erfolgt eine 1 x 1 Faltung, um die Feature Vektoren auf die gewünschte Anzahl an Klassen zuzuordnen.

Die Ergebnisse des U-Nets haben viele weitere Einsatzmöglichkeiten motiviert, unter anderem auch in Verbindung mit Satellitenbildern für eine prinzipiell ähnliche Segmentierung der Daten, wie sie bei der Biomedizin erforderlich ist, wo vor allem korrekte und akkurate Trennwanderkennung eine große Rolle spielt.

2.2 Geographic Information System

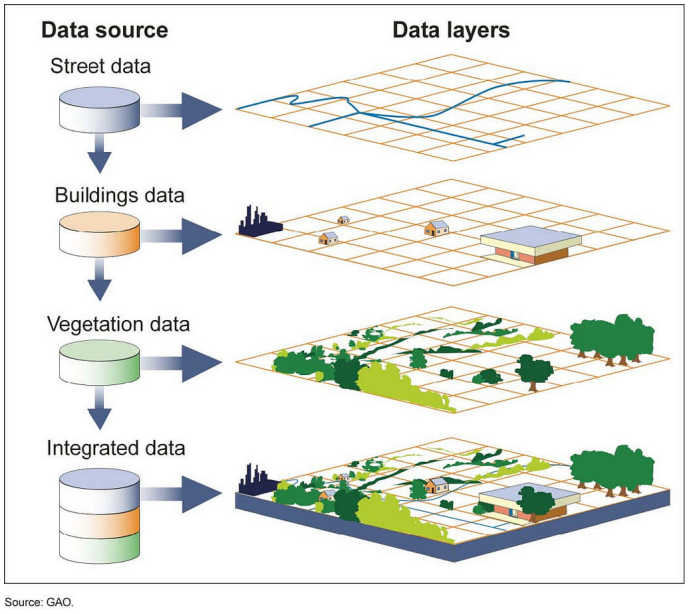

Abbildung 2: Aufbau eines GIS [8].

Die Definition des Begriffs **Geographic Information System (GIS)** hat sich im Laufe der Zeit geändert, um den technischen Änderungen und Einsatzmöglichkeiten gerecht zu werden [12]. *Nationalgeographic* definiert es dabei wie folgt:

Ein geographisches Informationssystem ist ein Computersystem zum Erfassen, Speichern, Prüfen und Anzeigen von Daten, die sich auf Positionen auf der Erdoberfläche beziehen. Durch die Verknüpfung scheinbar

nicht zusammenhängender Daten kann ein GIS Einzelpersonen und Organisationen helfen, räumliche Muster und Beziehungen besser zu verstehen[8].

In Abbildung 2 wird das Erstellen eines GIS-Datensatzes näher dargestellt. Dabei werden die zur Verfügung stehenden Informationen so miteinander kombiniert, dass sie für den Betrachter intuitiv verständlich sind. Auch wird damit gleichzeitiger Zugriff auf die voneinander unabhängigen Daten erlaubt. Es ermöglicht dem Nutzer geographische Verhältnisse und Zusammenhänge besser zu verstehen [7]. GIS kann dabei beispielsweise für Katastrophenmanagement bzw. deren Verhinderung, Bewässerungsversorgung und -Planung, oder für demografische Forschung verwendet werden, da damit reale Bedingungen abgebildet werden können [7].

3 Problemdefinition

In nachfolgenden Kapiteln soll eine ausgewählte Vorgehensweise zur automatisierten Extraktion von Gebäudeumrissen näher erläutert werden. Orientiert wird sich dabei an Li et al., welche mit Hilfe von GIS- und Satellitendatensätzen in Verbindung mit semantischer Segmentierung ein Modell für die sogenannte DeepGlobe Challenge trainiert haben [5].

