

Domain Adaptation für die semantische Segmentierung unter  
Einbezug von biophysikalisch-spektralen Eigenschaften

Zur Erlangung des akademischen Grades eines  
DOKTORS DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)

von der KIT-Fakultät für  
Bauingenieur-, Geo- und Umweltwissenschaften

des Karlsruher Instituts für Technologie (KIT)

genehmigte

DISSERTATION

von

Fabian Schenkel

Tag der mündlichen Prüfung: 18.12.2023

Referent: Prof. Dr. Stefan Hinz

Korreferent: Prof. Dr. Franz Rottensteiner

Karlsruhe (2023)



## Zusammenfassung

Die Segmentierung und Klassifikation von Bodenbedeckung ist für viele Anwendungen im Bereich der Fernerkundung ein wichtiger Prozess. In diesem Zusammenhang ist vor allem die Untersuchung der Oberfläche in urbanen Gebieten wichtig, da sich dort häufig die Struktur der Bodenbedeckung ändert. Dementsprechend sind im Rahmen der Stadtplanung und -verwaltung eine genaue Analyse dieser Flächen von Interesse. Mittlerweile lebt die Hälfte der Weltbevölkerung in urbanen Gebieten und es resultieren durch diese Verstädterung entsprechende Probleme. Aufgrund der entsprechend hohen Dichte an Gebäuden, bilden sich sogenannte städtische Wärmeinseln, welche für eine merklich höhere Temperatur in den urbanen Gebieten im Vergleich zum Umland sorgen. Außerdem steigt das Risiko für Hochwasser durch die umfangreiche Versiegelung des Bodens und durch den verdichteten Verkehr sinkt die Luftqualität. In diesem Zusammenhang spielt die Menge, die Verteilung und der Zustand von Vegetation in diesen Gebieten eine entscheidende Rolle. Dadurch lässt sich zum einen eine gemäßigte Temperatur erreichen, Ablaufmöglichkeiten bei Starkregen herstellen und für einen Schadstoffabbau in der Luft sorgen. Für die Analyse von Vegetation und weiteren Objektklassen in urbanen Gebieten wird in der Regel auf hochauflösende Luft- und Satellitenbilddaten zurückgegriffen, da die relevanten Objekte jeweils häufig keine großen Flächen abdecken. Für die Klassifikation von Bodenbedeckung wurden in den letzten Jahren vermehrt Methoden aus dem Bereich des maschinellen Lernens eingesetzt. Dabei handelt es sich vor allem um Anwendungen zur semantischen Segmentierung basierend auf Convolutional Neural Networks (CNNs). Diese neuronalen Netze werden auf Grundlage von Beispielen mit einer entsprechenden Annotation trainiert und können anschließend zur Auswertung auf weitere Daten angewandt werden. Allerdings ist es häufig der Fall, dass wenn ein Netzwerk auf einem bestimmten Datensatz trainiert wurde, die Klassifikationsqualität sinkt, falls ein anderer Datensatz mit unterschiedlichen Charakteristika zur Auswertung eingesetzt wird. Da das Annotieren von Daten aber einen zeitaufwändigen Prozess darstellt, ist es von Interesse, bereits trainierte neuronale Netze weiterzuverwenden. Eine Möglichkeit dies umzusetzen ist die sogenannte Domain Adaptation aus dem Bereich des Transferlernens. Dabei wird versucht die Methode oder die Datengrundlage dementsprechend anzupassen, dass der notwendige Annotationsaufwand reduziert wird. Es findet auf diese Art eine

Anpassung von der sogenannten Quelldomäne zur Zieldomäne statt. In dieser Arbeit werden Analysen und Methoden aus dem Bereich der Domain Adaptation für die semantische Segmentierung vorgestellt. Dabei wird sowohl die überwachte Domain Adaptation (annotierte Daten sind in der Zieldomäne vorhanden) wie auch die unüberwachte Domain Adaptation (keine annotierten Daten sind in der Zieldomäne vorhanden) berücksichtigt. Im Rahmen der überwachten Domain Adaptation wird auf Grundlage einer CNN-basierten Encoder-Decoder-Struktur zur semantischen Segmentierung eine Möglichkeit vorgestellt um die Auswertergebnisse bei Reduktion der notwendigen Trainingsdaten zu verbessern. Dies wird ermöglicht durch die Erweiterung des auf der Quelldomäne trainierten CNNs mit Adaptionmodulen. Anschließend wird eine unüberwachte Domain Adaptation unter Einsatz von Cycle-Consistent Adversarial Networks vorgestellt. Dabei wird ein Style-Transfer zwischen der Quelldomäne und der Zieldomäne ermöglicht, um so die Charakteristika der jeweiligen Datensätze entsprechend anpassen zu können. Um das CNN für die semantische Segmentierung anzupassen, wurde ein kontinuierliches Trainingsschema konzipiert und für den Domain Adaptation-Prozess eingesetzt. Dadurch konnte eine signifikante Verbesserung der Klassifikationsergebnisse erreicht werden. Im Folgenden wurde dieser Ansatz in Hinblick auf die Segmentierung von Vegetation weiterentwickelt. Dazu wurde zunächst ein Verfahren zu Erstellung eines geeigneten Quelldomäne-Datensatzes untersucht, welches auf der Kombination von Vegetationsindices und CNNs beruht. Anschließend wurde die Segmentierung von Vegetation im Zusammenhang mit einer Cycle-Consistent Adversarial Network-basierten Domain Adaptation untersucht. Dabei konnte festgestellt werden, dass zwar räumliche Informationen zuverlässig angepasst werden, aber spektrale Zusammenhänge, welche für die Klassifikation von Vegetation relevant sind, verloren gehen. Basierend darauf wurde eine Erweiterung der Fehlerfunktion durchgeführt, um die entsprechenden Relationen während des Transfers beizubehalten und somit die Auswertergebnisse zu verbessern.

## Abstract

The segmentation and classification of land cover from airborne and spaceborne data is an important task for many applications in the field of remote sensing. Especially interesting is the investigation of the surface of urban areas, since the structure of land cover is changing rapidly there. For urban planning and management an accurate analysis of these properties is important. Nowadays, half of the world population lives in urban areas and this results in different problems. Caused by the high density of buildings the so-called urban heat island effect occurs, leading to a higher temperature in cities compared to the surrounding areas. Additionally, the risk for floods increases due to the large areas of impervious surface and the high traffic causes a reduction of the air quality. An important factor for this is the amount, distribution and condition of vegetation in these areas. Vegetation can lower the temperature, give spaces in case for strong rainfall and increases the air quality. For the investigation of vegetation and other object classes in urban areas usually high resolution airborne and spaceborne data is used, since the size of the relevant objects is small. In the last years more and more machine learning-based methods have been used for the classification of land cover. Especially, methods for semantic segmentation based on convolutional neural networks (CNN). These neural networks are trained on example images with a corresponding ground truth and are then applied to following data. But it is usually the case that if a network is trained on a certain dataset, the classification performance decreases if another dataset with different characteristics is used for evaluation. Since the annotation of data is a time-consuming task, it is desirable to use already trained networks. One possibility for this is domain adaptation which is a part of transfer learning. For domain adaptation it is the goal to adapt a method or the data in a certain way to reduce the necessary amount of annotated data, which means that an adaption from a source to a target domain is performed. In this work analyses and methods from the field of domain adaptation for semantic segmentation are presented. Additionally, to the case of supervised domain adaptation (annotated data of the target domain is available) also unsupervised domain adaptation (no annotated data of the target domain is available) is investigated. For supervised domain adaptation a method is presented with a CNN-based encoder decoder structure for semantic segmentation to reduce the necessary training data. This is done by extending the CNN which is trained on the

source domain with adaption modules. Consequently, a cycle-consistent adversarial network-based unsupervised domain adaptation method is presented. Using this kind of network, a style-transfer between the source and the target domain is possible to adapt the corresponding characteristics of the datasets. To adapt the CNN for semantic segmentation a continuous training scheme is developed and used for the domain adaptation process. Doing this, a significant improvement of the classification results has been possible. In the next step this principle is transferred and adapted for the segmentation of vegetation. For this purpose, a method based on vegetation indices and CNNs has been investigated to create a suitable source domain dataset. Following, the segmentation of vegetation based on cycle-consistent adversarial networks has been investigated. A result was that these methods can reliably transfer spatial information but spectral relations get lost, which are important for the classification of vegetation. Based on this circumstance, an extension of the loss function has been performed to maintain the corresponding relations during the transfer and to increase the classification results.

Domain Adaptation für die semantische Segmentierung unter  
Einbezug von biophysikalisch-spektralen Eigenschaften

von

Fabian Schenkel



# Inhaltsverzeichnis

|       |  |    |
|-------|--|----|
| 1     | Einleitung und Einbindung der Arbeit in den wissenschaftlichen Kontext .....                             | 11 |
| 1.1   | Transferlernen und Domain Adaptation .....   | 12 |
| 1.2   | Segmentierung und Klassifikation von Vegetation .....  | 15 |
| 1.3   | Motivation und Beiträge der Arbeit.....  | 19 |
| 2     | Grundlagen.....  | 23 |
| 2.1   | Künstliche neuronale Netze .....   | 23 |
| 2.2   | Backpropagation .....  | 25 |
| 2.3   | Aktivierungsfunktionen.....  | 26 |
| 2.4   | Convolutional Neural Networks.....   | 29 |
| 2.5   | Metriken für die Evaluation .....  | 34 |
| 2.5.1 | F <sub>1</sub> -Wert .....   | 34 |
| 2.5.2 | Intersection over Union .....  | 36 |
| 3     | Datensätze .....   | 37 |
| 3.1   | ISPRS Potsdam-Datensatz .....  | 37 |
| 3.2   | Luftbilddatensatz Sommer und Luftbilddatensatz Herbst .....  | 38 |
| 3.3   | Satellitenbilddatensatz Sommer .....   | 39 |
| 4     | Domain Adaptation mit Adaptionmodulen .....  | 43 |
| 4.1   | Methode.....   | 44 |
| 4.2   | Ergebnisse und Diskussion .....  | 47 |
| 5     | Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung .....         | 51 |
| 5.1   | Methode.....   | 55 |
| 5.2   | Ergebnisse und Diskussion .....  | 58 |
| 6     | Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs ..... | 63 |
| 6.1   | Methode.....   | 66 |
| 6.2   | Ergebnisse und Diskussion .....  | 71 |
| 6.2.1 | Ergebnisse auf Grundlage des NDVI.....   | 72 |
| 6.2.2 | Ergebnisse auf Grundlage des SAVI.....   | 74 |
| 6.2.3 | Ergebnisse auf Grundlage des EVI.....  | 76 |

|       |  |     |
|-------|--|-----|
| 7     | Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks ..... | 79  |
| 7.1   | Methode.....   | 80  |
| 7.2   | Ergebnisse und Diskussion .....  | 83  |
| 7.2.1 | Style-Transfer zwischen den Datensätzen .....  | 83  |
| 7.2.2 | Transfer spektraler Eigenschaften .....  | 86  |
| 7.2.3 | Segmentierung von Vegetation .....   | 94  |
| 8     | Zusammenfassung und Ausblick .....   | 99  |
| 9     | Literaturverzeichnis.....  | 105 |
| 10    | Abbildungsverzeichnis.....   | 113 |
| 11    | Tabellenverzeichnis.....   | 117 |

## 1 Einleitung und Einbindung der Arbeit in den wissenschaftlichen Kontext

Für viele Anwendungen und Analysen im Bereich der Fernerkundung ist die Untersuchung von Luft- und Satellitenbilddaten von entscheidender Bedeutung um Informationen über die Bedeckung, die Nutzung und den Zustand der jeweiligen Oberfläche zu erhalten. Ein Anwendungsbereich ist dabei die Klassifikation der Bodenbedeckung, bei welcher das Ziel ist, eine möglichst korrekte Zuordnung einzelner Pixel zu bestimmten Objektklassen der entsprechenden Bodenoberfläche zu erhalten. Eine besonders anspruchsvolle Aufgabe stellt dabei die Segmentierung und Klassifikation von urbanen Gebieten dar, da sich in diesen Umgebungen häufig viele unterschiedliche und von der Größe stark begrenzte Objekte befinden. Zur Umsetzung dieser Analyse im Rahmen der Bodenbedeckungssegmentierung und -klassifikation können unterschiedliche Methoden eingesetzt werden. In den letzten Jahren haben sich vor allem Ansätze aus dem Bereich des maschinellen Lernens verstärkt für diese Aufgabe als sehr geeignet herausgestellt. Bei diesen Methoden handelt es sich um Anwendungen, welche anhand von Beispieldaten Strukturen in den Daten erkennen und lernen, diese für unterschiedliche Aufgaben zu verwenden. Für die Aufgabe der Klassifikation der Bodenbedeckung stellt dabei die semantische Segmentierung ein geeignetes Vorgehen dar. Dabei wird jedem Pixel in dem jeweiligen zu untersuchenden Bild eine Objektklasse zugeordnet, sodass eine entsprechende Auswertung des gesamten Bildes erfolgt. Für das Training dieser Methoden werden im Vorfeld die für die jeweilige Analyseaufgabe relevanten Objektklassen bestimmt und es findet eine entsprechende Annotation der Bilddaten statt. Dieser Annotationsschritt erfolgt in der Regel manuell und stellt einen zeitintensiven Prozess dar [1]. Zusätzlich ist es häufig der Fall, dass wenn eine auf einem bestimmten Datensatz trainierte Anwendung auf einen anderen Datensatz mit anderen Charakteristika angewandt wird, die Auswertequalität deutlich abnimmt. Dies geschieht aufgrund des sogenannten Domain-Shifts [2]. Dieser Effekt kann durch den Einsatz eines anderen Sensors eintreten, falls dieser beispielsweise über eine unterschiedliche Auflösung verfügt. Ebenfalls ist dies möglich durch einen anderen Winkel zum Boden bei einer unterschiedlichen Ausrichtung des Messinstruments. Außerdem

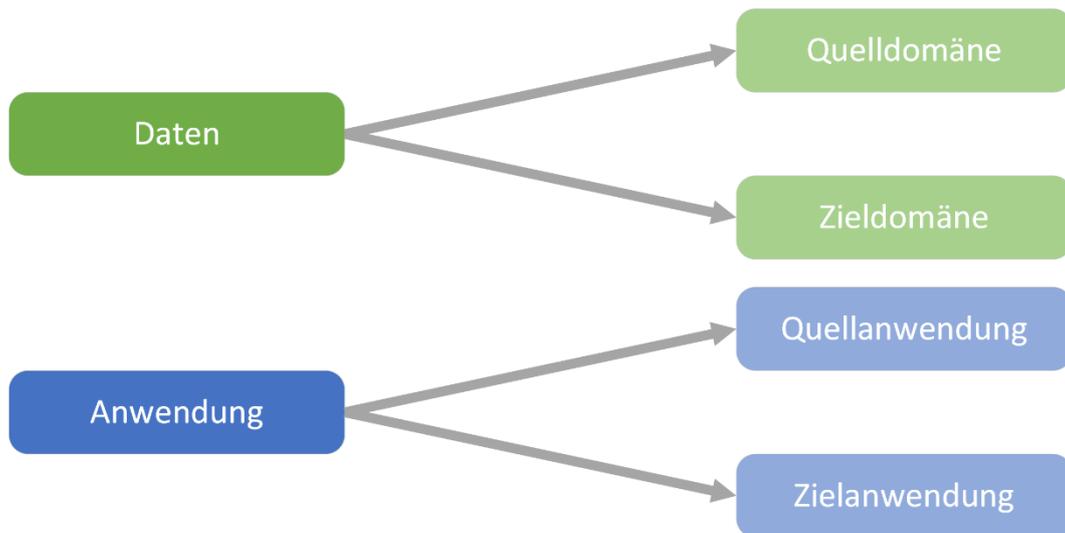


Abbildung 1: Die für den Bereich des Transferlernens relevanten Kriterien. Auf der Datenebene handelt es sich um die Quell- und Zieldomäne und auf der Anwendungsebene um die Quell- und Zielanwendung.

können noch Unterschiede in der Helligkeit oder des Schattenwurfs von Objekten bedingt durch Wetterbedingungen die Auswertequalität senken, ebenso wie andere Charakteristika in den Bildern der einzelnen Datensätze. Um diesen notwendigen Annotationsaufwand bei gleichbleibender Auswertequalität zu reduzieren, ist es dementsprechend von Interesse, bereits zuverlässige vorhandene Methoden und erlangtes Wissen für den entsprechenden neuen Analyseschritt wiederzuverwerten. Für eine mögliche Anpassung können dafür Techniken des Transferlernens verwendet werden.

### 1.1 Transferlernen und Domain Adaptation

Unter Transferlernen versteht man eine Technik aus dem Bereich des maschinellen Lernens, bei welcher Wissen aus einer bereits gelernten Aufgabe weiterverwendet wird, um die Qualität einer mit dieser verwandten Aufgabe zu verbessern. Dieser Transfer kann unter verschiedenen Aspekten und auf unterschiedlichen Ebenen stattfinden. Um die Art des Transferlernens zuzuordnen sind zwei Merkmale zu betrachten. Zum einen kann Transferlernen auf Ebene der Daten mit Quell- und Zieldomäne stattfinden und zum anderen auf Ebene der Anwendung mit Quell- und Zielanwendung (siehe Abbildung 1). Auf der Datenebene findet dabei der Transfer von der Quell- zur Zieldomäne und auf Ebene der Anwendung von der Quell- zur Zielanwendung statt. Je nachdem, auf welcher Ebene das Transferlernen stattfindet, also je nachdem, ob Quell- und Zieldomäne oder Quell-

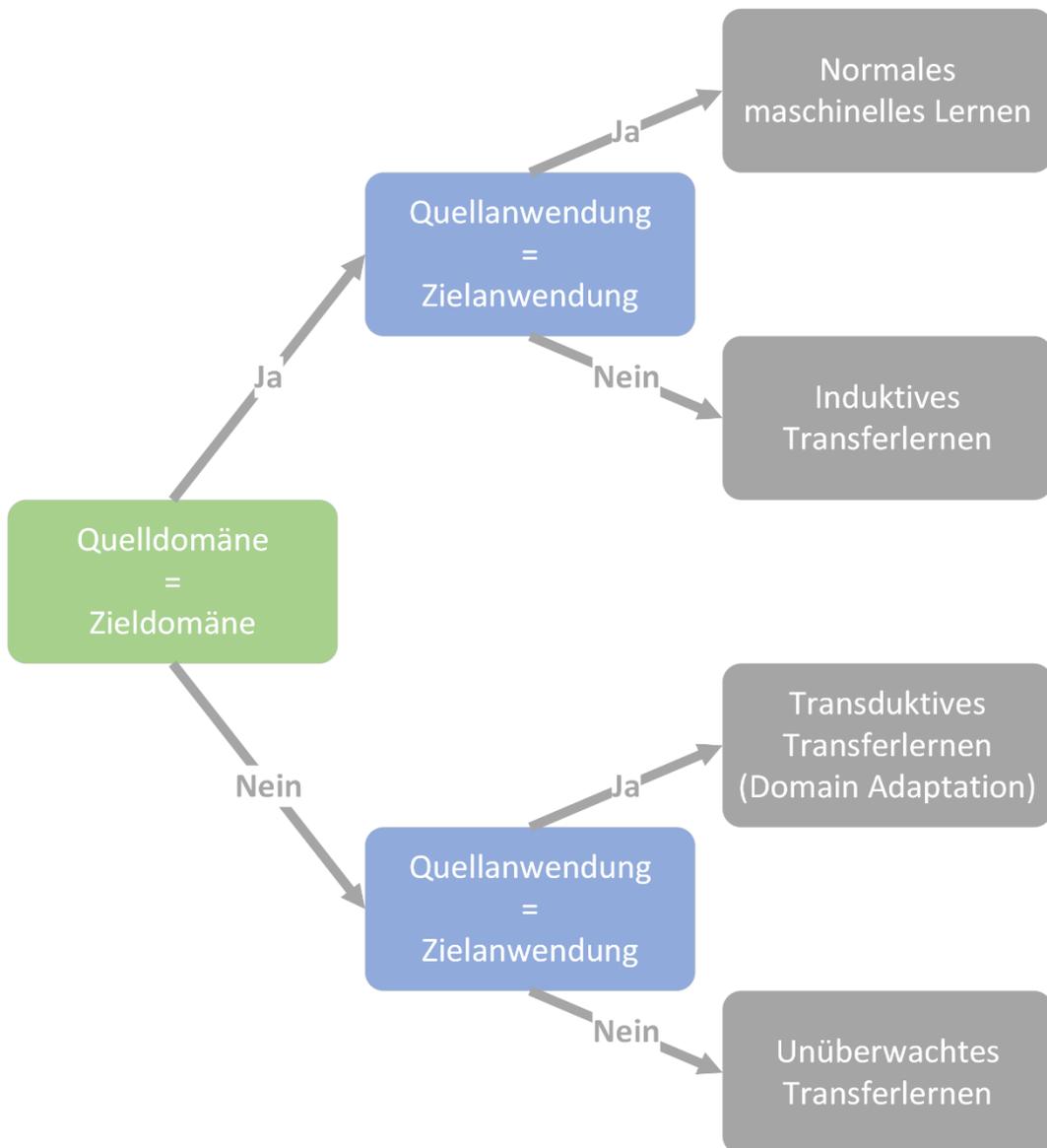


Abbildung 2: Die möglichen Varianten des Transferlernens: Induktives, transduktives (Domain Adaptation) und unüberwachtes Transferlernen.