DeepGlobe 2018 wurde als eine Satellite Image Understanding Challenge beworben [2]. Ziel dabei war es, Ansätze mit Machine Learning im Umgang mit Satellitendaten in den Vordergrund zu bringen. Dazu wurden öffentlich gestellte Probleme bezüglich Satellitenbildern und deren automatisierte Auswertung von Forschern aus unterschiedlichen Fachgebieten in Workshops gemeinsam gelöst [2]. Ebenso können die dadurch erworbenen Ergebnisse für zukünftige Herangehensweisen verwendet werden. Eine der Hauptaufgaben bezog sich dabei auf Building Detection mit dem Ziel, Gebäude in Satellitenbildern automatisch zu erkennen, um Informationen bezüglich der Urbanisierung zu erhalten [2].

Der zur Verfügung gestellte Datensatz stammt ursprünglich aus *SpaceNet*, einer ähnlichen Challenge, und beinhaltet 24.586 Bilder, die je 200m x 200m Fläche auf 650 x 650 Pixel abbilden. Dabei überlappen die Bilder nicht. Ebenso sind bereits manuell annotierte Daten bezüglich der Gebäude für die vier Städte Las Vegas, Paris, Shanghai und Khartoum vorhanden und liefern so etwas über 300.000 Gebäude [2]. Auch wird aufgrund der Natur des Vorgangs auf die Möglichkeit von falschen Annotationen hingewiesen. Als Baseline für die Auswertung werden die Ergebnisse [6] von *SpaceNet 2* herangezogen. Es sollte also mindestens ein total-F1-Score von 0.693 erreicht werden [2]. Der Datensatz beinhaltet einen 60/20/20 Split bezüglich Training/Validierung/Test. Der Input soll ein Satellitenbild und der Output eine Liste an Gebäudepolygonen sein [2].

4 Gebäudeumrisserkennung mit semantischer Segmentierung

4.1 Vorstellung des Netzes

Der von Li et al. verwendete Ansatz zur automatisierten Extraktion von Gebäudeumrissen wird in Abbildung 3 näher dargestellt. Dabei lässt sich der Ablauf in die drei Bereiche Data Preparation und Augmentation *(I)*, semantische Segmentierung *(II)* und Integration der Post-Processing Ergebnisse *(III)* unterteilen.

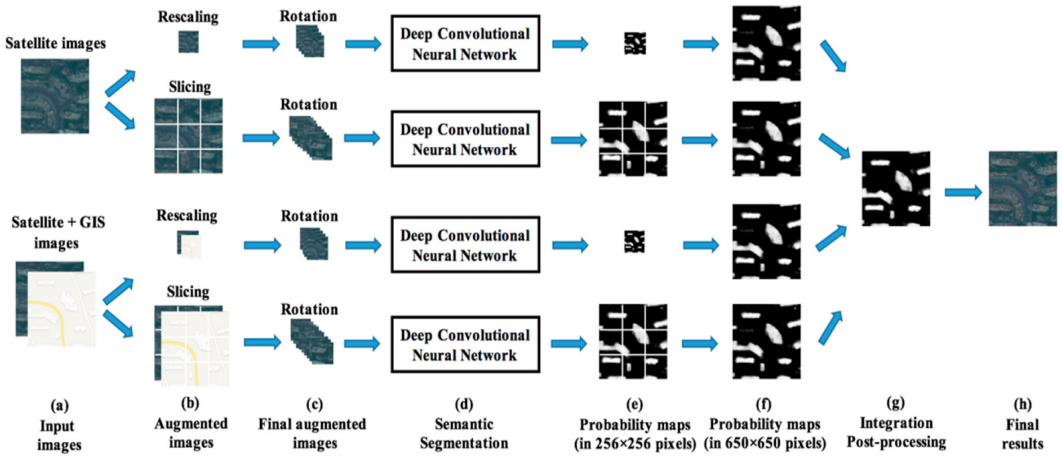

Abbildung 3: Flussdiagramm für die vorgeschlagene Herangehensweise einschließlich Data Preparation und Augmentation *(a-c)*, semantische Segmentierung für die Extraktion von Gebäudeumrissen *(d)* und Integration und Post-Processing der Ergebnisse *(e-h)* [5].

Zunächst wird der Informationsgehalt der von DeepGlobe zur Verfügung gestellten Datensätze mit Hilfsdatensätzen von weiteren öffentlich zugänglichen GIS-Maps, wie etwa **OpenStreetMap** oder **Google Maps**, erweitert, da dadurch weitere Informationen bezüglich Gebäudestandpunkten miteinbezogen werden können. Die Entscheidung zwischen der jeweiligen Quelle dafür fiel aufgrund variierender Informationsqualität sowie realitätsnaher Darstellung für jede Stadt separat [5]. Zusätzlich werden sie auf die Größe der Satellitenbilder (650 x 650 Pixel) normalisiert, um die Daten leichter verwenden und vergleichen zu können, wie in Abbildung 4 ersichtlich wird.

Um die Satelliten- und GIS-Bilder wie in Abb. 3 *(a)* zu kombinieren, werden die ersten fünf von acht Kanälen der Satellitenbilder beibehalten und die drei Kanäle der GIS-Bilder addiert, um deren Info mit einzubinden [5]. Somit erhält man für **„Satellite Images“** und für **„Satellite + GIS Images“** jeweils Bilder mit acht Kanälen. Ebenso können damit kleine Gebäude in den öffentlichen GIS-Maps besser abgebildet werden, da hier teilweise bedeutende Unterschiede mit der Ground Truth der Satellitenbilder vorliegen [5].

Anschließend wird die Data Augmentation mit Hilfe von Rescaling, Slicing und Rotation durchgeführt *(b-c)*. Dafür werden die 650 x 650 Bilder zu 256 x 256 Bildern

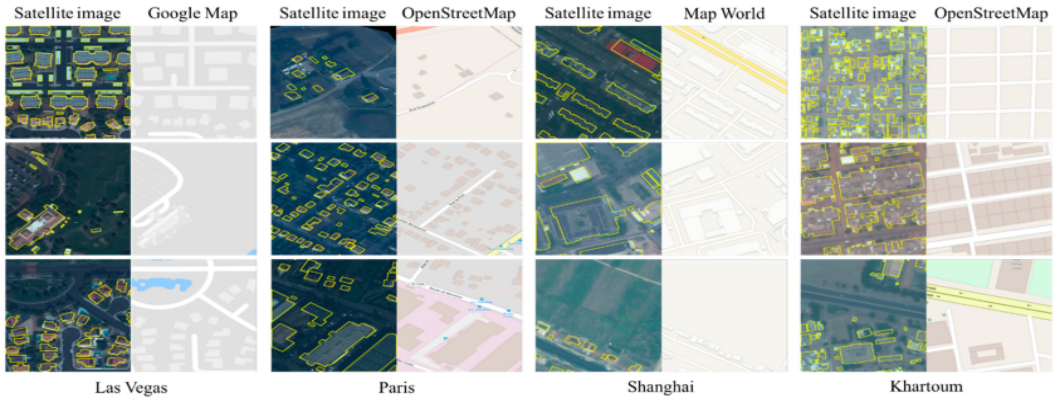

Abbildung 4: Beispiele von GIS-Datensätzen in Verbindung mit dazugehörigen Satellitenbildern. Die gelben Polygone stellen die jeweils annotierten Gebäudeumrisse dar [5].

skaliert, oder in 3 x 3 „Sub-Images“ geschnitten, die je 256 x 256 Pixel besitzen. Ebenso werden alle Trainingsdaten vier mal um je 90°-Drehungen rotiert [5]. Dies bewirkt eine Verbesserung der Ergebnisse bezüglich Overfitting, welches aufgrund der niedrigeren Anzahl der zur Verfügung stehenden Originalbilder ansonsten zu einem Problem werden kann [5]. Als Zwischenergebnis erhält man dadurch vier verschiedene Datensätze, wie in Abb. 3 *(c)* ersichtlich.

Darauf folgt in Abschnitt *(II)* die Anwendung der semantischen Segmentierung mit U-Net. Es wird deshalb gewählt, da U-Net ursprünglich für binäre Segmentierung mit einem relativ kleinen Trainingsdatensatz entwickelt wurde und dies mit der Aufgabenstellung der DeepGlobe Challenge übereinstimmt [2][5].