und Zielanwendung unterschiedlich sind, wird zwischen den einzelnen Methoden differenziert. Falls die Quell- und Zieldomäne gleich sind und die Quell- und Zielanwendung unterschiedlich, so wird von „Induktivem Transferlernen“ gesprochen. Falls die Quell- und Zieldomäne unterschiedlich, aber die Quell- und Zielanwendung gleich sind, handelt es sich um „Transduktives Transferlernen“ oder auch „Domain Adaptation“. Für den Fall, dass sowohl Quell- und Zieldomäne als auch Quell- und Zielanwendung verschieden sind, spricht man von unüberwachtem Transferlernen [3]. Eine Übersicht über die unterschiedlichen Kombinationen ist in Abbildung 2 dargestellt. Im Bereich der Klassifikation der

Bodenbedeckung ist der Einsatz von Domain Adaptation ein besonders wichtiges Verfahren um die Zuverlässigkeit unterschiedlicher Methoden zu gewährleisten. Häufig existiert eine bestimmte Auswertemethode, welche auf unterschiedliche Daten angewandt werden soll ohne dabei große Mengen der Zieldomäne manuell annotieren zu müssen. In der Regel ist das Ziel eine bestimmte Auswertemethode auf einem umfangreich annotierten Quelldomäne-Datensatz vorab zu trainieren und die entsprechende zuverlässige Auswerteeigenschaft auf einen bestimmten Zieldomäne-Datensatz, bei welchem wenige oder keine annotierten Daten vorliegen, übertragen werden. Für die Aufgabe der Domain Adaptation im Bereich der Bildauswertung kamen in der Vergangenheit unterschiedliche Methoden zum Einsatz [4]. Dabei gibt es zum einen Verfahren zur überwachten Domain Adaptation und der unüberwachten Domain Adaptation. Bei den überwachten Methoden ist es der Fall, dass ein Teil der Daten der Zieldomäne annotiert sind. Bei den unüberwachten Methoden wird vollständig auf den Einsatz von annotierten Daten in der Zieldomäne verzichtet. Es haben sich in diesem Zusammenhang in den vergangenen Jahren vor allem für den Bereich der unüberwachten Domain Adaptation Algorithmen basierend auf Generative Adversarial Networks [5] und den auf diesem Konzept aufbauenden Cycle-Consistent Adversarial Networks [6] als vielversprechend herausgestellt. Eine Arbeit, welche sich mit dem Training dieser Struktur in Bezug auf die semantische Segmentierung auseinandersetzt ist [7]. Dabei wird im Rahmen des Trainings eines Cycle-Consistent Adversarial Networks für diese Auswertemethode im Rahmen von Domain Adaptation ein zusätzlicher Term zur Fehlerfunktion hinzugefügt, welcher die semantische Konsistenz berücksichtigt. Dazu wird ein neuronales Netz, welches auf dem Quelldomäne-Datensatz vortrainiert ist in das Training des Cycle-Consistent Adversarial Networks miteingebunden. Zur Verbesserung der unüberwachten Domain Adaptation bei Adversarial-Learning-basierten Methoden zur Bild-zu-Bild-Übersetzung wird in [8] ein Fehlerfunktionsterm für Cross-Domain-Consistency hinzugefügt, welche zur Verbesserung der semantischen Segmentierung verwendet werden kann. Das Prinzip der unüberwachten Domain Adaptation basierend auf Cycle-Consistent Adversarial Networks wird in [9] auf Bilddaten zur pixelweisen Klassifikation angewandt. Dabei wird ein Transfer zwischen zwei Luftbilddatensätzen der International Society for Photogrammetry and Remote Sensing (ISPRS) durchgeführt. Dazu wird ein neuronales Netz auf einem Datensatz als Quelldomäne vortrainiert und anschließend mit einem Cycle-Consistent Adversarial Network auf den anderen Datensatz

angepasst. In [10] kommt für den Anwendungsfall der Bodenbedeckungsklassifikation ein Category-Sensitive basiertes Domain Adaptation Verfahren zum Einsatz, bei welchem ein Geometry-Consistent Generative Adversarial Network mit einem Co-Training Adversarial Learning Network kombiniert wird. Diese Methode wurde auf zwei Luftbilddatensätzen ausgewertet. Ein weiterer Ansatz zur unüberwachten Domain Adaptation wird in [11] umgesetzt, indem ein neuer Regulierungsterm zur Fehlerfunktion zum Diskriminator des Adversarial-Trainings hinzugefügt wird, ebenso wie eine Entropie-basierte Parameterselektion.

### 1.2 Segmentierung und Klassifikation von Vegetation

Ein wichtiger Anwendungsfall im Bereich der Bodenbedeckungsklassifikation ist die Auswertung von Daten urbaner Gebiete. Heutzutage lebt die Hälfte der Weltbevölkerung in Städten und in den nächsten Jahren wird diese Tendenz weiter zunehmen. Diese Veränderung des Lebensraums birgt allerdings auch einige Herausforderungen und dementsprechend ist es wichtig, den daraus resultierenden Problemen für die Menschen und die Umwelt frühzeitig und effektiv zu begegnen. In diesem Zusammenhang spielen im Rahmen der Stadtentwicklung und -planung der Ausbau und die Pflege von Vegetation eine zentrale Rolle. Negative Folgen einer starken Urbanisierung können unterschiedliche Erscheinungsformen haben. Durch den großflächigen und dichten Bau von Gebäuden entwickelt sich innerhalb von urbanen Gebieten ein sogenanntes Stadtklima [12]. Aufgrund seiner regional begrenzten Erscheinung stellt es ein Mesoklima dar. Die Folge dessen ist, dass sich im Vergleich zum Umland eine zum Teil deutlich höhere Temperatur in der Stadt feststellen lässt. Dies hängt damit zusammen, dass aufgrund der eingesetzten Baumaterialien Wärme besser gespeichert wird [13] und aufgrund der Anordnung der Gebäude und der zum Teil hohen Bebauung in Bodennähe auch durch Wind wenig Abkühlung stattfinden kann [14]. Dieser Effekt wird als städtische Wärmeinsel (Urban Heat Island Effect) bezeichnet und beeinträchtigt vor allem in warmen Jahreszeiten die Lebensqualität vieler Menschen sehr. Eine zentrale Rolle, um diesem Effekt entgegenzuwirken, ist das Anlegen und Pflegen von Grünanlagen, da diese Wärme weniger speichern und bei hohem Bewuchs zusätzlich noch Schatten liefern können [15] [16]. Ein weiteres Problem, mit welchem sich viele Städte auseinandersetzen müssen, ist das immer häufigere Auftreten von

Überschwemmungen. Da unter anderem durch den Straßenbau viele versiegelte Flächen in urbanen Gebieten vorhanden sind, existieren nur wenige Flächen, welche Regenwasser in diesen Situationen abfließen lassen können. Durch das gezielte Anlegen von Vegetation kann allerdings die Möglichkeit geschaffen werden, Wasser in größeren Mengen aufzunehmen und dieses anschließend durch Verdunstung wieder abzugeben [17]. Durch die große und dichte Menge an Menschen, die in einer Stadt leben existiert außerdem ein hohes Verkehrsaufkommen und es entstehen dementsprechend Emissionen, welche die Luftqualität mindern. Durch eine gezielte Begrünung von Straßenzügen und Plätzen kann dieser Effekt abgemildert werden, da Pflanzen in der Lage sind diese Schadstoffe zu binden und dementsprechend der Luft zu entziehen [18]. Weiterhin kann durch die innerstädtische Vegetation eine gewisse Biodiversität entstehen und aufrecht erhalten bleiben [19] [20]. Für die Erfassung von Vegetation und der Bestimmung ihres Zustands kann bei großflächigen Gebieten, wie z.B. Wäldern auf Satellitendaten zurückgegriffen werden wie MODIS [21] oder Sentinel-2 [22] mit einer vergleichsweise geringen Auflösung, da in der Regel eine grobe Zustandsermittlung ausreichend und nicht die Auswertung einzelner Pflanzen relevant ist. Da die Vegetationsflächen in urbanen Gebieten allerdings stark fragmentiert sind, gibt es im Zuge der Stadtplanung und der Pflege der Pflanzen einige Herausforderungen. Die einzelnen Grünflächen sind zum Teil sehr klein oder bestehen nur aus einzelnen räumlich nicht direkt zusammenhängenden Objekten, wie zum Beispiel einzelnen Bäumen auf Plätzen oder neben Straßen. Somit ist es notwendig für die Erfassung von Vegetation in urbanen Gebieten auf hochauflösende Bilder aus Luft- oder Satellitendaten zurückzugreifen. Auf Grundlage dieser Daten können die einzelnen Vegetationsbereiche automatisch erfasst und deren Entwicklung kontrolliert werden um bei negativen Veränderungen bereits frühzeitig Maßnahmen ergreifen zu können. Zu diesem Zweck kann mit Hilfe von semantischer Segmentierung bei hochauflösenden Bildern eine genaue Eingrenzung der räumlichen Ausdehnung ermittelt werden und im Anschluss eine Bewertung des Vegetationszustands auf Basis von biophysikalischen spektralen Zusammenhängen in den Bilddaten durchgeführt werden. Ein typisches Spektrum von Vegetation im Falle einer NIR-RGB-Aufnahme ist in Abbildung 3 gegeben. Eine Methode zur Segmentierung von Vegetation auf Luft- und Satellitenbilddaten ist die bereits seit Jahrzehnten verwendete Berechnung von Vegetationsindices. Dabei kommt vor allem der Normalized Difference Vegetation Index (NDVI) [23] zum Einsatz [24] [25] [26]. Allerdings findet dabei

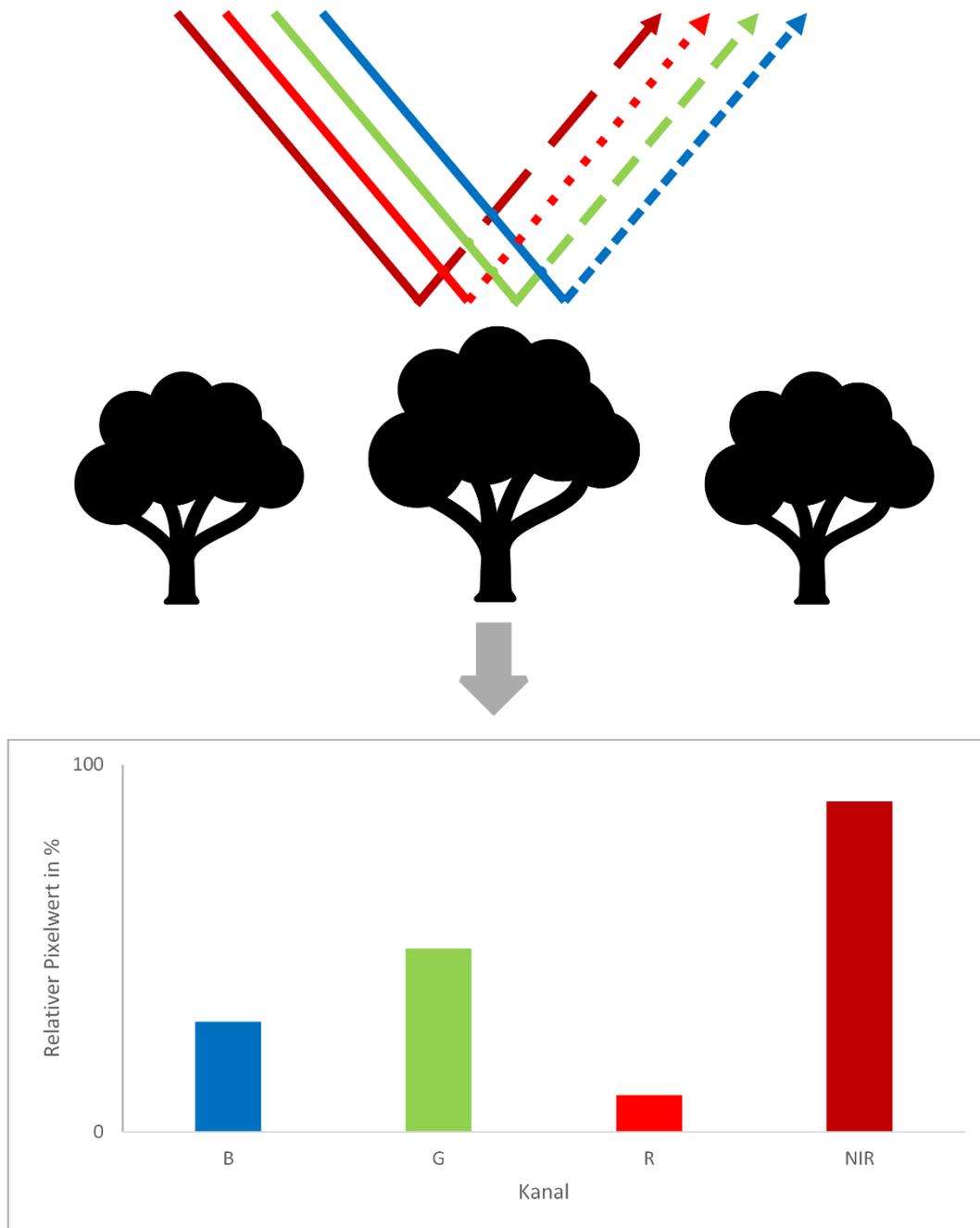


Abbildung 3: Biochemische Prozesse führen bei Vegetation zu bestimmten Absorptions- und Reflexionscharakteristika und stellen sich dementsprechend in den einzelnen Farbkanäle von Sensoren dar und bilden ein entsprechendes spezifisches Spektrum im Pixelwertebereich.

ausschließlich eine auf spektralen Eigenschaften beruhende pixelweise Klassifikation statt und räumliche Informationen werden nicht berücksichtigt. Da sich allerdings gerade in urbanen Gebieten auch Objekte befinden können, welche ähnliche spektrale Eigenschaften wie Pflanzen haben, wäre es von Vorteil auch räumliche Informationen in eine

entsprechende Analyse miteinzubeziehen. Denn die spektralen Eigenschaften sind manchmal von unterschiedlichen anderen Faktoren abhängig und eine ergänzende Auswertung mit räumlichen Komponenten kann diesem Problem entgegenwirken. Um diesem Umstand zu begegnen kann die Segmentierung von Vegetationsflächen auf Luft- oder Satellitenbildaufnahmen in urbanen Gebieten bei Daten mit sehr hoher Auflösung basierend auf einem mehrstufigen Prozess durchgeführt werden, welcher neben spektralen auch räumliche Informationen verarbeitet wie beispielsweise Verfahren wie Object-Based Image Analysis (OBIA). Dazu werden Pixel räumlich innerhalb des Bildes zu einzelnen Objekten verbunden und anschließend werden ebendiese den entsprechenden Objektklassen zugeordnet [27]. In [28] wurde eine Bodenbedeckungsklassifikation basierend auf hochauflösten Satellitenbildaufnahmen mit Daten von IKONOS durchgeführt. Die Bilder bestanden aus Vierkanal-Aufnahmen mit RGB und nahem Infrarot (NIR-RGB) und es wurde zusätzlich der Soil-Adjusted Vegetation Index (SAVI) berechnet. Die Auswertung der Daten erfolgte auf dem Vorgehen von OBIA. Bei der Arbeit [29] wurde anhand von hochauflösenden Satellitendaten von IKONOS im Vergleich zu WorldView-2 untersucht, wie eine Auswertung von Vegetation anhand von Bäumen im urbanen Gebiet durchgeführt werden kann. Dabei kamen sowohl Vier-Kanalbilder wie auch Acht-Kanalbilder als Datengrundlage zum Einsatz. Es wurde in beiden Fällen gezeigt, dass eine ungefähr doppelt so hohe Auflösung deutlich zu einer Verbesserung der Auswertergebnisse führt. In [30] wurde eine Segmentierung und Klassifizierung von Vegetation auf hochauflösenden Luftbildaufnahmen im urbanen Raum durchgeführt. Zum Einsatz kamen dabei neben RGB ebenfalls auch der Kanal des nahen Infrarots. Auf Grundlage des Vier-Kanalbildes, des daraus resultierenden NDVI und der Anwendung der Principal Component Analysis (PCA) wurde anschließend eine Kombination aus mehreren Segmentierungsmethoden eingesetzt und eine entsprechende Klassifizierung durchgeführt wobei mehrere Parameter manuell eingestellt wurden. In den letzten Jahren gab es im Bereich der Fernerkundung erste vielversprechende Ansätze basierend auf CNNs zur Klassifikation und Segmentierung von Vegetation [31]. In [32] wurde ein auf U-Net basiertes Convolutional Neural Network für die semantische Segmentierung im Bereich Vegetation eingesetzt um hochauflösende RGB-Daten zu klassifizieren. Dabei wurden einzelne Pflanzenstrukturen erkannt und segmentiert.

### 1.3 Motivation und Beiträge der Arbeit

In dieser Arbeit werden Methoden zur Verbesserung von Domain Adaptation im Bereich der semantischen Segmentierung für die Bodenbedeckungsklassifikation von urbanen Gebieten mit einem Schwerpunkt auf Vegetation vorgestellt. Das Ziel ist es den notwendigen Einsatz von annotierten Daten in der Zieldomäne zu reduzieren bzw. die Qualität der Auswertergebnisse zu erhöhen. Zum einen werden in diesem Zusammenhang die eigentlichen Methoden zur semantischen Segmentierung analysiert und angepasst und zu anderen auch die Datengrundlage an sich. Im Bereich der Segmentierung von Vegetation wurde außerdem untersucht, welche Informationen (räumlich und spektral) für die Auswertung basierend auf Convolutional Neural Networks relevant sind. Weiterhin wurde evaluiert, inwiefern spektrale Eigenschaften basierend auf biophysikalischen Zusammenhängen relevant für die Segmentierung von Vegetation sind und ob diese bei dem Einsatz von Domain Adaptation erhalten bleiben. Untersucht wurden im Rahmen dieser Arbeit sowohl Anwendungen aus dem Bereich der überwachten wie auch der unüberwachten Domain Adaptation für die semantische Segmentierung. Eine Übersicht ist in Abbildung 4 gegeben.

Im Bereich der überwachten Domain Adaptation wurde evaluiert, ob sich bei Anwendung von Encoder-Decoder-Netzwerken, welche auf umfangreichen Quelldomäne-Datensätzen trainiert wurden, eine Verbesserung der Anpassung an die Zieldomäne erreichen lässt, indem die trainierte Netzstruktur vor dem Domain Adaptation-Prozess an sich erweitert wird, anstelle das gesamte neuronale Netz auf dem Zieldomäne-Datensatz weiter zu trainieren. Dazu wurden zwischen den Encoder und Decoder Adaptionmodule integriert, welche die Encoder-Ausgabe basierend auf der Zieldomäne dementsprechend verändern, dass für den Decoder eine der Quelldomäne entsprechende Feature-Repräsentation vorliegt. Die Methode und Ergebnisse wurden bereits in [33] veröffentlicht.