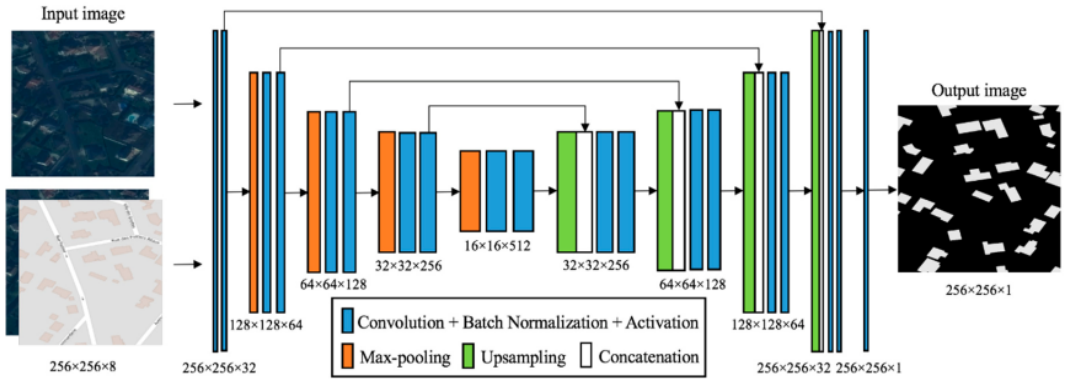

Abbildung 5: Architektur der U-Net-Implementierung für die Extraktion von Gebäudeumrissen [5].

Die Architektur der angepassten U-Net-Implementierung setzt sich, wie in Abb. 5 ersichtlich, aus sechs verschiedenen Teilen zusammen. Diese sind:

Die Feature-Extraktion mit Hilfe von 3 x 3 Faltungskernen in den Convolution-Layers *(blau)*, Batch-Normalization, um die Lernrate zu erhöhen *(blau)* [3], die nichtlineare Aktivierung, welche in dieser Anwendung mit der ReLU-Funktion umgesetzt

wird *(blau)* und die Max-Pooling-Schichten *(orange)* für den Convolution-Teil[5]. Daraufhin folgt die Deconvolution durch Upsampling *(grün)* und Concatenation mit den zugehörigen Feature Maps aus höheren Schichten *(weiß)* [5].

Für die letzte Batch-Normalization-Schicht wird die Sigmoidfunktion als Aktivierungsfunktion benutzt, welche Ergebnisse zwischen 0 und 1 liefert, um die pixelweise Wahrscheinlichkeit für das Auftreten eines Gebäudes im Bild zu erhalten [5]. Dieses Ergebnis wird schlussendlich in eine binäre Darstellung umgewandelt, wie es in Abb. 5 mit dem Output Image dargestellt ist, indem man sich auf 0.5 als Schwellwert geeinigt hat [5].

Die Integration und das Post-Processing der Ergebnisse dient dazu, die Ergebnisse weiter zu detaillieren und anzupassen. Dabei werden die Ergebnispixel einem Gebäudebereich zugeordnet. Das Ergebnis dabei ist eine Probability Map, also eine zweidimensionale Darstellung der Wahrscheinlichkeiten. Jedes Pixel dieser Map steht also für die vorhergesagte Wahrscheinlichkeit, dass es sich dabei um ein Gebäude handelt. [5]. Im Beispiel von Li et al. liegen durch Kombination der Ergebnisse vier Gruppen von Probability Maps pro Validierungsset mit einer Größe von 650 x 650 Pixel vor. Diese entstehen durch *(1)* die Satellitenbilder + Rescaling, *(2)* Satellitenbilder + Slicing und selbes für die fusionierten Daten aus Satellitenbildern + GIS-Maps *(3+4)* (siehe Abb. 3 *(c)*).