Im Rahmen der unüberwachten Domain Adaptation wurde basierend auf der Grundstruktur des Encoder-Decoder-Segmentierungsnetzwerks untersucht, inwiefern sich durch Cycle-Consistent Adversarial Networks ein Style-Transfer zwischen Quell- und Zieldomäne realisieren und in Hinblick auf die Aufgabenstellung der semantischen Segmentierung

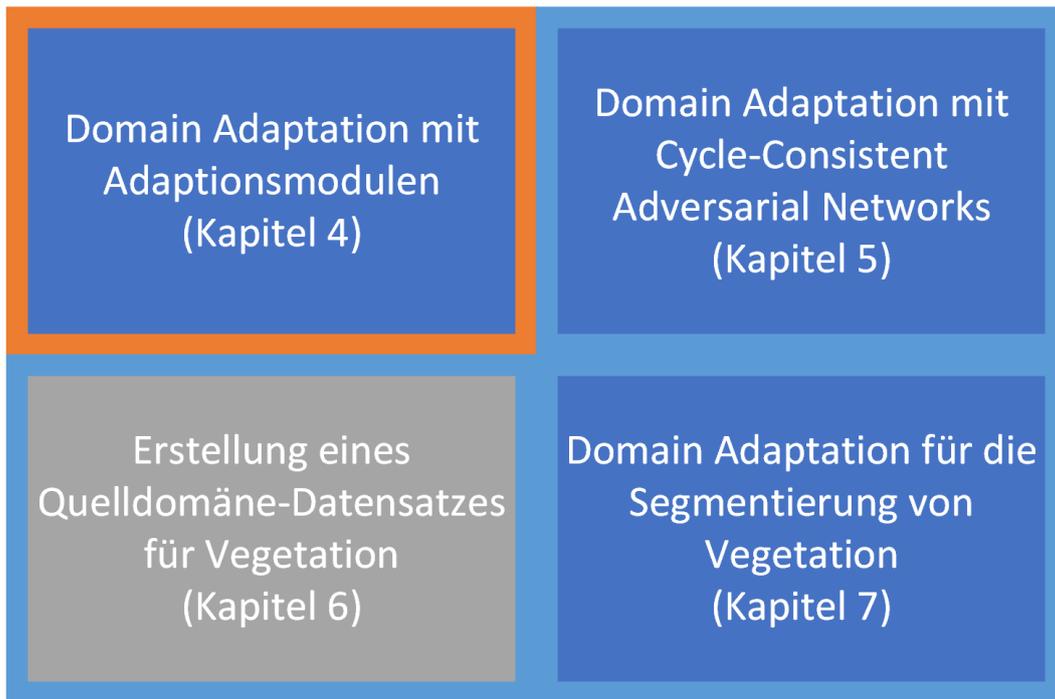


Abbildung 4: Blau hinterlegt: Methoden zur Domain Adaptation. Grau hinterlegt: Methode zur Quelldomäne-Datensatzerstellung. Roter Rahmen: Überwachtes Lernen (Annotationen in Zieldomäne vorhanden). Blauer Rahmen: Unüberwachtes Lernen (Annotationen in Zieldomäne nicht vorhanden).

verbessern lässt. Dazu wurde das Training des Cycle-Consistent Adversarial Networks kombiniert mit dem Training des Segmentierungsnetzwerks wobei ein kontinuierliches Trainingschema entwickelt und angewandt wurde. Dieses Vorgehen konnte eine höhere Generalisierung der Methode ermöglichen und dementsprechend die Auswertergebnisse verbessern. Die Methode und Ergebnisse wurden bereits in [34] veröffentlicht.

Für die unüberwachten Domain Adaptation der Objektklasse Vegetation wurde zunächst in Hinblick auf eine teilautomatisierte Annotation der entsprechende Quelldomäne-Datensatz untersucht. Dazu wurden basierend auf Vegetationsindices mit einem entsprechenden Schwellwert eine erste Segmentierung durchgeführt und auf Grundlage einer Kombination aus Bildverarbeitungsmethoden und dem Einsatz von Convolutional Neural Networks eine Verbesserung dieser erreicht. Dabei konnte gezeigt werden, dass für den entsprechenden Quelldomäne-Datensatz auf Grundlage der spektralen Informationen, welche durch die Vegetationsindices gegeben sind, sich eine Gewinnung von räumlichen Informationen umsetzen ließ, welche von den Convolutional Neural Networks erlernt werden konnten.

Basierend auf dem Training des Cycle-Consistent Adversarial Networks zur unüberwachten Domain Adaptation wurde nun eine Anpassung der Fehlerfunktion durchgeführt um die spezifischen biophysikalisch-spektralen Eigenschaften von Vegetation in den Bildern während des Transfers beizubehalten. Dabei konnte gezeigt werden, dass diese relevant für die Auswertung basierend auf dem Segmentierungsnetzwerk sind und das Ergebnis verbessert werden konnte. Die Methode und Ergebnisse wurden bereits in [35] veröffentlicht.

Die Arbeit gliedert sich in mehrere Kapitel. Zunächst wird in Kapitel 2 auf die Grundlagen zu künstlichen neuronalen Netzen und den eingesetzten Metriken zur Auswertung eingegangen. In Kapitel 3 werden die in dieser Arbeit eingesetzten Datensätze zusammengefasst. In Kapitel 4 wird die Methode zu überwachten Domain Adaptation basierend auf Adaptionmodulen vorgestellt. In Kapitel 5 wird auf die Methode zur unüberwachten Domain Adaptation basierend auf Cycle-Consistent Adversarial Networks eingegangen, wobei ein kontinuierliches Trainingschema zur Verbesserung der Auswerteergebnisse vorgestellt wird. In Kapitel 6 wird anhand eines neu erstellten Luftbilddatensatzes ein Schema zur Unterstützung der Erstellung eines Quelldomäne-Datensatzes für die Segmentierung von Vegetation angewandt, welches auf der Kombination von Vegetationsindices und Convolutional Neural Networks beruht. In Kapitel 7 wird das Verfahren erläutert, welches den Ansatz der Cycle-Consistent Adversarial Networks für eine unüberwachte Domain Adaptation verwendet, um mit einer auf die charakteristischen biophysikalisch-spektralen Eigenschaften von Vegetation angepasste Fehlerfunktion, einen besseren Transfer zu ermöglichen. Kapitel 8 beinhaltet abschließend eine zusammenfassende Diskussion der Ergebnisse und einen Ausblick.

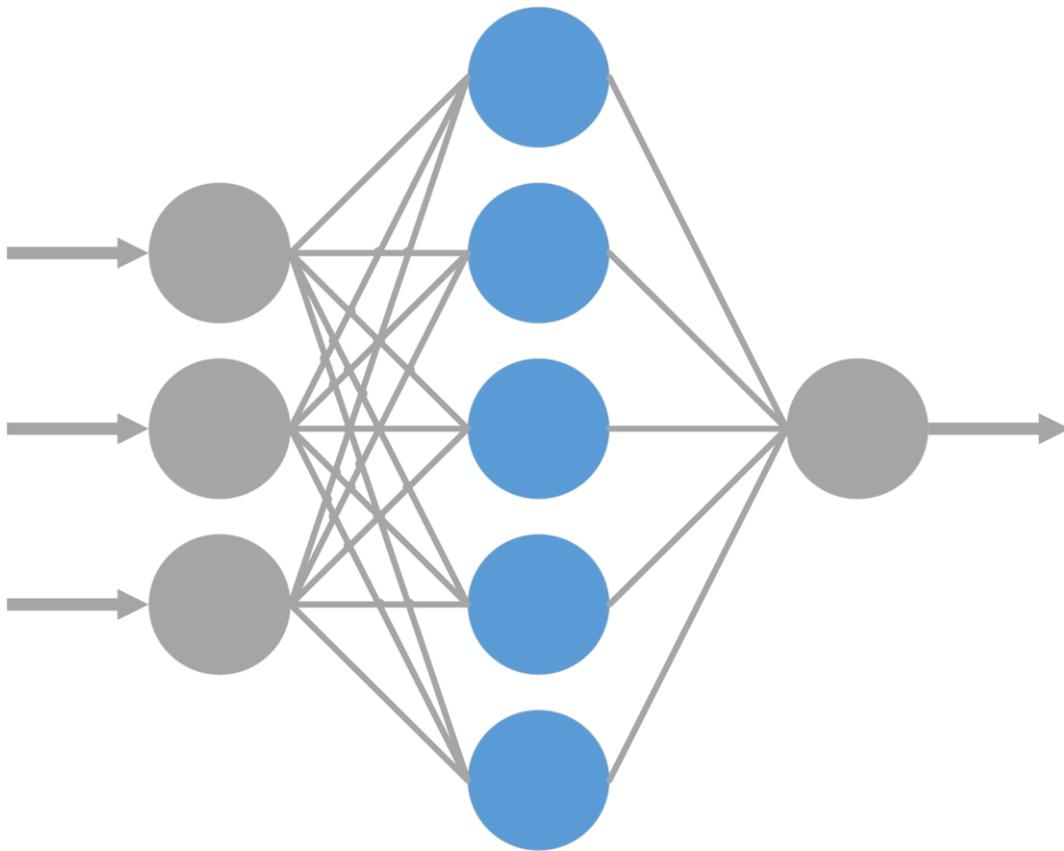


## 2 Grundlagen

In diesem Kapitel wird auf die Grundlagen des Aufbaus und Trainings künstlicher neuronaler Netze eingegangen, sowie die Struktur und Funktionsweise von Convolutional Neural Networks erläutert. Dabei werden die unterschiedlichen eingesetzten Schichten erläutert und die entsprechende Zusammensetzung dieser zu Netzwerken. Zusätzlich werden noch die für diese Arbeit relevanten Metriken zur Auswertung beschrieben.

### 2.1 Künstliche neuronale Netze

Bei künstlichen neuronalen Netzen handelt es sich um ein System aus künstlichen Neuronen, welche sich zu einem Netzwerk zusammensetzen. Diese Struktur ist dabei dem biologischen Vorbild der Vernetzung von Neuronen im Nervensystem von Lebewesen nachempfunden [36]. Es gibt künstliche neuronale Netze mit unterschiedlichsten Architekturen, welche als Graph beschrieben werden können. Dabei werden die Neuronen als Knoten und die Verbindungen zwischen diesen als Kanten dargestellt. Diese bestehen aus einer Eingabeschicht, gegebenenfalls mehreren verdeckten Schichten und einer Ausgabeschicht. In der einfachsten Form besteht ein neuronales Netz neben der Eingabeschicht nur aus einer Schicht, welche auch gleichzeitig die Ausgabeschicht darstellt. Meistens sind neuronale Netze aber mehrschichtig und besitzen mindestens eine verdeckte Schicht. In Abbildung 5 ist beispielhaft ein neuronales Netz mit einer Eingabeschicht, einer einzelnen verdeckten Schicht und einer Ausgabeschicht dargestellt. Bei der Art der Verbindung der einzelnen Schichten wird bei künstlichen neuronalen Netzen zwischen vorwärtsgerichteten (Feedforward) und rekurrenten Netzwerken unterschieden. Wenn eine einzelne Schicht immer nur mit der darauffolgenden Schicht verbunden ist, werden diese als Feedforward-Netze bezeichnet. Es findet dementsprechend keine Rückführung von Informationen an vorherige Neuronen statt. Falls in einem neuronalen Netzwerk auch rückwärtsgerichtete Kanten vorhanden sind, dementsprechend eine Rückkopplung stattfindet, wird von einem rekurrenten Netz gesprochen [37]. Das künstliche Neuron stellt die Grundlage für das Konzept künstlicher neuronaler Netze dar. Dabei wird ein Netzwerk aus künstlichen Neuronen gebildet, welches in der Lage ist beliebig komplexe Funktionen zu



*Abbildung 5: Schematische Darstellung eines neuronalen Netzes mit einer Eingabeschicht bestehend aus drei Neuronen auf der linken Seite, einer verdeckten Schicht mit fünf Neuronen in der Mitte und einer Ausgangschicht, bestehend aus einem einzelnen Neuron auf der rechten Seite. Alle Neuronen einer jeweiligen Schicht sind mit allen Neuronen der vorherigen und darauffolgenden Schicht verbunden.*

approximieren. Entsprechend dem biologischen Vorbild einer Nervenzelle kann das künstliche Neuron mehrere Eingaben verarbeiten und auf Grundlage der Aktivierung Informationen weitergeben. Das erste Konzept eines künstlichen Neurons geht zurück auf Warren McCulloch und Walter Pitts, welche bereits 1943 ein Neuronenmodell vorgeschlagen haben. Dieses kann als Ausgabe nur eine 0 oder 1 erzeugen und entsprechend dem biologischen Vorbild hemmende Signale verarbeiten. Die Eingangssignale werden addiert und anschließend mit einem Schwellwert verglichen. Falls die Summe den Schwellwert überschreitet, wird von dem Neuron der Wert 1, ansonsten der Wert 0 zurückgegeben [38]. Ein künstliches Neuron erhält mehrere Eingaben  $x_i$  wie in Abbildung 6 dargestellt. Dabei wird jeder Eingabe ein Gewicht  $w_{ji}$  zugeteilt. Diese Gewichte bestimmen wie stark die entsprechende Eingabe  $i$  die Ausgabe des Neurons  $j$  beeinflusst. Auf Grundlage der

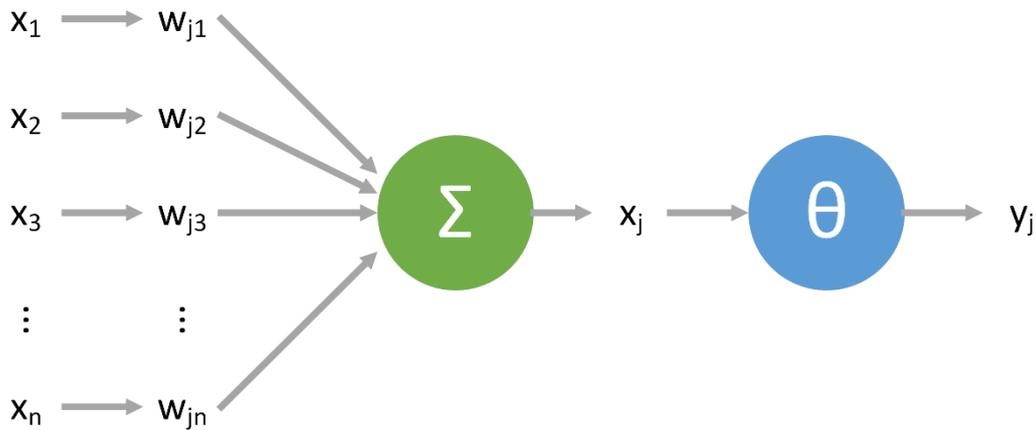


Abbildung 6: Funktionsweise eines künstlichen Neurons mit mehreren Eingangssignalen  $x_i$ , Gewichten  $w_{ji}$ , einer Summenfunktion  $\Sigma$ , der Summe der Netzeingabe  $x_j$ , der Aktivierungsfunktion  $\Theta$  und der entsprechenden Ausgabe des Neurons  $y_j$ .

Gewichte wird anschließend basierend auf einer Summenfunktion  $\Sigma$  die Netzeingabe des Neurons berechnet:

$$x_j = \sum_i x_i \cdot w_{ji}$$

Die Ausgabe des Neurons wird daraufhin durch das Anwenden der entsprechenden Aktivierungsfunktion  $\theta$  auf die Netzeingabe bestimmt durch:

$$y_j = \theta(x_j)$$

Das Anwenden der Aktivierungsfunktion ist hierbei insofern wichtig, um Nichtlinearität in das künstliche neuronale Netz zu integrieren, da sich nicht alle Aufgaben durch lineare Funktionen abbilden lassen [37] [39].

## 2.2 Backpropagation

Bei dem Backpropagation-Algorithmus (Fehlerrückführung) handelt es sich um ein Lernverfahren, welches für künstliche neuronale Netze eingesetzt werden kann. Es ist dabei möglich, diesen auf mehrschichtige Netze anzuwenden und er zählt zu den überwachten Lernverfahren. D.h., es muss für jede Eingabe auch ein entsprechender Zielwert vorhanden sein. Mit Hilfe dieser Zielwerte wird die Größe des Fehlers ermittelt und die Gewichte des Netzes können angepasst werden. Zunächst werden mit der Fehlerrückführung dazu

die Gradienten bestimmt, um diese anschließend auf Basis eines Optimierungsverfahrens anzupassen. Das Ziel ist die Minimierung der Fehlerfunktion, wobei in der Regel allerdings nur ein lokales Minimum gefunden wird. Zunächst wird eine Eingabe in das Netz gegeben und auf Grundlage der Verbindungen der Neuronen, der Summen- und Aktivierungsfunktionen eine Ausgabe ermittelt. Anschließend wird mit einer Fehlerfunktion  $L$  die Abweichung zu einem Sollwert ermittelt. Dabei gibt es je nach Anwendung unterschiedliche Varianten. Auf Grundlage dieses Fehlers können nun die Gewichte angepasst werden. Das Prinzip der Fehlerrückführung beruht darauf, den Gradienten schichtweise zu berechnen, was wiederum durch die partielle Ableitung der Fehlerfunktion über die Kettenregel passiert:

$$\frac{dL}{dw_{ji}} = \frac{dL}{dy_j} \cdot \frac{dy_j}{dx_j} \cdot \frac{dx_j}{dw_{ji}}$$

Daraus kann nun folgender Zusammenhang für die Änderung der Gewichte berechnet werden:

$$\Delta w = -\epsilon \frac{dL}{dw_{ji}}$$

mit  $\epsilon$  als Lernrate, welche angibt wie groß die Schritte sein sollen in denen sich dem Minimum genähert werden soll. Da entsprechend ein Minimum gesucht wird, und somit ein Gradientenabstieg vorgenommen wird, hat der Term ein negatives Vorzeichen. Abschließend werden die Gewichte entsprechend aktualisiert:

$$\Delta w_{neu} = -\epsilon \frac{dL}{dw_{neu}} + \Delta w_{alt}$$

Dieses Verfahren wird so lange durchgeführt bis Konvergenz zu einem meist lokalen Minimum eintritt. Vor dem Beginn des Lernalgorithmus werden die Gewichte initial durch zufällige Werte belegt [39] [40].

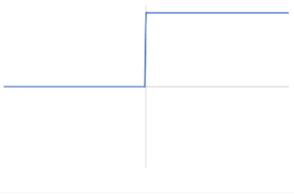
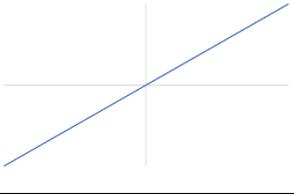
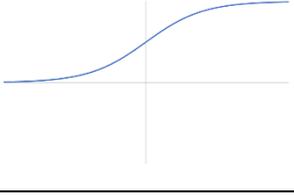
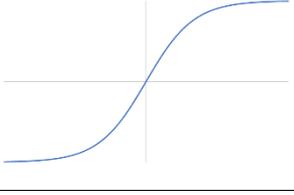
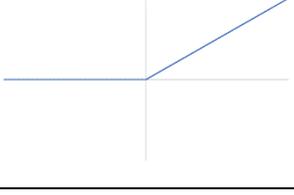
### 2.3 Aktivierungsfunktionen

Als Aktivierungsfunktion können unterschiedliche Funktionstypen gewählt werden. Die einfachste Form ist die Schwellwertfunktion, welche dem Neuronenmodell nach Warren McCulloch und Walter Pitts entspricht. Diese nimmt nur den Wert 1 für Eingaben  $x \geq 0$

und ansonsten 0 an. Des Weiteren gibt es noch die lineare Funktion bzw. auch die stückweise lineare Funktion, welche nur auf einem begrenzten Intervall bei  $-0,5 < x < 0,5$  linear abbildet. Ansonsten ist die Funktion für Werte von  $x \geq 0,5$  bei 1 und von  $x \leq -0,5$  bei 0. Häufig eingesetzte Aktivierungsfunktionen sind ebenfalls die sogenannten sigmoide Funktionen. Eine spezielle Eigenschaft dieser Funktion ist die Differenzierbarkeit, welche für den Backpropagation-Algorithmus benötigt wird. Eine weitere häufig verwendete Aktivierungsfunktion ist der Tangens Hyperbolicus, welcher eine ähnliche Form wie die Sigmoid-Funktion besitzt und Werte zwischen  $-1$  und  $1$  annehmen kann. Sie ist ebenfalls differenzierbar. Ein allgemeines Problem beider Funktionen ist das Verschwinden des Gradienten, da die beiden Funktionen bei  $0$  und  $1$  bzw. bei  $-1$  und  $1$  sättigen. Dementsprechend wird der Wert der Ableitung fast  $0$  und es gibt keinen Gradienten, welcher für die Fehlerrückführung verwendet werden könnte. Außerdem bestehen sowohl die Sigmoid-Funktion als auch der Tangens Hyperbolicus aus Exponentialoperatoren, welche sehr rechenaufwändig sind. Die Rectified Linear Unit (ReLU) ist eine Aktivierungsfunktion, welche stark an das biologische Vorbild angelehnt ist und als Positivteil seines Arguments definiert ist. D.h., dass die Funktion für  $x < 0$  den Wert  $0$  und für  $x \geq 0$  den Eingabewert zurückgibt. Die Funktion ist überall differenzierbar, außer bei dem Wert  $0$ . Die ReLU hat sich als vorherrschende Aktivierungsfunktion für Anwendungen im Deep Learning im Bereich Computer Vision herausgestellt [37] [41]. In Tabelle 1 sind unterschiedliche Aktivierungsfunktionen dargestellt.