Anschließend wird eine zweistufige Herangehensweise vorgeschlagen, um die Ergebnisse der einzelnen Modelle einzubinden und zu verwenden. Zunächst wird für **„Satellite Images“** und für **„Satellite + GIS Images“** je der durchschnittliche Pixelwert von zwei erhaltenen Probability Maps in einer neuen Probability Map festgehalten. Das Ergebnis sind also je eine Map für **„Satellite Images“** und für **„Satellite + GIS Images“**. Daraufhin wird selbes Verfahren für die zwei neuen Maps durchgeführt, was schlussendlich zu einer finalen Probability Map führt.

Für das darauf folgende Post-Processing werden zwei Strategien für eine Optimierung der Ergebnisse umgesetzt. Zunächst wird der Schwellwert für die Zuordnung der Ergebnispixel zum Label „Gebäude“ von 0.45 auf 0.55 gesetzt und auf die Probability Map angewandt [5]. Im zweiten Schritt wird die minimale Polygongröße, welche die kleinste zugelassene Größe eines Gebäudeelements beschreibt, von 90 auf 240 Pixel angehoben. Beide Schritte werden ebenfalls in den Testdaten angewendet.

4.2 Auswertung des Models

Für die Auswertung der Ergebnisse werden grundsätzlich zwei Möglichkeiten vorgestellt [5]. Diese belaufen sich auf Pixel-Based und Object-Based Methoden.

Erstere beinhaltet einen direkten Abgleich des binären Ergebnisbildes mit dem binären Ground-Truth-Bild. Bei der objektbasierten Methode hingegen werden die Polygone der Ergebnisse mit denen der Ground Truth, welche im sogenannten geojson-Format vorliegt, verglichen und müssen dementsprechend vorher konvertiert werden [5]. Letzteres findet bei Li et al. Anwendung, da dies vergleichsweise mehr Fokus auf die korrekte Erkennung und Identifizierung der kompletten Gebäudebereiche inklusive Umriss legt. Zusätzlich wurde es von der DeepGlobe Challenge als Vorgabe gestellt [5]. Für die Auswertung, ob ein vorhandenes Polygon korrekt erkannt wurde

wird folgende Formel gemäß DeepGlobe Challenge für Intersection over Union (IoU) verwendet [2][5]:

$$\text{IoU} = \frac{Area(A \cap B)}{Area(A \cup B)}$$

Dabei wird die vorhandene Schnittmenge eines vorhergesagten Objektes *(A)* mit der vorher bereits annotierten Ground Truth *(B)* im Verhältnis zur Vereinigung der beiden Mengen betrachtet. Je mehr sich dieser Wert an 1 annähert, desto zutreffender ist die Vorhersage des Netzes. Falls ein vorhergesagtes Polygon *(A)* mit mehr als einem Ground-Truth-Polygon *(B)* überlappt, wird für sämtliche Ground-Truth-Polygone *(B)* aus der Schnittmenge der *IoU-Wert* berechnet und sich dann für das eine Ground-Truth-Polygon *(B)* entschieden, welches das größte Ergebnis liefert [5]. Dies kann beispielsweise der Fall sein, falls eine Ansammlung von kleineren Gebäuden als ein großes Gebäude erkannt wurde.

Bezüglich der Klassifikation stellen korrekt erkannte Polygone richtig positive Einträge dar. Fälschlicherweise erkannte Gebäude werden durch die falsch positive Menge abgebildet. Falsch negative Erkennungen beziehen sich auf nicht erkannte Polygone der Gebäude. Ein Gebäude gilt für die Auswertungsphase der DeepGlobe Challenge dann als korrekt erkannt, sobald der IoU-Wert größer ist als 0.5 [2].

Als finaler Bewertungsparameter der Challenge wird der F1-Score für alle vier Städte einzeln berechnet und daraus wie folgt der Durchschnitt erhalten[2]:

$$\textbf{F1} = \frac{F1_{Vegas} + F1_{Paris} + F1_{Shanghai} + F1_{Khartoum}}{4}$$

Der F1-Score ist dabei ein Qualitätsmaß, welches mit Hilfe von harmonischer Mittelung aus Precision und Recall erstellt wird [11].

Precision ist ein Maß für Genauigkeit. Beispielsweise die Zahl der korrekt vorhergesagten Gebäude verglichen mit Gesamtzahl **aller vorhergesagten** Gebäude:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

Recall ist ein Maß für Vollständigkeit. Beispielsweise die Zahl der korrekt vorhergesagten Gebäude verglichen mit Gesamtzahl **aller** Gebäude:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

4.3 Ergebnisse

Für das Training und die Auswertung des Netzes wurde das *Keras-Framework* verwendet, welches sich durch seine leichte und nutzerfreundliche Handhabung auszeichnet [1]. Auch wurde aufgrund der Unterschiede zwischen den Orten für jede der vier Städte separat trainiert und validiert [5].

Die Ergebnisse der Validierungsdaten bezüglich Precision, Recall und F1-Score lassen sich aus Abb. 6 entnehmen. Zwischen den Städten tauchen große Unterschiede auf. Die Bilder für Khartoum weißen eine große Varianz bezüglich der Gebäudestrukturen auf [5]. Selbst bei manueller Betrachtung ist schlecht erkennbar, ob ein

Index	Las Vegas	Paris	Shanghai	Khartoum
TP	27,526	3097	11,323	3495
FP	1629	564	3835	1968
FN	5098	1441	9661	3951
Precision	0.9441	0.8459	0.7470	0.6398
Recall	0.8437	0.6825	0.5396	0.4694
F1-score	0.8911	0.7555	0.6266	0.5415

Abbildung 6: Ergebnisse bezügl. der Validierungsdaten [5].

Gebäudekomplex ein oder mehrere Polygone aufweisen sollte. Auch fehlen hier viele Einträge der verwendeten GIS-Maps [5].

Für das Finale der DeepGlobe Challenge wurde das Model ebenso mit nicht-zugänglichen Satellitenbildern getestet. Dieser Datensatz besteht aus unbekannten, nicht-annotierten Bildern aus ähnlichen Regionen in den vier Städten[5]. Die Ergebnisse dieses Tests sind fast identisch mit denen aus Abb. 6 und unterscheiden sich nur geringfügig in der zweiten bis dritten Nachkommastelle [5].

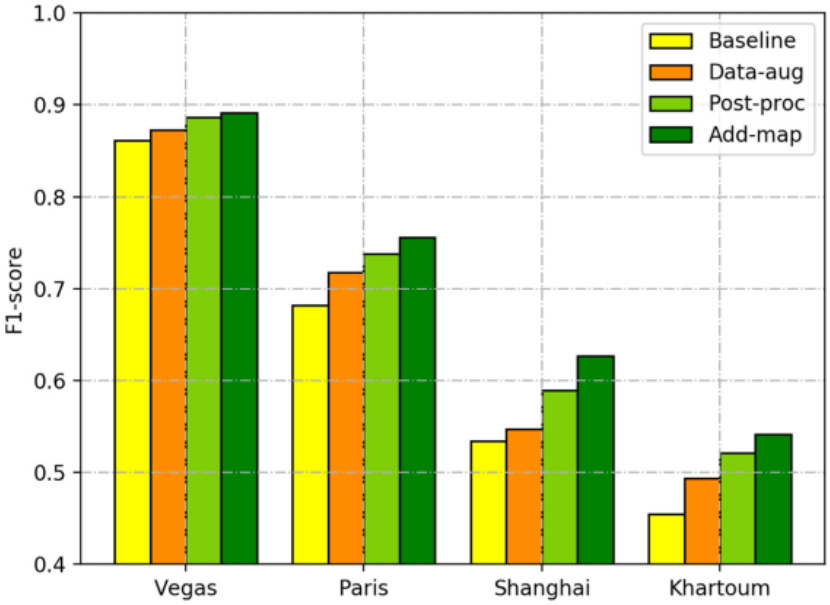

Abbildung 7: F1-Score der verschiedenen Zwischenschritte [5].