## Grundlagen

Tabelle 1: Übersicht über unterschiedliche Aktivierungsfunktionen die im Bereich neuronaler Netze zum Einsatz kommen.

|                      |   |  |
|----------------------|---|--|
| Stufenform           | $x \mapsto \begin{cases} 1 & : x \geq 0 \\ 0 & : x < 0 \end{cases}$ |    |
| Linear               | $x \mapsto x$   |    |
| Sigmoid              | $x \mapsto \frac{e^x}{1 + e^x}$                                     |    |
| Tangens Hyperbolicus | $x \mapsto \frac{e^x - e^{-x}}{e^x + e^{-x}}$                       |   |
| ReLU                 | $x \mapsto \begin{cases} x & : x \geq 0 \\ 0 & : x < 0 \end{cases}$ |  |

### 2.4 Convolutional Neural Networks

Bei einem Convolutional Neural Network (CNN) handelt es sich um ein künstliches neuronales Netz, welches besonders häufig im Bereich der Computer Vision eingesetzt wird. Das Grundprinzip dieses neuronalen Netzwerks beruht auf dem Anwenden zahlreicher Faltungsmatrizen (Convolutional Matrix). Die Architektur der CNNs geht dabei zurück auf eine Struktur namens Neocognitron aus dem Jahre 1980 [42]. Später hat sich gezeigt, dass das Training von CNNs effektiv mit dem Backpropagation-Algorithmus durchgeführt werden kann [40]. Seitdem werden CNNs vor allem in den Bereichen der Bildklassifikation [43] [44] [45] [46], bei welcher einzelne Bilder als Ganzes klassifiziert werden, der semantischen Segmentierung [47] [48] [49] [50], bei welcher eine pixelweise Klassifikation durchgeführt wird und der Objektdetektion [51] [52] [53], wo das Ziel ist Objekte zu erkennen und mit einer Bounding Box zu umranden, eingesetzt. Die wesentlichen Eigenschaften, die ein CNN hat sind: Lokale Konnektivität, Geteilte Gewichte und Pooling-Schichten.

Ein klassisches CNN besteht prinzipiell aus einer oder mehreren Convolutional-Schichten und einer darauffolgenden Pooling-Schicht. Dieses Element kann dabei grundsätzlich beliebig häufig wiederholt werden und bildet so ein tiefes neuronales Netz (Deep Learning). Die Neuronen eines CNNs sind als Feature Maps organisiert, wobei jede dieser lokal mit der vorherigen Feature Map verbunden ist [54].

#### **Convolutional-Schicht**

Normalerweise stellt die Eingabe eine zwei- oder dreidimensionale Matrix dar und repräsentiert die Pixel eines Graustufen- oder Farbbildes. Die Neuronen in der Convolutional-Schicht sind dementsprechend angeordnet.

Durch eine diskrete Faltung wird die Aktivität eines Neurons ermittelt. Dafür wird nach und nach eine Faltungsmatrix über die Eingabe geschoben und das innere Produkt der Faltungsmatrix und des entsprechenden Bildausschnitts berechnet. Dabei reagiert ein Neuron in dieser Schicht somit auf die Aktivität der vorherigen Schicht in einer lokalen Umgebung, was wiederum dem biologischen Prinzip des rezeptiven Feldes entspricht. Im Gegensatz dazu werden bei vollverbundenen Neuronenschichten globale Muster erlernt.

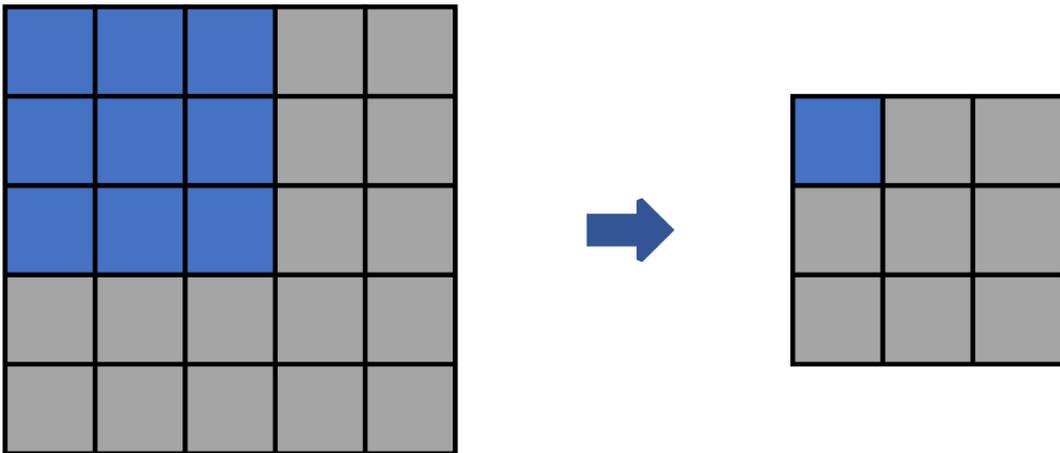


Abbildung 7: Schematische Darstellung der Funktionsweise einer Convolutional-Schicht mit einer Filter-Größe von  $3 \times 3$ .

Außerdem sind die Gewichte einer Convolutional-Schicht für alle Neuronen gleich, weshalb man dabei von geteilten Gewichten spricht. Aus diesem Umstand folgt die Translationsinvarianz als grundlegende Eigenschaft von CNNs. Das Resultat aus dem inneren Produkt wird nun mit der Aktivierungsfunktion zur Ausgabe des Neurons. Bei Convolutional-Schichten können unter anderem folgende Parameter angepasst werden:

- Größe der Faltungsmatrizen (Filter): Üblicherweise wird die Matrixgröße  $3 \times 3$  für CNNs gewählt (siehe Abbildung 7). Allerdings werden zur Veränderung des rezeptiven Feldes auch andere Größen eingesetzt.
- Anzahl der Faltungsmatrizen: Zusätzlich zur Größe der Faltungsmatrizen kann auch noch die Anzahl dieser in jeder Convolutional-Schicht bestimmt werden um eine höhere Komplexität des neuronalen Netzes zu ermöglichen.
- Aktivierungsfunktion: Bei CNNs wird in der Regel die ReLU-Aktivierungsfunktion verwendet. Es können aber auch andere Aktivierungsfunktionen eingesetzt werden.
- Schrittweite (Stride): Hierunter versteht man die Anzahl der Pixel, mit der sich die Faltungsmatrix über das Bild bewegt. Dadurch können Überlappungen vermieden und die Größe der Feature Map reduziert werden, also ein Downsampling durchgeführt werden.
- Padding: Bei Anwendung der Faltungsoperation verkleinert sich die Feature Map automatisch gegenüber der vorherigen um die äußeren Bildpunkte. Um diesen

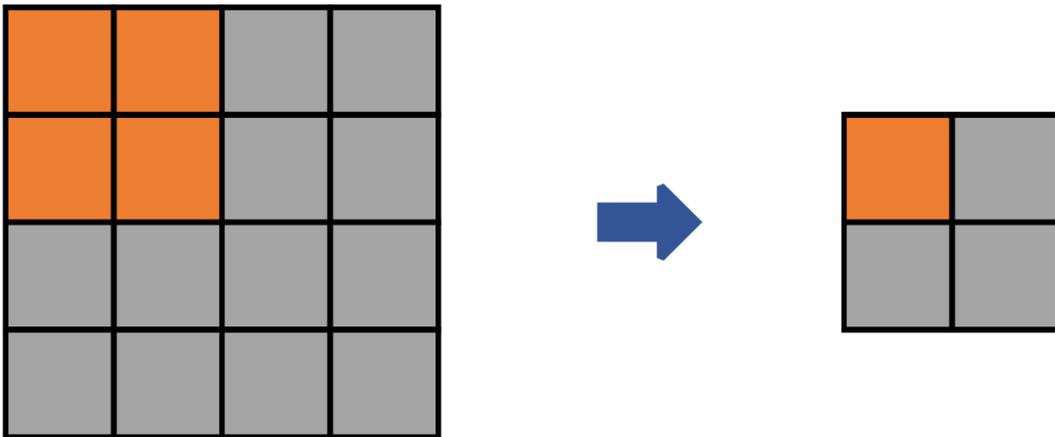


Abbildung 8: Schematische Darstellung der Funktionsweise einer Pooling-Schicht mit einer entsprechenden Fenstergröße von  $2 \times 2$ .

Umstand zu verhindern, können zusätzliche Zeilen und Spalten hinzugefügt werden, sodass die Größe der Feature Maps konstant bleibt und keine Informationen verloren gehen.

### **Pooling-Schicht**

Die Pooling-Schicht kann dazu eingesetzt werden, um semantisch ähnliche Informationen zusammenzufassen, da häufig die ungefähre Lokalisierung eines Features ausreichend ist. Dabei wird in der Regel bei einem  $2 \times 2$  großen Fenster aus Neuronen nur der Wert des aktivsten Neurons, also das mit dem höchsten Wert, für die folgenden Schichten weitergegeben (siehe Abbildung 8). Diese Art von Pooling wird dementsprechend Max-Pooling genannt und stellt die gängigste Form von Pooling-Schicht dar. Es wird entsprechend ein Downsampling der Feature-Map um den Faktor 2 durchgeführt. Der Vorteil dieser Art von Schicht ist, dass die Parameteranzahl verringert werden kann und das rezeptive Feld in Bezug auf die Eingabe erhöht wird, da die folgenden Convolutional-Schichten immer größere Fenster betrachten und somit räumliche Hierarchien hervorgebracht werden können.

### **Dropout-Schicht**

Bei dieser Schicht handelt es sich um ein Regularisierungsverfahren, mit dem Ziel eine Überanpassung des neuronalen Netzes zu verhindern. Dabei wird während des Trainings ein gewisser Prozentsatz an zufällig ausgewählten Neuronen deaktiviert um eine Co-Anpassung zu verhindern [55].

### Batch Normalization-Schicht

Die Batch Normalization-Schicht stellt ebenfalls eine Regularisierungsmethode dar, bei welcher die Eingabewerte normalisiert werden, um das Training sowohl schneller als auch stabiler zu machen [56].

### Softmax-Schicht

In der letzten Schicht des CNNs, welche die Ausgabe erzeugt, wird für Klassifikationsaufgaben in der Regel die Softmax-Aktivierungsfunktion eingesetzt. Die Softmax-Funktion transformiert einen Vektor  $z$  mit  $K$  realen Zahlen in einen normalisierten Vektor  $\sigma(z)$  mit einem Wertebereich von  $(0,1)$ , wobei sich alle Werte zu 1 aufsummieren:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

mit  $i = 1, \dots, K$ . Dies kann für eine kategoriale Verteilung genutzt werden und bei CNNs in Zusammenhang mit der kategorische Kreuzentropie eingesetzt werden. Diese berechnet sich durch:

$$L = - \sum_{i=1}^K p_i \log(q_i)$$

mit  $K$  als Anzahl an Objektklassen,  $p_i$  als tatsächliche Klasse und  $q_i$  als Softmax-Ausgabe der Klasse  $i$ , sodass für jede mögliche Objektklasse eine Wahrscheinlichkeit ausgegeben werden kann.

### Aufbau eines CNNs

Der Aufbau eines typischen CNNs soll hier am Beispiel des VGG16-Klassifikationsnetzes dargestellt werden [44]. Dieses besteht aus einer Bildeingabeschicht, fünf Convolutional-Blöcken, drei vollverbundenen Schichten und einer abschließenden Ausgabeschicht. Die Convolutional-Blöcke bestehen jeweils aus zwei (Block 1 und 2) oder drei Convolutional-Schichten (Block 3 bis 5), gefolgt von einer Pooling-Schicht. Die Abfolge an entsprechenden Convolutional-Schichten generiert immer komplexere Features, während die Pooling-Schichten die Größe der Feature Maps kontinuierlich von einer ursprünglichen Bildeingangsgröße von  $224 \times 224$  Pixeln reduzieren. Die Anzahl an Faltungsmatrizen in den einzelnen Convolutional-Blöcken nimmt dabei zu (siehe Tabelle 2). Im Anschluss an die

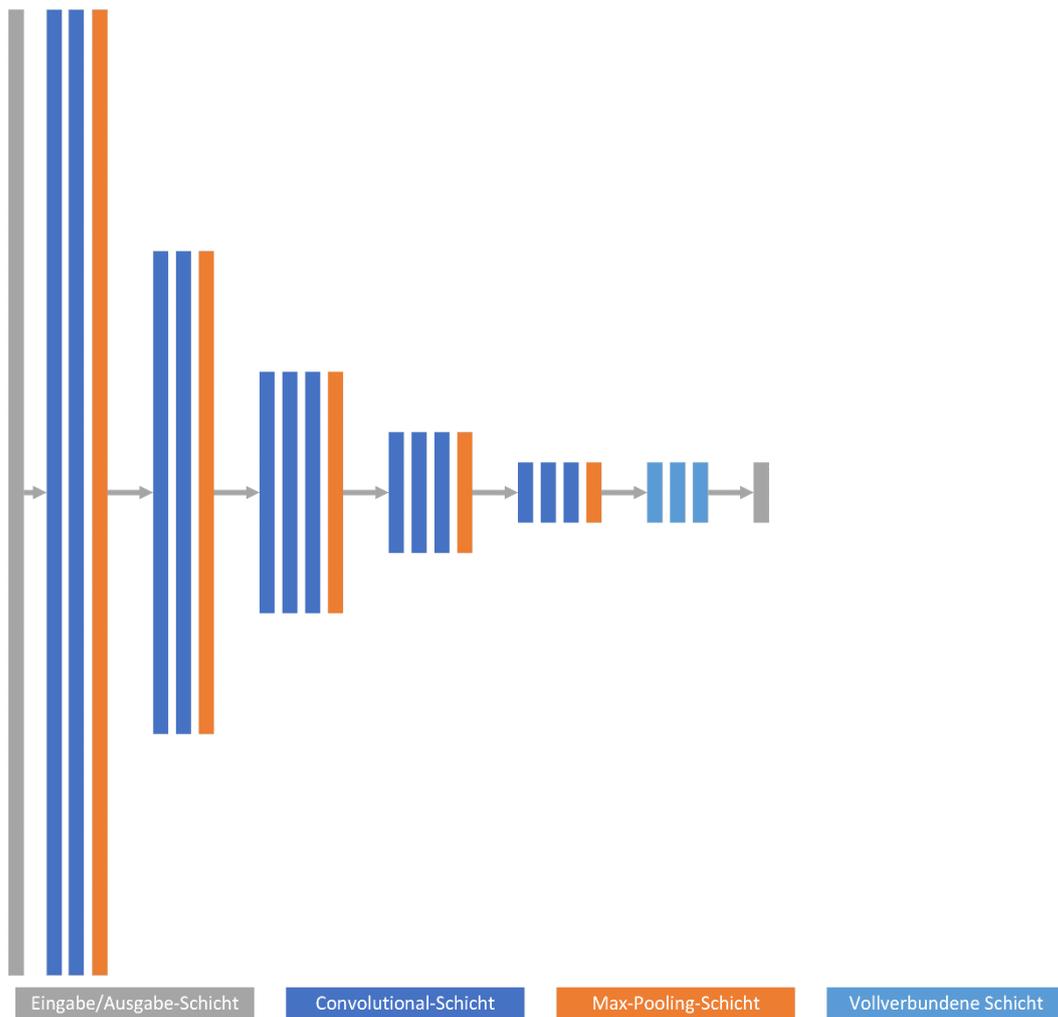


Abbildung 9: Schematische Darstellung des VGG16-Klassifikationsnetzwerks, bestehend aus fünf Convolutional-Blöcken mit Convolutional-Schichten und Pooling-Schichten zur Generierung von Features sowie drei vollverbundenen Schichten für die anschließende Klassifikation.

Convolutional-Blöcke werden noch drei vollverbundene Schichten eingesetzt. Diese sind für die Klassifikationsaufgabe eingebunden und bestehen jeweils aus 4095 Neuronen. Abschließend kommt noch eine Ausgabeschicht zum Einsatz, welche die entsprechende Objektkategorie zu dem Eingabebild ausgibt. Der Aufbau des VGG16-Klassifikationsnetzwerks ist in Abbildung 9 dargestellt.

Tabelle 2: Aufbau eines VGG16-Klassifikationsnetzwerks.

| Convolutional-Block | Feature Map-Größe | Anzahl Faltungsmatrizen |
|---------------------|-------------------|-------------------------|
| 1                   | 224 × 224         | 64                      |
| 2                   | 112 × 112         | 128                     |
| 3                   | 56 × 56           | 256                     |
| 4                   | 28 × 28           | 512                     |
| 5                   | 14 × 14           | 512                     |

## 2.5 Metriken für die Evaluation

Für die Auswertung der Ergebnisse von Methoden im Bereich der semantischen Segmentierung, bei welcher eine pixelweise Klassifikation stattfindet, können unterschiedliche Metriken verwendet werden. Dazu wird jeweils verglichen, ob die von der Methode auf dem Evaluationsdatensatz ermittelte Objektklasse eines Pixels der tatsächlichen Klasse entspricht. Im Folgenden werden diesbezüglich die Berechnung und die Eigenschaften des  $F_1$ -Werts und der Intersection over Union (IoU) erläutert [57]. Für die Auswertung der Ergebnisse dieser Arbeit wurden alle Werte, falls nicht anders angegeben, auf zwei Nachkommastellen gerundet und auf dieser Grundlage miteinander in Vergleich gesetzt.

Für die Ermittlung der Auswertegrößen wird jeder Pixel hinsichtlich der korrekten Objektklassenzuordnung überprüft und entsprechend einer Konfusionsmatrix zugeordnet, wie in Abbildung 10 dargestellt.

### 2.5.1 $F_1$ -Wert

Bei dem  $F_1$ -Wert handelt es sich um eine Größe, welche aus der Precision (Präzision)  $P$  und dem Recall (Sensitivität)  $R$  berechnet wird. Der Wert für die Precision berechnet sich aus dem Verhältnis zwischen den korrekt klassifizierten Pixeln einer Objektklasse  $TP$  und der Summe aus  $TP$  und den falsch klassifizierten Pixeln  $FP$ :

$$P = \frac{TP}{TP + FP}$$

|                     |         | Ermittelte Klasse    |                      |
|---------------------|---------|----------------------|----------------------|
|                     |         | Positiv              | Negativ              |
| Tatsächliche Klasse | Positiv | Richtig Positiv (TP) | Falsch Negativ (FN)  |
|                     | Negativ | Falsch Positiv (FP)  | Richtig Negativ (TN) |

Abbildung 10: Darstellung einer Konfusionsmatrix als Grundlage zur Ermittlung der Größen für die Metriken zur Auswertung der Ergebnisse.

Mit der Präzision kann somit ausgedrückt werden, wie viele der klassifizierten Pixel einer Objektklasse auch tatsächlich dieser Klasse entsprechen. Der Recall-Wert  $R$  ermittelt sich durch

$$R = \frac{TP}{TP + FN}$$

und beschreibt wie viele der Pixel einer bestimmten Objektklasse in Bezug auf die tatsächliche Anzahl von Pixeln dieser Klasse in dem Evaluierungsdatensatz gefunden wurden. Der  $F_1$ -Wert berechnet sich schließlich aus dem harmonischen Mittel zwischen von Precision und Recall:

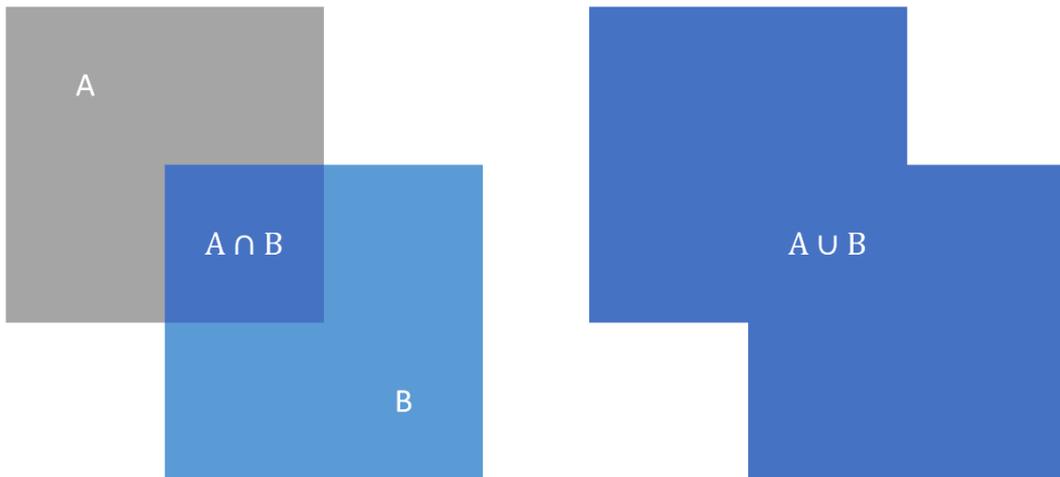


Abbildung 11: Schematische Darstellung der Schnittmenge und der Vereinigungsmenge von A und B.

$$F_1 = 2 \frac{P \cdot R}{P + R}$$

und berücksichtigt dementsprechend beide Größen bei der Bewertung der Ergebnisse. Der  $F_1$ -Wert kann Werte zwischen 0 und 1 annehmen, wobei Werte bei 1 auf eine gute Kombination von Precision und Recall hindeuten. Wenn bei einer Aufgabenstellung mehrere Objektklassen vorhanden sind, wird zur Bewertung der Methode ein Mittelwert aus den einzelnen  $F_1$ -Werten berechnet.

### 2.5.2 Intersection over Union

Bei der Intersection over Union (IoU) handelt es sich um eine Kennzahl für die Bewertung der Ähnlichkeit zweier Mengen A und B. Um diese zu berechnen dividiert man die Schnittmenge von A und B durch die Vereinigungsmenge:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}$$

Die Berechnung ist schematisch in Abbildung 11 dargestellt. Dies entspricht im binären Fall ausgedrückt mit den Einträgen der Konfusionsmatrix:

$$\text{IoU} = \frac{TP}{TP + FP + FN}$$

Die IoU kann Werte zwischen 0 und 1 annehmen, wobei Werte bei 1 auf eine hohe Ähnlichkeit und dementsprechend eine gute Klassifikation der Pixel hindeuten.

### 3 Datensätze

Bei der Entwicklung und Evaluierung der unterschiedlichen Methoden kamen mehrere Datensätze mit unterschiedlichen Eigenschaften zum Einsatz. Alle Datensätze sind dementsprechend annotiert, dass sie für das Training eines CNN für die semantische Segmentierung verwendet werden können. Es wurde somit jedem Pixel in den Bildern eine der Anwendungsaufgabe entsprechende Objektklasse zugeordnet. Während der vergleichsweise umfangreiche ISPRS Potsdam-Datensatz für das Vortrainieren von CNNs zur semantischen Segmentierung verwendet wurde, wurden für die Untersuchungen zur unüberwachten Domain Adaptation zusätzlich weitere Datensätze mit entsprechenden Eigenschaften erstellt. Dabei handelt es sich um den Luftbilddatensatz Sommer, den Luftbilddatensatz Herbst und den Satellitenbilddatensatz Sommer, wobei der Satellitenbilddatensatz Sommer speziell zur Untersuchung von Domain Adaptation im Zusammenhang mit der Segmentierung von Vegetation erstellt wurde und dementsprechend nur diese Objektklasse annotiert wurde.

#### 3.1 ISPRS Potsdam-Datensatz

Bei dem ISPRS Potsdam-Datensatz handelt es sich um zusammenhängenden Luftbilddatensatz eines Teils der Stadt Potsdam in Brandenburg. Der Datensatz besteht aus 38 Bildern mit einer Auflösung von 6000×6000 Pixeln und einer Bodenauflösung (Ground-Sampling-Distance, GSD) von 0,05 m. Der Datensatz beinhaltet neben RGB-Kanälen auch einen NIR-Kanal und ein Höhenmodell, welches allerdings bei dieser Arbeit nicht verwendet wurde. Bei dem Datensatz wurden folgende Objektklassen berücksichtigt:

- Fahrzeuge
- Bäume
- Niedrige Vegetation
- Gebäude
- Versiegelter Boden (z.B. Straßen)
- Sonstiges

Bei den Objektklassen zur Vegetation (Bäume und niedrige Vegetation) muss angemerkt werden, dass die Bäume auf den Bildern bereits kaum noch Blattwerk tragen und die Vegetation, welche sich auf Gebäuden befindet, in der Regel als Gebäude und nicht als Vegetation annotiert wurde.

### 3.2 Luftbilddatensatz Sommer und Luftbilddatensatz Herbst

Bei den beiden neu erstellten Datensätzen Luftbilddatensatz Sommer und Luftbilddatensatz Herbst handelt es sich um Bilder, welche aus einer Höhe von ca. 550 m aufgenommen wurden und eine GSD von ca. 0,09 m besitzen. Zum Einsatz kamen zwei Sensoren mit einer Auflösung von 4864×3232 wobei ein Sensor mit einem „Bayer“-Filter zur Erstellung von RGB-Bildern ausgestattet wurde und bei dem anderen Sensor ein Filter bei 780 nm für die Erstellung von Bildern im nahen Infrarotbereich verwendet wurde. Die jeweiligen korrespondierenden Bilder der beiden Sensoren wurden auf Grundlage geeigneter Passpunkte pixelweise zusammengefügt. Bildinhalt sind bei beiden Datensätzen Luftbilddatensatz Sommer und Luftbilddatensatz Herbst aufgenommen aus dem Jahr 2018 von jeweils unterschiedlichen urbanen Gebieten in Norddeutschland. Die Datensätze umfassen jeweils 20 Bilder wobei 10% für Luftbilddatensatz Sommer und 5% für Luftbilddatensatz Herbst annotiert wurden. Bei den Datensätzen wurden entsprechend dem ISPRS Potsdam-Datensatz folgende Objektklassen berücksichtigt:

- Fahrzeuge
- Bäume
- Niedrige Vegetation
- Gebäude
- Versiegelter Boden (z.B. Straßen)

Objekte, welche bei dem ISPRS Potsdam-Datensatz der Klasse „Sonstiges“ angehören, wurden hierbei einer der anderen Objektklassen zugeordnet. Der Unterschied zwischen beiden Datensätzen besteht hinsichtlich der verschiedenen Charakteristika bezüglich der Jahreszeiten und der Lichtverhältnisse. Während beim Luftbilddatensatz Sommer die Bäume grüne Blätter haben, ist bei dem Luftbilddatensatz Herbst bereits eine Färbung dieser zu erkennen. Weiterhin ist der Schattenwurf auf Bildern des Luftbilddatensatz Sommer deutlich ausgeprägter als auf dem Luftbilddatensatz Herbst und es sind

## Datensätze

dementsprechend viele Objekte wie beispielsweise Fahrzeuge davon bedeckt. Außerdem weisen beide Datensätze eine wahrnehmbar unterschiedliche Helligkeit und allgemeine Farbgebung auf, was durch die Beleuchtung und die jeweiligen Wetterbedingungen gegeben ist. Die Gemeinsamkeiten und Unterschiede der beiden Datensätze sind in Tabelle 3 zusammengefasst.

*Tabelle 3: Gemeinsamkeiten und Unterschiede zwischen Luftbilddatensatz Sommer und Luftbilddatensatz Herbst.*

| <b>Gemeinsamkeiten</b> | <b>Unterschiede</b>                    |
|------------------------|--|
| Sensor                 | Jahreszeit (Färbung der Bäume)         |
| GSD                    | Beleuchtungsbedingungen (Schattenwurf) |
|                        | Helligkeit und Farbgebung              |

### 3.3 Satellitenbilddatensatz Sommer

Bei diesem Datensatz handelt es sich um Bilder aus Gebieten in Norddeutschland, welche mit dem WorldView-2-Sensor im Sommer 2017 aufgezeichnet wurden. Der Sensor verfügt über einen panchromatischen Kanal und acht zusätzliche multispektrale Bänder. Zum Einsatz in dieser Arbeit kamen dabei allerdings nur die RGB-Kanäle und der von der Wellenlänge zu oben beschriebenen Luftbilddatensätzen her passende NIR-Kanal. Eine Übersicht ist in Tabelle 4 gegeben. Der Sensor befindet sich in einer Höhe von 770 km und verfügt im panchromatischen Kanal über eine GSD von 0,46 m (bei Nadir) und 0,52 m (bei 20° zu Nadir). Für den multispektralen Bereich ist die GSD bei 1,85 m (bei Nadir) und 2,07 m (bei 20° zu Nadir). Um die Auflösung der Farbkanäle an den panchromatischen Kanal anzupassen, wurde ein IHS (Intensity-Hue-Saturation) basiertes Pansharpeneing durchgeführt [58] [59]. Das Prinzip dabei ist, dass das zu bearbeitende Farbbild in den IHS-Farbraum überführt und der Intensitätskanal durch das panchromatische Bild ersetzt wird. Daraufhin findet eine Rücktransformation statt. Anschließend wurde die Auflösung der Bilder noch entsprechend der GSD and die Luftbilddatensätze angepasst.

## Datensätze

Tabelle 4: Die entsprechenden Wellenlängen der einzelnen Kanäle des WorldView-2-Sensors.

| Kanal        | Wellenlänge [nm] |
|--------------|------------------|
| Coastal      | 400 – 450        |
| <b>Blue</b>  | <b>450 – 510</b> |
| <b>Green</b> | <b>510 – 580</b> |
| Yellow       | 585 – 625        |
| <b>Red</b>   | <b>630 – 690</b> |
| Red Edge     | 705 – 745        |
| <b>NIR 1</b> | <b>770 – 895</b> |
| NIR 2        | 860 – 1040       |

Der annotierte Datensatzumfang entspricht dem von Luftbilddatensatz Herbst. Bei dem Datensatz wurden nur die beiden folgende Objektklassen berücksichtigt:

- Vegetation
- Sonstiges

Die Gemeinsamkeiten und Unterschiede zum Luftbilddatensatz Sommer sind in Tabelle 5 zusammengefasst. Beispielhaft sind in Abbildung 12 RGB-Bilder der vier in diesem Kapitel vorgestellten Datensätze dargestellt.

Tabelle 5: Gemeinsamkeiten und Unterschiede zwischen Luftbilddatensatz Sommer und Satellitenbilddatensatz Sommer.

| Gemeinsamkeiten                        | Unterschiede              |
|--|---------------------------|
| Jahreszeit (Färbung der Bäume)         | Sensor                    |
| Beleuchtungsbedingungen (Schattenwurf) | GSD                       |
|  | Helligkeit und Farbgebung |

## Datensätze

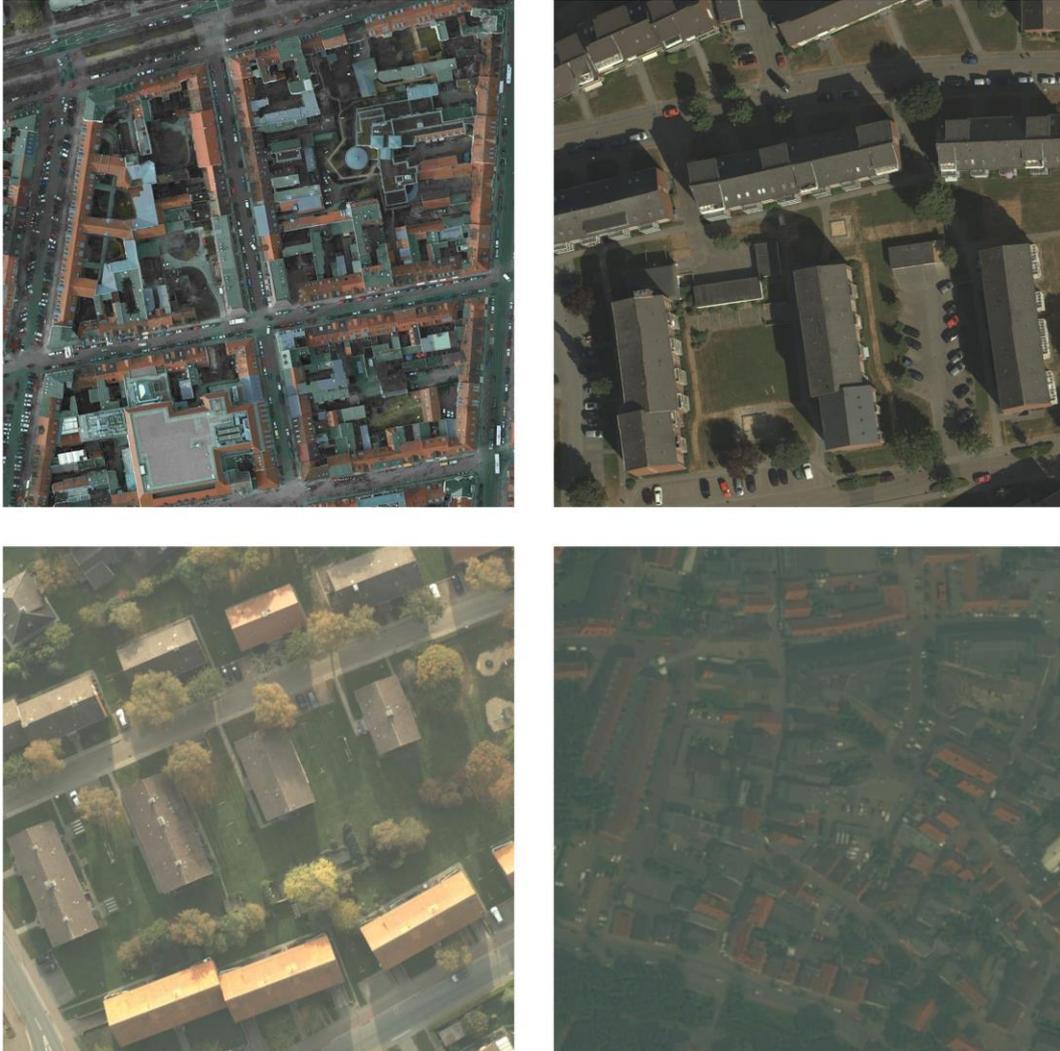


Abbildung 12: Oben links: RGB-Bild von ISPRS Potsdam-Datensatz. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer. Unten links: RGB-Bild von Luftbilddatensatz Herbst. Unten rechts: RGB-Bild von Satellitenbilddatensatz Sommer.



## 4 Domain Adaptation mit Adaptionmodulen

Im folgenden Kapitel wird eine Methode zur Verbesserung der überwachten Domain Adaptation für die semantische Segmentierung vorgestellt. Dabei wird eine Encoder-Decoder-Struktur, welche einer Variante der SegNet-Architektur entspricht, allerdings mit Skip-Connections wie bei U-Net, zunächst auf einem umfangreich annotierten Quelldomäne-Datensatz für die semantische Segmentierung trainiert. Anschließend wird diese Struktur für den Domain Adaptation-Anpassungsschritt zur Zieldomäne erweitert. Dazu werden zwischen den Encoder und Decoder sogenannte Adaptionmodule platziert, welche den hauptsächlichen Anteil des Anpassungsvorgangs übernehmen, während die Gewichte des Encoders und Decoders weitgehend unverändert bleiben, um die bereits erlangte Klassifikationsqualität auf der entsprechenden Feature-Basis möglichst beizubehalten. Für das Training des neuronalen Netzes wird ein alternierendes Schema angewandt. Die Auswertung der Methode wurde mit dem ISPRS Potsdam-Datensatz als Quelldomäne und dem Luftbilddatensatz Sommer als Zieldomäne durchgeführt (siehe Abbildung 13).

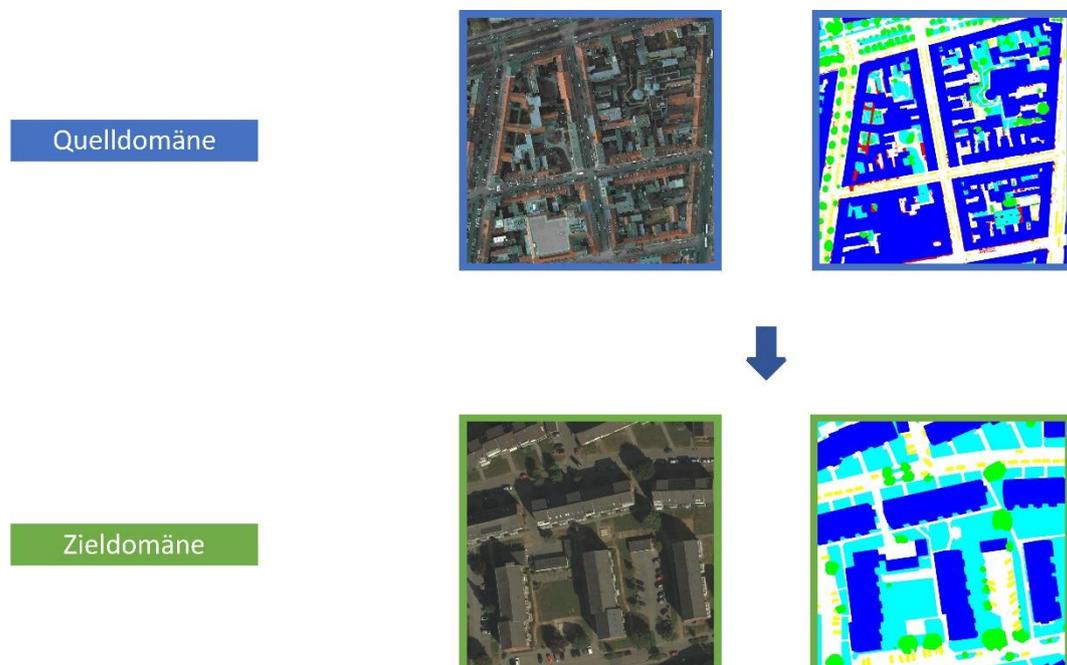


Abbildung 13: Die Domain Adaptation erfolgt vom ISPRS-Datensatz als Quelldomäne zum Luftbilddatensatz Sommer als Zieldomäne. Da es sich um überwachtes Lernen handelt, ist für alle RGB-Bilder, welche im Trainingsvorgang verwendet werden auch eine entsprechende Annotation vorhanden.

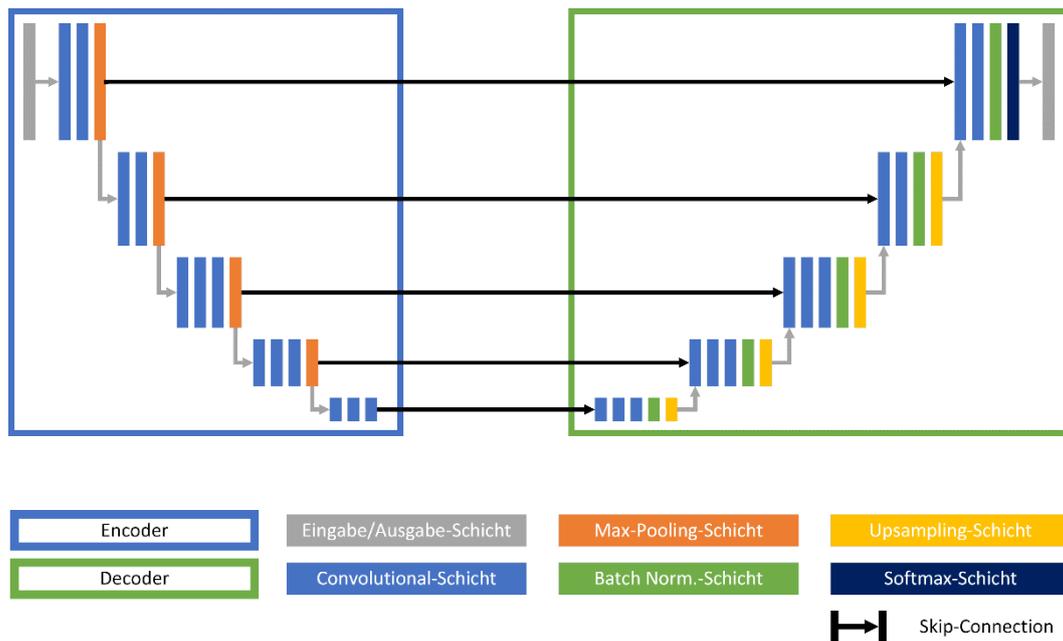


Abbildung 14: Dargestellt ist die Encoder-Decoder-Struktur für die semantische Segmentierung, welche auf dem Quelldomäne-Datensatz trainiert wird.

Die Methode wurde mit den jeweiligen RGB-Bildern der Datensätze trainiert und evaluiert unter Berücksichtigung aller verfügbaren Objektklassen. Da es sich bei der Methode um überwachte Domain Adaptation handelt, sind sowohl für die Quelldomäne als auch für die Zieldomäne Annotationen zu den jeweiligen RGB-Bildern vorhanden.