Die effektiven F1-Scores der verschiedenen Strategien werden in Abb. 7 dargestellt. *Baseline* bezieht sich auf den F1-Score einzig mit Rescaling der Satellitenbilder. *Data-aug* bindet die Satellitenbilder mit Slicing und Rotation mit ein. *Post-proc* stellt die Ergebnisse auf den Baseline + augmentierten Daten nach Durchführung der Post-Processing Strategien dar. *Add-map* bildet die Kombination mit den Hilfsdatensätzen der GIS-Maps mit ab. Im Vergleich zur bloßen Baseline lässt sich ein Anstieg von drei bis über neun Prozent feststellen, wobei dieser am stärksten in

Khartoum ausgeprägt ist, was vermuten lässt, dass die Vorgehensweise vor allem in weniger eindeutigen Datensätzen Verbesserungspotential bewirkt[5].

Der Total-F1-Score beträgt 0.704, womit sich, verglichen mit der festgelegten Baseline, eine Verbesserung verzeichnen lässt (0.693, 0.643 und 0.579) [5]. Verglichen mit der bisher besten Methode der Baseline (ebenfalls eine U-Net-Implementierung in Verbindung mit GIS-Datensätzen) findet ein Anstieg von *1.1%*, mit der zweitbesten Methode (ein kantenbasierter Ansatz) um *6.1%* und mit der drittbesten Methode (ebenfalls kantenbasiert) um *12.5%* statt.

5 Zusammenfassung

Zusammenfassend lässt sich also durch die Einbindung von GIS-Datensätzen im Bereich Machine Learning Potential feststellen, wobei U-Net dabei aufgrund seines ursprünglichen Anwendungsgebiets gut geeignet ist, Gebäude effizient zu segmentieren. Die GIS-Datensätze können dabei je nach Varianz der Gebäudestruktur großen positiven Einfluss auf die Erkennungsrate haben. Ebenso verbessert Data Augmentation die Ergebnisse dann merklich, wenn weniger ursprüngliche Trainigsdaten vorhanden sind. Dennoch ist eine durchgehend effiziente Extraktion nicht ohne weiteres möglich, da vor allem in weniger industrialisierten Gegenden wie Khartoum die Gebäude nur schwer vom Hintergrund und untereinander trennbar sind.

Ebenso lässt die vorgestellte Methodik an manchen Stellen einige Fragen offen. So werden beispielsweise Werte abweichend von der Vorgabe seitens DeepGlobe festgelegt, ohne die Wahl davon genauer zu begründen, was vermuten lässt, dass die Grenzwerte durch zufälliges Testen gefunden wurden. Hier herrscht zusätzlich Unstimmigkeit mit den Vorgaben, da laut DeepGlobe Challenge ein Gebäude mindestens 21 Pixel besitzen muss. In vorliegendem Paper wird sich ohne Quellenangabe auf eine Mindestgröße von 90 Pixel und einen Mindestscore von 0.45 für das Label „Gebäude“ berufen. Da keine Referenzen angegeben sind, könnten diese Werte eventuell noch von der originalen U-Net-Implementierung stammen.

Literatur

[1] Francois Chollet. *Deep learning with Python*. Simon and Schuster, 2021.

[2] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[4] Will Koehrsen. Overfitting vs. underfitting: A complete example. *Towards Data Science*, 2018.

[5] Weijia Li, Conghui He, Jiarui Fang, Juepeng Zheng, Haohuan Fu, and Le Yu. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source gis data. *Remote Sensing*, 11(4):403, 2019.

[6] David Lindenbaum. 2nd spacenet competition winners code release, Jul 2017.

[7] Shyama Mohan and MVSS Giridhar. A brief review of recent developments in the integration of deep learning with gis. *Geomatics and Environmental Engineering*, 16(2), 2022.

[8] Nationalgeographic. Gis (geographic information system). `https://education.nationalgeographic.org/resource/geographic-information-system-gis`, 2022. Aufgerufen: 2023-07-01.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[10] Sanjeevan Shrestha and Leonardo Vanneschi. Improved fully convolutional network with conditional random fields for building extraction. *Remote Sensing*, 10(7):1135, 2018.

[11] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15(1):1–28, 2015.

[12] William F Wieczorek and Alan M Delmerico. Geographic information systems. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(2):167–186, 2009.