#### 4.1 Methode

Die Netzarchitektur der Methode basiert auf einer Encoder-Decoder-Struktur in Anlehnung an die Struktur von SegNet [49]. Der Encoder besteht aus einer Eingabeschicht mit einer Eingangsgröße von  $224 \times 224$  gefolgt von fünf Convolutional-Blöcken. Jeder dieser Convolutional-Blöcke besteht aus einer Folge von Convolutional-Schichten und einer abschließenden Max-Pooling-Schicht mit einer Halbierung der vorangegangenen Auflösung ( $2 \times 2$  als Pooling-Fenstergröße). An jede der Convolutional-Schichten ist außerdem eine ReLU-Schicht als Aktivierungsfunktion angefügt. Die Zusammensetzung der Convolutional-Blöcke entspricht dabei den Schichten der VGG16-Netzwerkarchitektur [44]. Der Decoder entspricht einer gespiegelten Struktur, wobei die Max-Pooling-Schichten der einzelnen Convolutional-Blöcke jeweils durch eine Batch Normalization-Schicht [56] und eine den Pooling-Schichten des Encoders entsprechende Upsampling-Schicht ersetzt werden.

## Domain Adaptation mit Adaptionmodulen

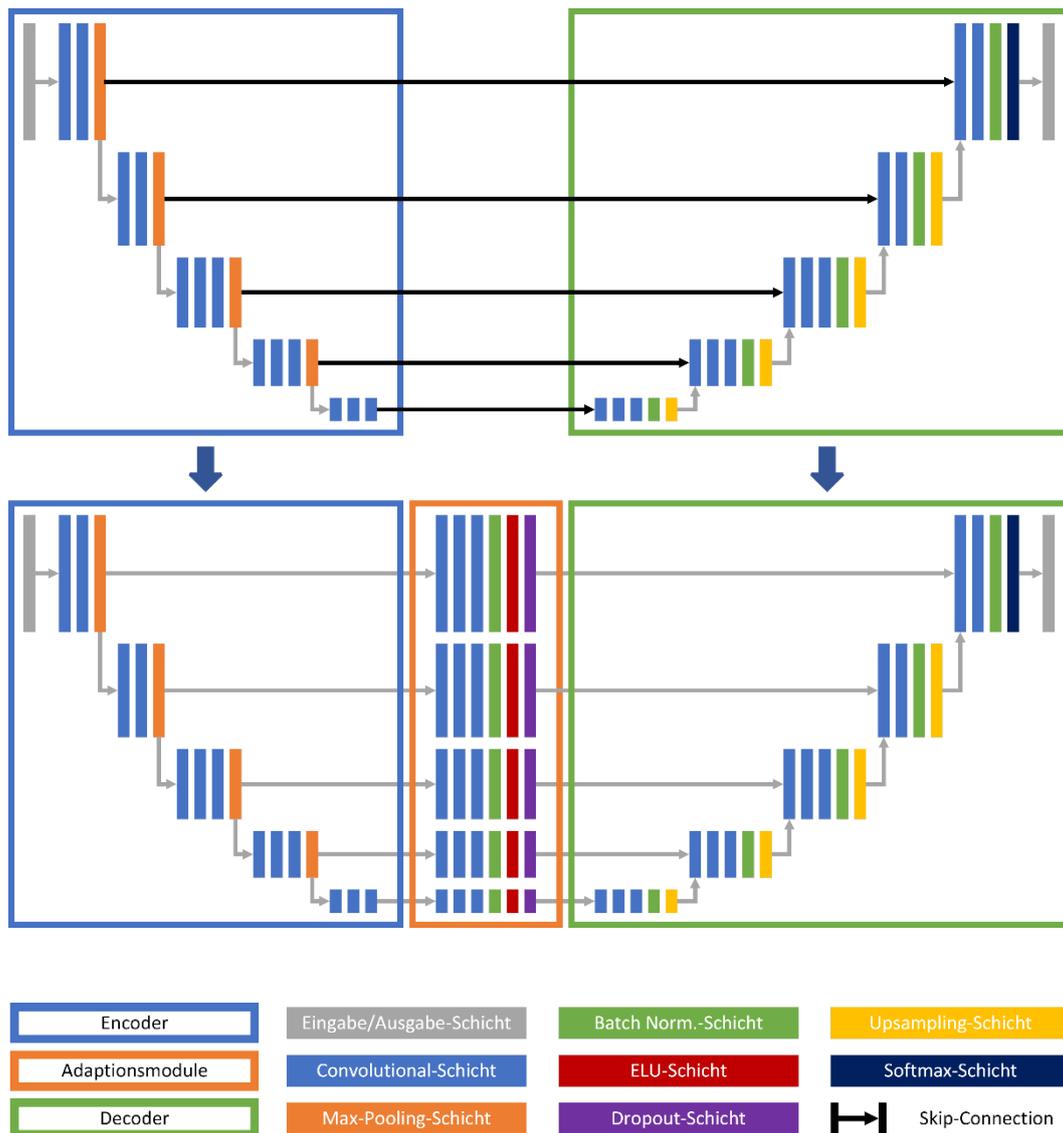


Abbildung 15: Dargestellt ist die Ergänzung der Encoder-Decoder-Struktur um die für den Anpassungsprozess eingesetzten Adaptionmodule. Dazu wird jede Skip-Connection, der auf der Quelldomäne trainierten Grundstruktur, durch ein entsprechendes Modul ersetzt.

Zwischen den jeweiligen Ebenen des Encoders und des Decoders wurden wie bei U-Net-Netzwerkarchitekturen noch Skip-Connections hinzugefügt [48]. Im Anschluss an den letzten Convolutional-Block des Decoders kommt eine Softmax-Schicht für den pixelweisen Klassifikationsschritt zum Einsatz. Die Netzwerkarchitektur dieser Grundstruktur ist in Abbildung 14 dargestellt.

Für den Domain-Adaptation-Prozess wird diese Grundstruktur nun noch um fünf Adaptionmodule ergänzt. Diese ersetzen die Skip-Connections und werden dementsprechend

auf jeder Ebene zwischen dem Encoder und dem Decoder platziert. Jedes der Adaptionmodule besteht aus einer Folge von drei Convolutional-Schichten (mit entsprechenden ReLU-Schichten), einer Batch Normalization-Schicht, einer ELU-Schicht [60] und einer Dropout-Schicht [55] mit einer Rate von 0,5. Die Batch Normalization-Schicht und die Dropout-Schicht wurden als Regularisierungsmethoden für eine Stabilisierung des Trainings eingesetzt. Die Anzahl der Filter der Convolutional-Schichten und die Anzahl der Feature-Maps nimmt mit jeder Ebene entsprechend der Struktur eines VGG16-Netzwerks zu. Eine Übersicht dazu ist in Tabelle 6 dargestellt. Die finale Netzwerkarchitektur mit der Integration der Adaptionmodule in den einzelnen Ebenen zwischen den Convolutional-Blöcken des Encoders und Decoders ist in Abbildung 15 dargestellt.

Im ersten Schritt wird die Grundstruktur auf Daten der Quelldomäne trainiert. Dazu werden zunächst die Gewichte des Encoders auf Grundlage eines auf ImageNet [43] vortrainierten VGG16-Klassifikationsnetzwerks initialisiert. Anschließend findet ein Fine-Tuning mit den annotierten Daten der Quelldomäne statt. Als Fehlerfunktion kommt die kategorische Kreuzentropie zum Einsatz und als Optimierer wird Adam [61] mit einer Lernrate von 0,001 verwendet. Zusätzlich wurde für das Training Data Augmentation eingesetzt mit den gängigen Bildoperatoren. Im nächsten Schritt findet die Anpassung des auf der Quelldomäne trainierten CNNs zur Zieldomäne statt. Dazu werden der Encoder und der Decoder, mit der um die Adaptionmodule ergänzten Netzarchitektur, auf Grundlage des Encoders und des Decoders der auf der Quelldomäne trainierten Grundstruktur initialisiert. Für das Training der Gesamtstruktur werden zunächst die Gewichte des Encoders und Decoders eingefroren und nur die Gewichte der Adaptionmodule trainiert. Anschließend werden die Gewichte der Adaptionmodule für wenige Epochen eingefroren und die Encoder-Decoder-Struktur nachtrainiert. Dieses alternierende Schema wird für das weitere Training beibehalten, wobei Encoder und Decoder jeweils nur für wenige Epochen trainiert werden.

Tabelle 6: Anzahl der Filter und die Größe der entsprechenden Feature-Maps bei den jeweiligen Adaptionmodulen.

| Adaptionsmodulebene | Größe der Feature-Maps | Anzahl der Filter |
|---------------------|------------------------|-------------------|
| 1                   | $224 \times 224$       | 64                |
| 2                   | $112 \times 112$       | 128               |
| 3                   | $56 \times 56$         | 256               |
| 4                   | $28 \times 28$         | 512               |
| 5                   | $14 \times 14$         | 512               |

## 4.2 Ergebnisse und Diskussion

Für die Auswertung der Ergebnisse wurde als Referenzmethode das Fine-Tuning der ursprünglichen Encoder-Decoder-Struktur verwendet, was die gängige Methode zur Anpassung von neuronalen Netzen an neue Daten darstellt. Dazu wurde auf Grundlage der annotierten Trainingsdaten der Zieldomäne ein Nachtraining des auf der Quelldomäne trainierten CNNs durchgeführt. Für die Evaluation wurden im Folgenden fünf Teildatensätze der Zieldomäne mit unterschiedlichen Größen erstellt, um die Abhängigkeit der Domain Adaptation-Methoden zur verfügbaren annotierten Datenmenge zu bestimmen. Dabei wurden jeweils 4, 8, 16, 32 und 48 Bilder der Größe  $224 \times 224$  verwendet. Entsprechend wurde bei den Datensätzen überprüft, dass auch jede Objektklasse vertreten ist. Dies ist für eine vollständige Auswertung notwendig, da es vor allem bei dem sehr kleinen Teildatensatz mit vier Bildern dazu hätte führen können, dass bestimmte Klassen nicht als Beispiele in der Zieldomäne vertreten wären, was wiederum allerdings für eine überwachte Domain Adaptation notwendig gewesen wäre. Für jeden dieser Teildatensätze wurde jeweils ein Domain Adaptation-Prozess basierend auf dem Einsatz von Adaptionmodulen und als Referenz mit der Fine-Tuning-Methode durchgeführt. Die Ergebnisse angegeben als Gesamtgenauigkeit und als durchschnittlicher  $F_1$ -Wert sind in Tabelle 7 und Tabelle 8 zusammengefasst. Eine grafische Darstellung der semantischen Segmentierung basierend auf der Methode mit Adaptionmodulen ist in Abbildung 16 dargestellt. Bei dem Vergleich der Methode mit dem Einsatz von Adaptionmodulen zur Fine-Tuning-Methode kann man feststellen, dass durch den Einsatz der hier vorgestellten Struktur für alle eingesetzten

Teildatensätze eine Verbesserung sowohl in Hinblick auf die Gesamtgenauigkeit, wie auch den durchschnittlichen  $F_1$ -Wert beobachtet werden kann. Dabei zeigt sich eine zunehmende Differenz bei der Klassifikationsgenauigkeit, je kleiner der jeweilige Teildatensatz ist. Während bei dem Einsatz von 48 Bildern die Gesamtgenauigkeit sich um 0,43% und der durchschnittliche  $F_1$ -Wert um 2,36% verbessert, beträgt die Verbesserung bei dem Einsatz des Teildatensatzes von lediglich vier Bildern für die Gesamtgenauigkeit 2,68% und für den durchschnittlichen  $F_1$ -Wert 5,41%. Somit kann die Qualität der semantischen Segmentierung durch den Einsatz von Adaptionmodulen im Besonderen bei vorhanden sein von nur sehr wenigen annotierten Daten für den Domain Adaptation-Prozess erhöht werden. Außerdem kann man beobachten, dass die Ergebnisse bei dem Einsatz von Adaptionmodulen und dem Teildatensatz mit vier Bildern bereits besser ausfallen, als bei der Referenzmethode bei dem Teildatensatz mit acht Bildern. Dadurch lässt sich zeigen, dass durch die Erweiterung des Segmentierungsnetzwerks auf Grundlage des Einsatzes von Adaptionmodulen auf Basis von überwachter Domain Adaptation die Menge an notwendigen annotierten Trainingsdaten stärker reduziert werden kann, bei einer gleichzeitigen Verbesserung der Klassifikationsgenauigkeit. Dementsprechend konnte dargestellt werden, dass durch das Integrieren dieser zusätzlichen Strukturen, die bereits guten Klassifikationseigenschaften des Segmentierungsnetzwerks, basierend auf der Quelldomäne, besser erhalten bleiben konnten im Vergleich zu einem Fine-Tuning der Gesamtstruktur. Dementsprechend kann für die überwachte Domain Adaptation eine Verbesserung in Form einer nachträglichen Erweiterung des Netzwerks und eines entsprechenden Trainings erreicht werden.

## Domain Adaptation mit Adaptionmodulen

*Tabelle 7: Ergebnisse der Fine-Tuning-Methode und der Adaptionmodule-Methode als Gesamtgenauigkeit in Prozent für die unterschiedlichen Teildatensätze.*

| <b>Anzahl Bilder</b>  | <b>4</b> | <b>8</b> | <b>16</b> | <b>32</b> | <b>48</b> |
|-----------------------|----------|----------|-----------|-----------|-----------|
| <b>Fine-Tuning</b>    | 79,97    | 81,41    | 84,71     | 89,13     | 91,61     |
| <b>Adaptionmodule</b> | 82,11    | 85,04    | 86,44     | 90,88     | 92,00     |

*Tabelle 8: Ergebnisse der Fine-Tuning-Methode und der Adaptionmodule-Methode als durchschnittlicher  $F_1$ -Wert für die unterschiedlichen Teildatensätze.*

| <b>Anzahl Bilder</b>  | <b>4</b> | <b>8</b> | <b>16</b> | <b>32</b> | <b>48</b> |
|-----------------------|----------|----------|-----------|-----------|-----------|
| <b>Fine-Tuning</b>    | 0,74     | 0,76     | 0,79      | 0,81      | 0,86      |
| <b>Adaptionmodule</b> | 0,78     | 0,81     | 0,83      | 0,87      | 0,88      |

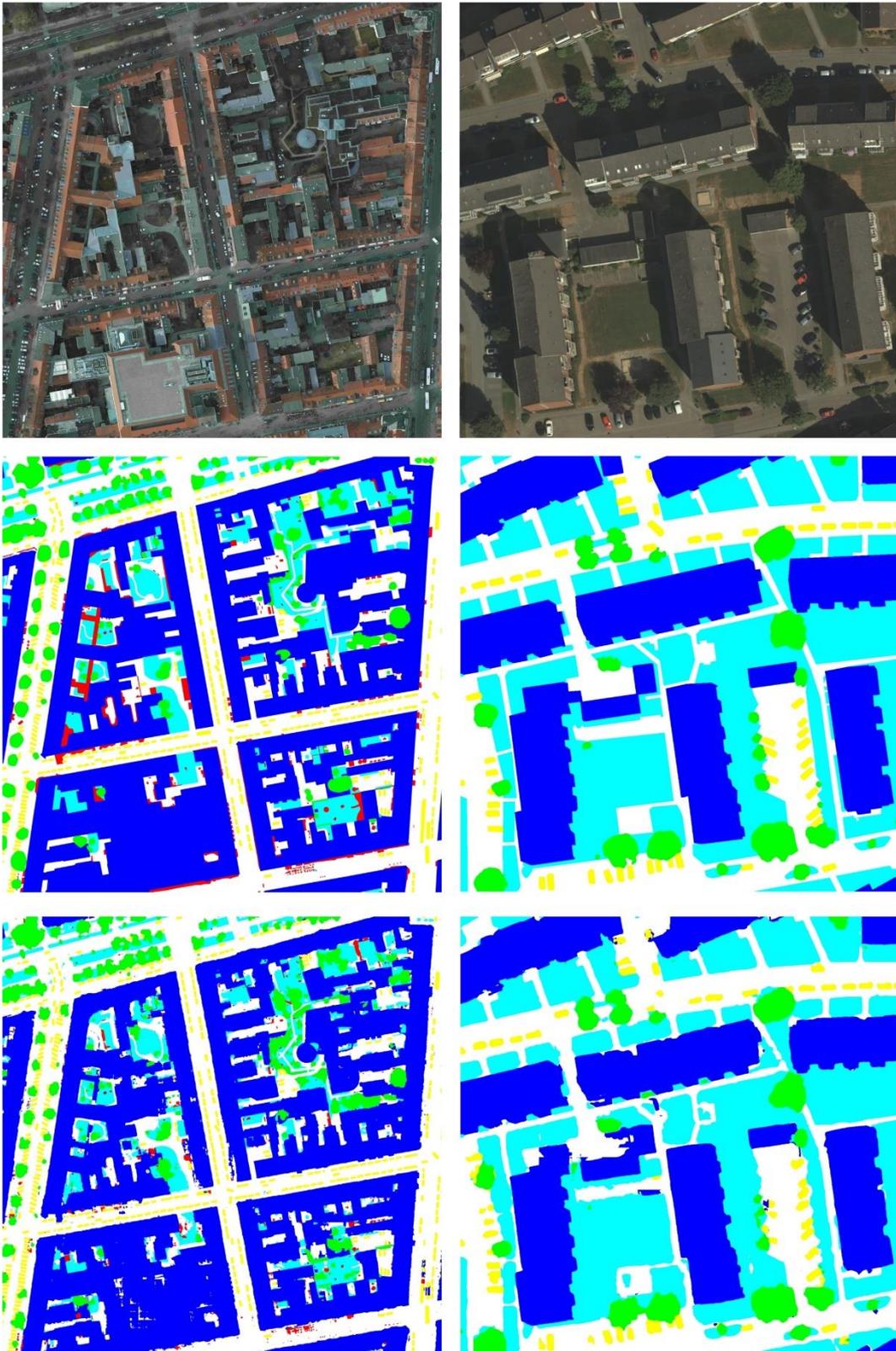


Abbildung 16: Dargestellt ist auf der linken Seite von oben nach unten: RGB-Bild, Ground Truth und Ergebnisse der semantischen Segmentierung des ISPRS Potsdam-Datensatzes und auf der rechten Seite von oben nach unten: RGB-Bild, Ground Truth und Ergebnisse der semantischen Segmentierung von Luftbilddatensatz Sommer.

## 5 Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

In diesem Kapitel wird eine Methode zur unüberwachten Domain Adaptation für die semantische Segmentierung vorgestellt. Dazu werden Cycle-Consistent Adversarial Networks eingesetzt um einen Style-Transfer zwischen der Quell- und der Zieldomäne durchzuführen. Zunächst wird ein CNN für die semantische Segmentierung basierend auf dem Quelldomäne-Datensatz trainiert. Das Cycle-Consistent Adversarial Network wird im Anschluss wiederum auf nicht-annotierten Daten des Quell- und Zieldomäne-Datensatzes trainiert. Anschließend werden auf Grundlage dieser Style-Transfer-Methode synthetische Daten erzeugt, um das CNN für die semantische Segmentierung zu optimieren und entsprechend für die Zieldomäne anzupassen. Dazu wurde ein kontinuierliches Trainingschema konzipiert und verwendet. Für die Auswertung wurde als Quelldomäne der Luftbilddatensatz Sommer und für die Zieldomäne der Luftbilddatensatz Herbst verwendet (siehe Abbildung 17). Die Methode wurde auf die jeweiligen RGB-Bilder unter

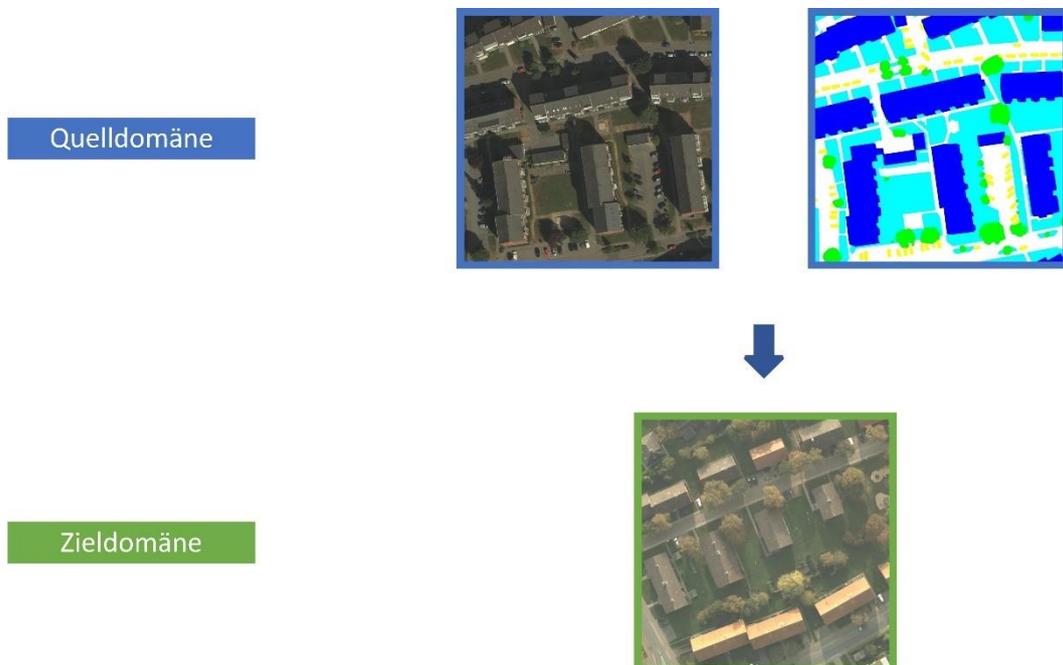


Abbildung 17: Die Domain Adaptation erfolgt vom Luftbilddatensatz Sommer als Quelldomäne zum Luftbilddatensatz Herbst als Zieldomäne. Da es sich um unüberwachtes Lernen handelt, ist für alle RGB-Bilder der Zieldomäne, welche im Trainingsvorgang verwendet werden keine Annotation vorhanden.

## Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

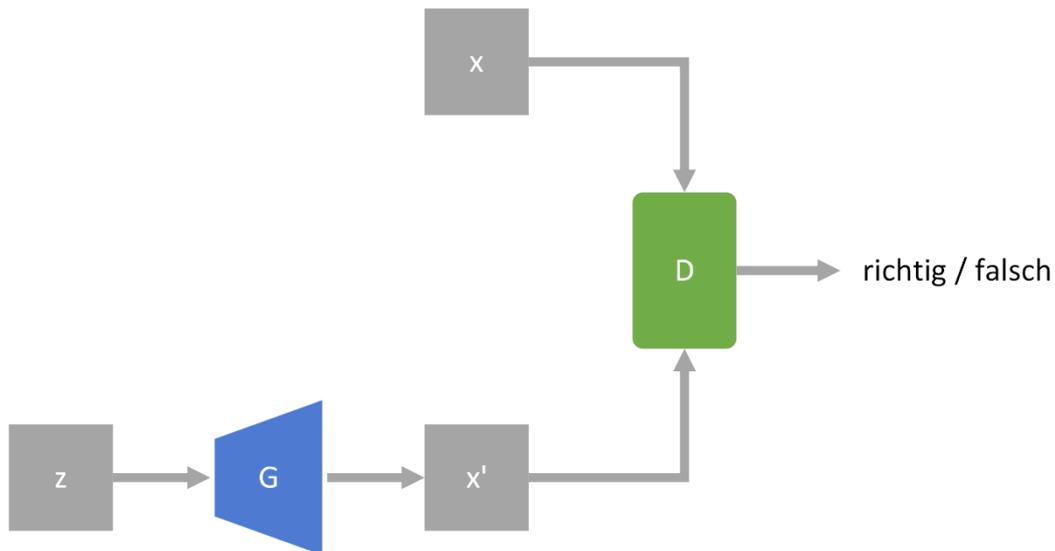


Abbildung 18: Darstellung eines Generative Adversarial Networks mit einem Vektor  $z$ , einem Generator  $G$ , der Ausgabe des Generators  $x'$ , einem Diskriminator  $D$  und einem Datensatz  $x$ .

Berücksichtigung aller verfügbaren Objektklassen angewandt. Da es sich bei der Methode um unüberwachte Domain Adaptation handelt, sind während des Trainings für die Zieldomäne keine Annotationen zu den jeweiligen RGB-Bildern vorhanden. Das Grundprinzip dieser Methode beruht auf der Synthese von Bildern basierend auf Generative Adversarial Networks (GANs). Dabei handelt es sich um einen Algorithmus, welcher dazu verwendet werden kann Repräsentationen zu lernen ohne eine große Menge an annotierten Daten. Dies kann durch die Kombination mehrerer neuronaler Netze erreicht werden. GANs gehören zu den unüberwachten Lernverfahren und können mit CNNs realisiert werden [62]. Ein GAN besteht in der einfachsten Form aus einem Generator und einem Diskriminator. Dabei wird der Generator dahingehend trainiert, synthetische Bilder zu erzeugen, welche von der Erscheinung her einem vorhandenen Datensatz entsprechen. Der Diskriminator lernt wiederum, die von dem Generator erzeugten synthetischen Bilder von echten Bildern aus dem jeweiligen verwendeten Datensatz zu unterscheiden (siehe Abbildung 18). Um ein GAN zu trainieren, werden zunächst auf Basis eines Generators  $G$  mit den Werten eines Vektors  $z$  mehrere Bilder synthetisch erzeugt. Anfangs sind diese Beispielbilder  $x'$  basierend auf  $G(z)$  nur Rauschen, da noch kein Training des Generators stattgefunden hat. Im zweiten Schritt wird der Diskriminator  $D$  trainiert. Dieser bekommt die Beispiele  $x'$ , welche vom Generator erzeugt wurden und reale Beispielbilder  $x$  aus einem Datensatz. Anhand der Bilder aus  $x'$  und  $x$  lernt der Diskriminator so zwischen synthetischen

und realen Bildern zu unterscheiden und eine entsprechende Wahrscheinlichkeit dieser Zuordnung basierend auf einer Fehlerfunktion  $L(G, D)$  auszugeben. Sobald der Diskriminator ausreichend optimiert wurde und nur noch eine geringe Unsicherheit aufweist, werden die Gewichte des Diskriminators eingefroren und es wird der Generator trainiert. Basierend auf der Ausgabe des Diskriminators während des Trainings, kann nun der Generator dahingehend optimiert werden, Bilder so zu synthetisieren, dass diese vom Diskriminator nun nicht mehr als synthetisch erzeugte Bilder erkannt werden. Sobald bei dem Generator wiederum eine ausreichende Optimierung stattgefunden hat und dieser ausreichend stabil Bilder erzeugt, welche vom Diskriminator nicht von den realen Bildern unterschieden werden können, werden die Gewichte des Generators eingefroren und der Diskriminator erneut trainiert, diesmal basierend auf den neuen synthetischen Bildern. Dieses abwechselnde Trainingsschema findet nun fortlaufend statt, bis der Generator in der Lage ist Bilder zu erzeugen, welche den Bildern aus dem eigesetzten Datensatz entsprechen. Für das fortlaufend abwechselnde Training des Generators und des Diskriminators wird die Fehlerfunktion

$$L_{GAN}(G, D) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log (1 - D(G(z)))]$$

eingesetzt, wobei für die Anpassung der Gewichte der beiden neuronalen Netze eine Optimierung mit der Zielsetzung

$$\min_G \max_D L_{GAN}(G, D)$$

durchgeführt werden soll [5]. Eine Möglichkeit, um in das Konzept von GANs die Berücksichtigung von objektkategorischen Bedingungen zu integrieren, ist der Einsatz eines Conditional Adversarial Networks (Conditional GAN, cGAN). Während bei einem einfachen GAN der Generator die Abbildung  $G: z \rightarrow x'$  lernt, ist bei einem Conditional GAN noch eine Bedingung  $c$  (beispielsweise ein Bild) vorhanden, sodass entsprechend  $G: \{c, z\} \rightarrow x'$  erlernt wird (siehe Abbildung 19). Es ergibt sich also dementsprechend für die Fehlerfunktion  $L_{cGAN}(G, D)$  des Conditional GANs die Form:

$$L_{cGAN}(G, D) = E_{x \sim p_{data}(x)} [\log D(x|c)] + E_{z \sim p_z(z)} [\log (1 - D(G(z|c)))]$$

## Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

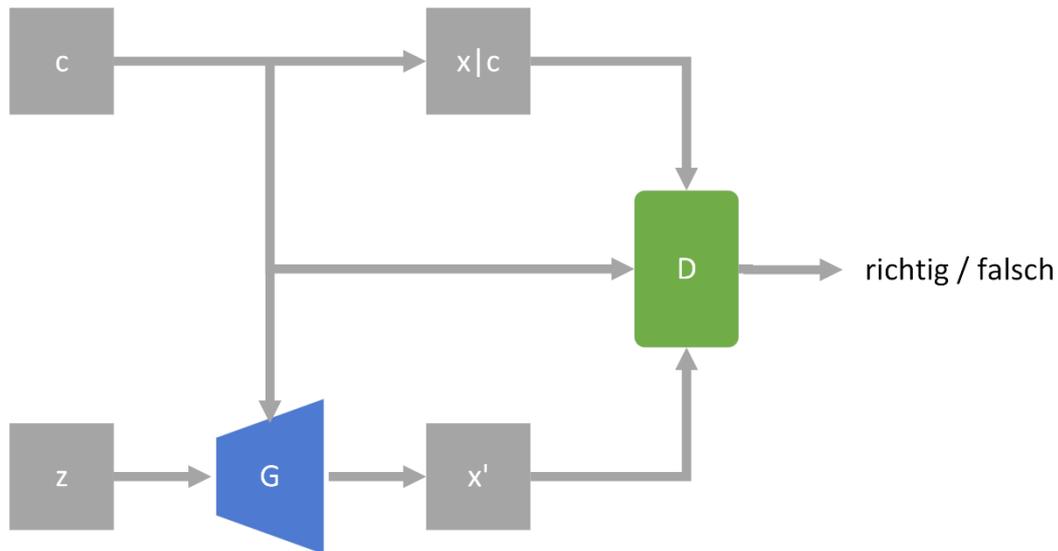


Abbildung 19: Darstellung eines Conditional Adversarial Networks mit einem Vektor  $z$ , einer Bedingung  $c$ , einem Generator  $G$ , der Ausgabe des Generators  $x'$ , einem Diskriminator  $D$  und einem Datensatz  $x$ .

Basierend auf diesem Prinzip können unterschiedliche Anwendungen im Bereich der Bild-zu-Bild-Übersetzung umgesetzt werden [63]. Wenn für entsprechende Style-Transfers zwischen zwei Datensätzen allerdings keine paarweisen Bilder  $\{x_i, y_i\}_{i=1}^N$  vorhanden sind, entsprechend der Conditional GANs, sondern nur Trainingsdaten im Quelldatensatz  $\{x_i\}_{i=1}^N \in X$  und Zieldatensatz  $\{y_k\}_{k=1}^M \in Y$ , ohne dass diese zusammenpassen, so können Cycle-Consistent Adversarial Networks eingesetzt werden. Dabei handelt es sich um eine Struktur, welche aus zwei Generatoren ( $G$  und  $F$ ), welche zyklisch verknüpft sind besteht und zwei Diskriminatoren ( $D_X$  und  $D_Y$ ) beinhaltet [6]. Die Grundstruktur ist in Abbildung 20 dargestellt. Als Fehlerfunktionen kommen dabei zum einen für die eine Seite der Struktur mit dem Generator  $G$  und dem Diskriminator  $D_Y$  der Term  $L_{GAN}(G, D_Y, X, Y)$  zum Einsatz und entsprechend für die andere Seite mit dem Generator  $F$  und dem Diskriminator  $D_X$  der Term  $L_{GAN}(F, D_X, Y, X)$ . Zusätzlich wird bei einem Cycle-Consistent Adversarial Network noch die Bedingung der zyklischen Konsistenz hinzugefügt:

$$L_{cyc}(G, F) = E_{y \sim p_{data}(y)} [\|G(F(y)) - y\|_1] + E_{x \sim p_{data}(x)} [\|F(G(x)) - x\|_1]$$

Dabei wird überprüft, ob ein Bild, welches einen vollen Zyklus durchläuft, wieder seine ursprüngliche Darstellung hat. Die gesamte Fehlerfunktion ergibt sich also zu:

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F)$$

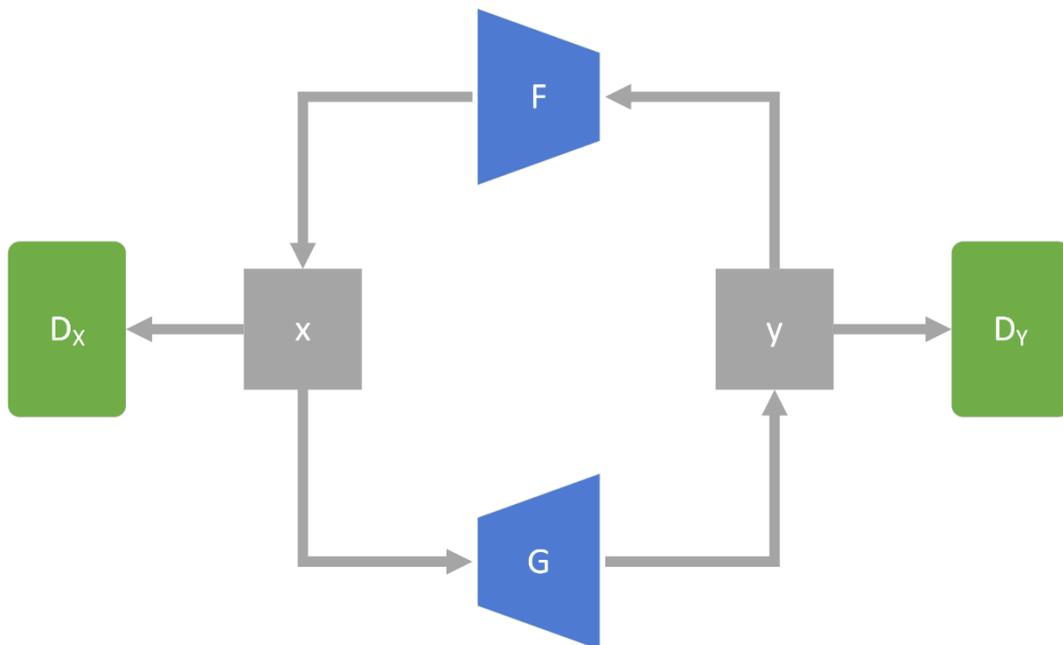


Abbildung 20: Darstellung eines Cycle-Consistent Adversarial Networks mit den Generatoren  $G$  und  $F$ , den Diskriminatoren  $D_X$  und  $D_Y$  und den Datensätzen  $x$  und  $y$ .

Es kann zusätzlich noch bestimmt werden, wie stark die einzelnen Terme gewichtet werden und so zur Gesamtfehlerfunktion beitragen.

## 5.1 Methode

Die hier vorgestellte Methode zur Domain Adaptation setzt sich aus zwei Teilen zusammen. Der erste Teil stellt ein Cycle-Consistent Adversarial Network dar, welches daraufhin trainiert wird einen Style-Transfer zwischen Bildern der Quelldomäne  $s \in S$  und der Zieldomäne  $t \in T$  durchzuführen. Das Cycle-Consistent Adversarial Network besteht aus zwei Generatoren ( $G$  und  $F$ ) und zwei Diskriminatoren ( $D_S$  und  $D_T$ ). Der Generator  $G$  übersetzt dabei Bilder von der Quelldomäne in die Zieldomäne ( $G: S \rightarrow T$ ) und der Generator  $F$  entsprechend umgekehrt ( $F: T \rightarrow S$ ). Der Diskriminator  $D_S$  wird dahingehend trainiert, um zwischen echten Bildern von  $s$  und übersetzten Bildern von  $F(t)$  zu unterscheiden und  $D_T$  zwischen echten Bildern der Zieldomäne  $t$  und übersetzten Bildern der Quelldomäne  $G(s)$ . Der zweite Teil der Methode führt ein kontinuierliches Fine-Tuning der CNN-basierten Methode zur semantischen Segmentierung durch. Dabei werden die von Generator  $G$  übersetzten Bilder kontinuierlich für ein Fine-Tuning des Segmentierungsnetzwerks verwendet.

Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

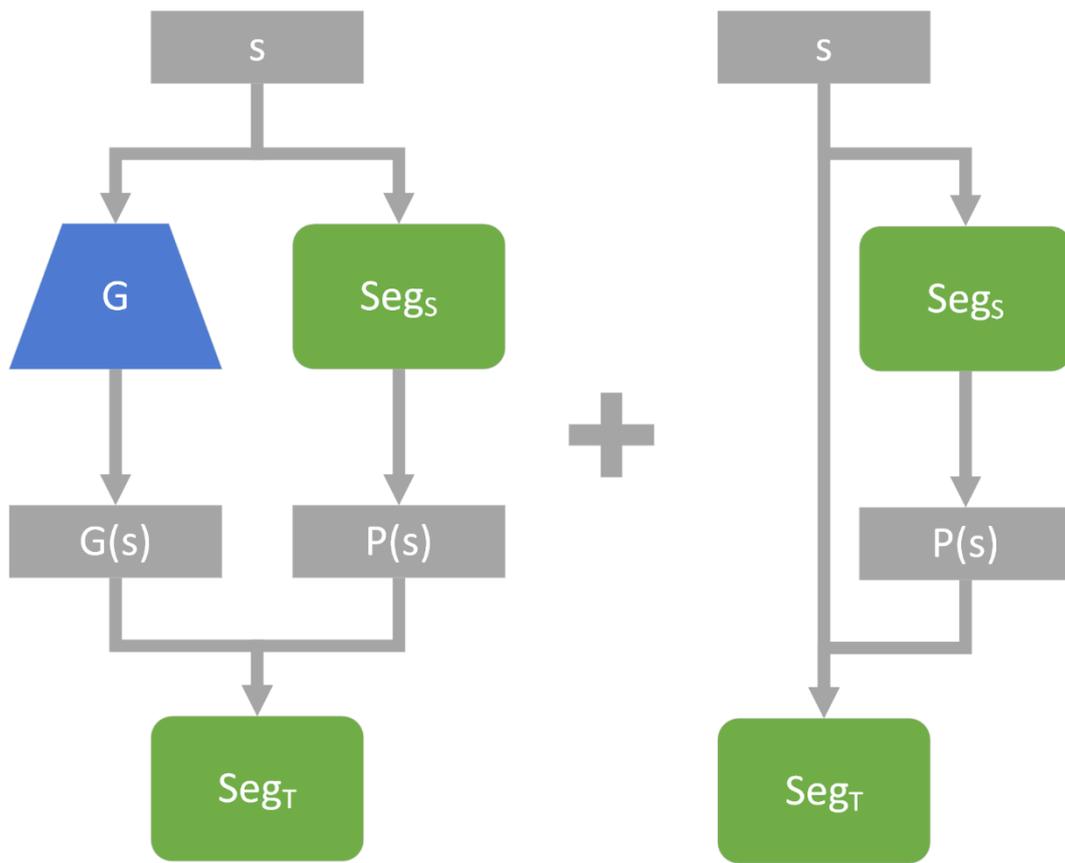


Abbildung 21: Während des Trainings von  $Seg_T$  kommen sowohl vom Generator  $G$  erzeugte Bilder wie auch unveränderte Bilder der Quelldomäne zum Einsatz.

Bei dieser Methode wurden die Eingangsgrößen der Netzwerke auf  $224 \times 224$  mit drei Kanälen (RGB) festgelegt. Für die beiden Generatoren wurden U-Net basierte Strukturen mit Leaky ReLU als Aktivierungsfunktionen eingesetzt [6]. Bei der Zielanwendung der semantischen Segmentierung wurde die Encoder-Decoder-Struktur mit Skip-Connections aus Kapitel 4 integriert. Für das Training des Cycle-Consistent Adversarial Networks werden die Fehlerfunktionen wie oben beschrieben verwendet. Außerdem kommt noch ein Term  $L_{Identität}$  hinzu, welcher bestimmt inwiefern die Generatoren ein Bild unverändert lassen, falls dieses bereits aus der entsprechenden Domäne kommt [6]. Die gesamte Fehlerfunktion ergibt sich dementsprechend zu

$$L(G, F, D_S, D_T) = L_{GAN}(G, D_T, S, T) + L_{GAN}(F, D_S, T, S) + L_{cyc}(G, F) + L_{Identität}(G, F)$$

wobei eine Optimierung der Gesamtstruktur mit den Generatoren und Diskriminatoren unter Einbezug der Gesamtfehlerfunktion während des Trainings hinsichtlich der Zielsetzung:

## Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

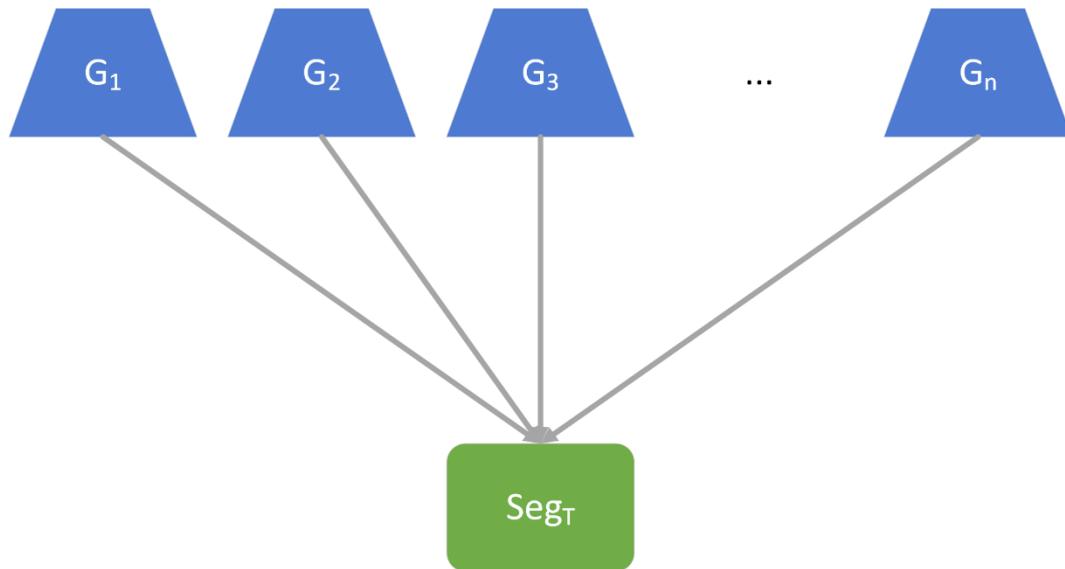


Abbildung 22: Während des Trainings kommen mehrere Varianten des Generators  $G$  zum Einsatz um das Segmentierungsnetzwerk  $Seg_T$  zu trainieren.

$$G^*, F^* = \min_G \max_D L(G, F, D_S, D_T)$$

durchgeführt wird. Das Encoder-Decoder-Netzwerk beinhaltet im Encoder ein auf ImageNet vortrainiertes VGG16-Netzwerk und wurde anschließend auf dem ISPRS Potsdam-Datensatz trainiert. Im Anschluss hat ein Fine-Tuning auf Daten der hier verwendeten Quelldomäne stattgefunden. Während des Trainings des Cycle-Consistent Adversarial Networks kommen zwei zunächst gleiche Segmentierungsnetzwerke zum Einsatz. Das erste Netzwerk  $Seg_S$  bleibt während des gesamten Trainingsvorgangs für die Erstellung von Annotationen unverändert und das zweite Netzwerk  $Seg_T$  wird an die Zieldomäne angepasst. Im ersten Schritt wird das Cycle-Consistent Adversarial Network für mehrere Epochen trainiert, bis sich stabile Bildübersetzungen für  $G$  und  $F$  ergeben haben. Im zweiten Schritt werden, basierend auf Bildern der Quelldomäne, die entsprechenden Übersetzungen  $G(s)$  erzeugt. Außerdem werden noch entsprechende Pseudoannotationen  $P(s)$  mit Hilfe von  $Seg_S$  erstellt. Nun wird  $Seg_T$  auf Grundlage von  $G(s)$  und  $P(s)$  nachtrainiert. Um dabei ein stabiles Training zu gewährleisten und um eine interne Verschiebung der Objektklassen zu verhindern [64] werden zusätzlich noch Bilder der Quelldomäne ohne ein zusätzliches Anwenden von  $G$  in das Training von  $Seg_T$  integriert (siehe Abbildung 21). Dieses Trainingschema wird in abwechselnder Form fortgeführt. Dabei wird regelmäßig das Cycle-Consistent Adversarial Network trainiert und auf Grundlage der jeweiligen Variante

von  $G$  das neuronale Netz  $\text{Seg}_T$  entsprechend nachtrainiert. Auf diese Weise werden bei  $\text{Seg}_T$  unterschiedlichste Varianten von  $G(s)$  für das Training eingesetzt (siehe Abbildung 22).

## 5.2 Ergebnisse und Diskussion

In einem ersten Schritt wurde das Netzwerk für die semantische Segmentierung basierend auf Daten der Quelldomäne trainiert und erreichte eine Gesamtgenauigkeit von 88,32% und einen  $F_1$ -Wert von 0,88. Ein robustes Segmentierungsnetzwerk mit einer hohen Klassifikationsgenauigkeit ist dabei von essenzieller Bedeutung um Pseudoannotationen mit ausreichender Qualität während des Trainingsvorgangs zu erzeugen.

Zunächst wurde überprüft, wie gut die Ergebnisse sind, wenn die Methode zur semantischen Segmentierung auf Daten aus der Zieldomäne ohne weitere Verarbeitung angewendet wird. Dabei ergab sich eine Gesamtgenauigkeit von 24,76% und ein  $F_1$ -Wert von 0,25. Wie in Abbildung 23 exemplarisch zu erkennen ist, ist ohne eine Anpassung praktisch keine korrekte Objektklassenzuordnung möglich. Außerdem wurden zum Vergleich unserer Methode zwei zusätzliche Varianten zur Domain Adaptation untersucht. Dabei handelt es sich um die Style-Transfer-Methode und die Fine-Tuning-Methode.

Bei der Style-Transfer-Methode wurden nach Beendigung des Trainings die Evaluationsdaten der Zieldomäne auf Grundlage des Generators  $F$  übersetzt, um dem Charakter des Quelldatensatzes zu entsprechen. Diese Daten  $F(t)$  wurden anschließend von  $\text{Seg}_S$  klassifiziert und es ergab sich eine Gesamtgenauigkeit von 63,82% und ein  $F_1$ -Wert von 0,62. Dies stellt eine deutliche Verbesserung zu dem Szenario dar, in welchem keine Anpassung durchgeführt wurde. Während bei der Style-Transfer-Methode das Segmentierungsnetzwerk nicht nachtrainiert wird, ist bei der Fine-Tuning-Methode eine Anpassung dieser Methode vorgesehen. Dabei werden die annotierten Trainingsdaten der Quelldomäne mit dem Generator  $G$  nach Beendigung des Trainings des Cycle-Consistent Adversarial Networks zu  $G(s)$  übersetzt und auf Grundlage dieser angepassten Bilder für das Nachtraining von  $\text{Seg}_T$  verwendet. Anschließend wird das nachtrainierte Netzwerk auf die Evaluationsdaten zu Klassifizierung angewandt. Dabei ergab sich eine Gesamtgenauigkeit von 60,95% und ein  $F_1$ -Wert von 0,63. Bei der hier vorgestellten kontinuierlichen

Trainingsmethode wurden während des Trainings verschiedene Varianten von  $G$  verwendet um auf Grundlage der Pseudoannotationen das Segmentierungsnetzwerk  $\text{Seg}_T$  zu trainieren. Dabei ergaben sich für die Gesamtgenauigkeit 76,16% und für den  $F_1$ -Wert ein Ergebnis von 0,77. Die Ergebnisse sind in Tabelle 9 und Tabelle 10 zusammengefasst und in Abbildung 23 dargestellt. Es konnte gezeigt werden, dass durch den Einsatz von Cycle-Consistent Adversarial Networks ohne vorhandene annotierte Trainingsdaten sowohl in der Style-Transfer- als auch in der Fine-Tuning-Methode die Genauigkeit der Klassifikationsergebnisse gegenüber der Variante ohne Anpassung deutlich gesteigert werden konnte. Durch das hier vorgestellte und eingesetzte kontinuierliche Nachtraining von  $\text{Seg}_T$  mit unterschiedlichen Varianten von  $G$  konnte gegenüber der Style-Transfer-Methode eine weitere Verbesserung der Gesamtgenauigkeit von 19,34% und von 24,19% beim  $F_1$ -Wert erreicht werden. Die Verbesserungen gegenüber der Fine-Tuning-Methode beliefen sich auf 24,95% bei der Gesamtgenauigkeit und 22,22% beim  $F_1$ -Wert. Somit konnte gegenüber beiden Referenzmethoden eine deutliche Verbesserung der Auswertergebnisse erreicht werden. Dies lässt sich darauf zurückführen, dass durch das kontinuierliche Nachtrainieren von  $\text{Seg}_T$  viele unterschiedliche Bildübersetzungen basierend auf unterschiedlichen Varianten von  $G$  zum Einsatz kamen. Bei dem unüberwachten Training eines Cycle-Consistent Adversarial Networks ist es schwierig, aufgrund von fehlenden Annotationsdaten die optimale Variante der Generatoren für einen passenden Style-Transfer zu finden. Durch das kontinuierliche Trainieren der Generatoren auf unterschiedlichen Varianten wurde eine deutlich höhere Generalisierung der Auswertmethode erreicht, was wiederum die Ergebnisse gegenüber den Referenzmethoden, welche nur auf den finalen Generatoren beruhen, verbessern konnte.

Tabelle 9: Ergebnisse der Style-Transfer-Methode, der Fine-Tuning-Methode und der kontinuierlichen Trainingsmethode als Gesamtgenauigkeit in Prozent.

| Methoden                | Gesamtgenauigkeit [%] |
|-------------------------|-----------------------|
| Keine Anpassung         | 24,76                 |
| Style-Transfer-Methode  | 63,82                 |
| Fine-Tuning-Methode     | 60,95                 |
| Kontinuierliche Methode | 76,16                 |

## Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

*Tabelle 10: Ergebnisse der Style-Transfer-Methode, der Fine-Tuning-Methode und der kontinuierlichen Trainingsmethode als  $F_1$ -Wert.*

| <b>Methode</b>          | <b><math>F_1</math>-Wert</b> |
|-------------------------|------------------------------|
| Keine Anpassung         | 0,25                         |
| Style-Transfer Methode  | 0,62                         |
| Fine-Tuning Methode     | 0,63                         |
| Kontinuierliche Methode | 0,77                         |

Domain Adaptation mit Cycle-Consistent Adversarial Networks und kontinuierlicher Anpassung

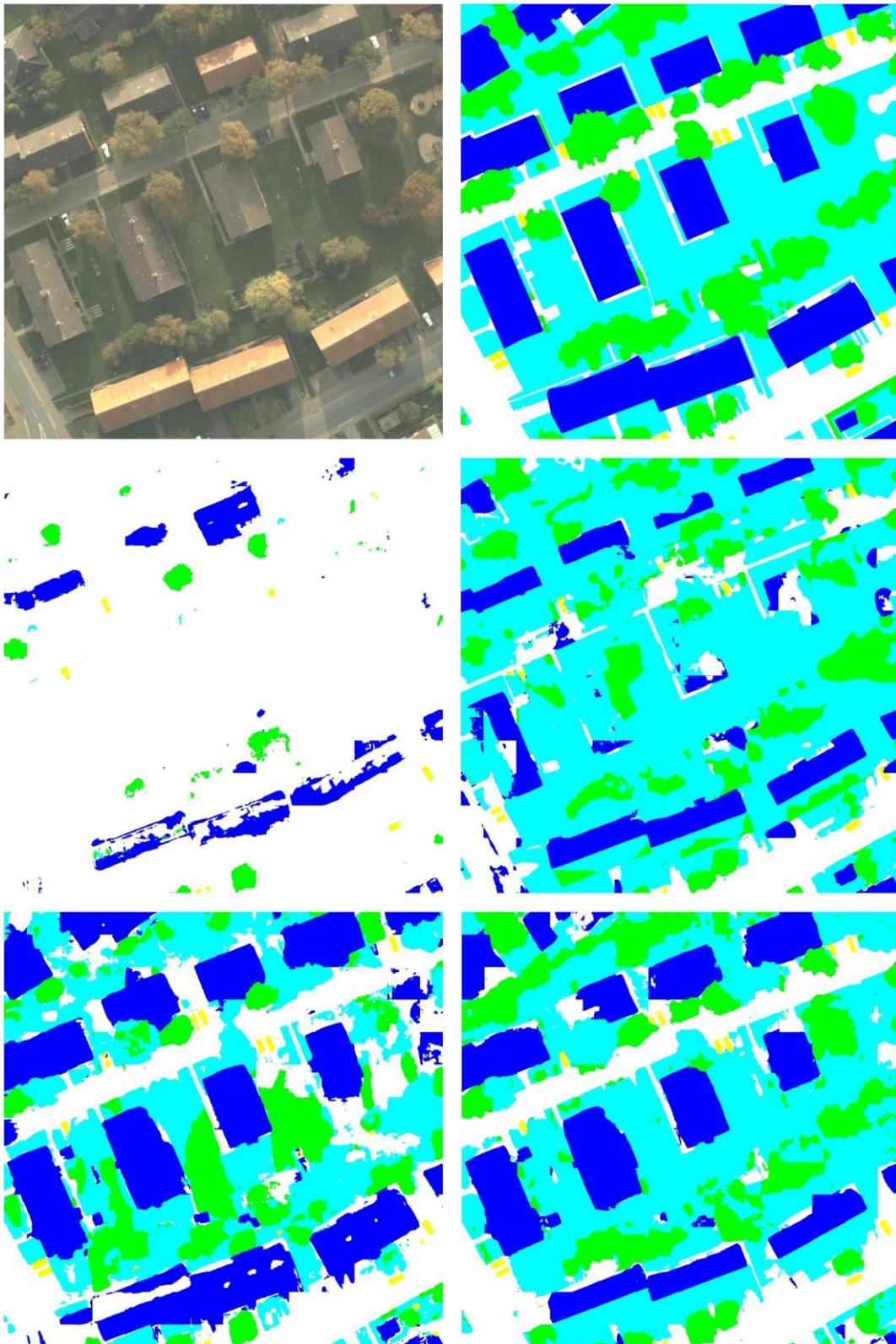


Abbildung 23: Obne links: RGB-Bild des Luftbilddatensatz Herbst. Oben rechts: Ground Truth, Mitte links: Ergebnis ohne Anpassung. Mitte rechts: Ergebnis der Style-Transfer-Methode. Unten links: Ergebnis der Fine-Tuning-Methode. Unten rechts: Ergebnis der kontinuierlichen Methode.



## 6 Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

In diesem Kapitel wird vorgestellt, wie unter Einsatz von Vegetationsindices in Kombination mit neuronalen Netzen zur semantischen Segmentierung eine teilautomatisierte Ground Truth für den Luftbilddatensatz Sommer erstellt werden kann (siehe Abbildung 24). Dieses Verfahren bedarf keiner manuell annotierten Daten für das Training des entsprechenden CNNs. Ziel dieser Methode ist das Verbessern der Ground Truth-Erstellung basierend auf dem Einsatz von Vegetationsindices. Auf Grundlage dessen wird ein Transfer von spektralen Informationen auf die Gewinnung räumlicher Zusammenhänge durchgeführt. Durch dieses Verfahren soll ein erster Schritt für einen qualitativ hochwertigen Quelldomäne-Datensatz zur Domain Adaptation für die Segmentierung von Vegetation geschaffen werden. Evaluiert wurde die Methode auf manuell annotierten Daten der Quelldomäne. Im Bereich der Klassifikation der Bodenbedeckung spielt die Segmentierung von Vegetation eine wichtige Rolle für viele Anwendungsbereiche. Ein wichtiger

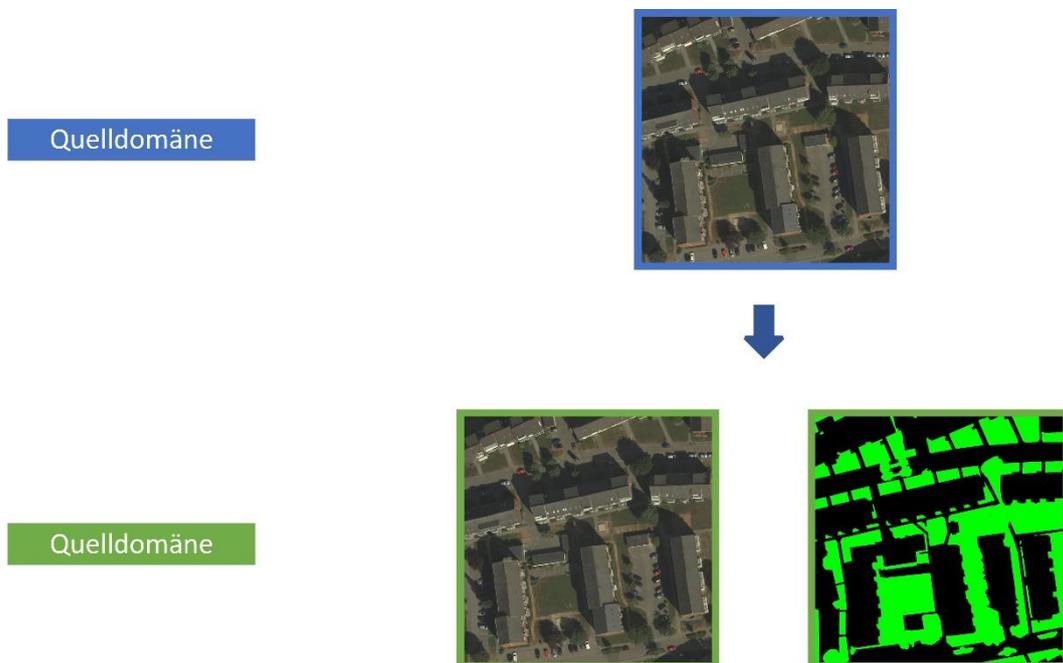


Abbildung 24: Auf Grundlage von Vegetationsindices wird durch den Einsatz von CNNs zur semantischen Segmentierung eine teilautomatisierte Annotation ermöglicht.

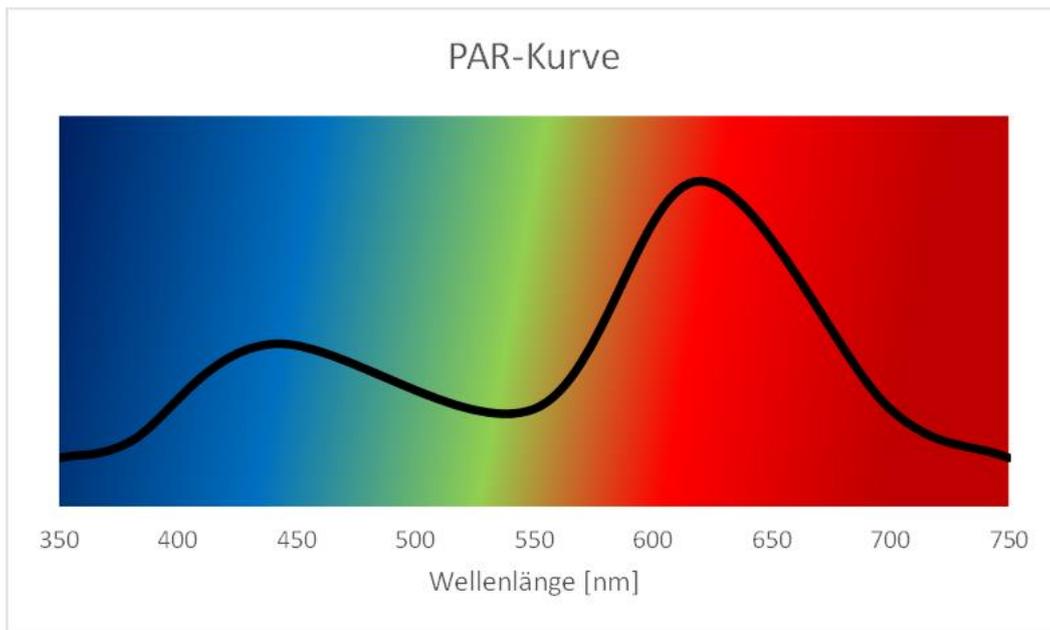


Abbildung 25: Darstellung der PAR-Kurve. Aufgetragen ist dabei schematisch die relative Quantenausbeute über die Wellenlänge in einem Bereich von 350 bis 750 nm.

Faktor ist in diesem Zusammenhang die Analyse von Vegetationsstrukturen in Bildaufnahmen auf Grundlage der spezifischen spektralen Eigenschaften. Diese lassen sich auf den Prozess der Photosynthese zurückführen. Die Photosynthese ist bei vielen Pflanzen ein wichtiger physiologischer Prozess um Lichtenergie in chemisch gebundene Energie umzuwandeln. Die Lichtabsorption wird in grünen Pflanzen auf Grundlage von Farbstoffen wie Chlorophyll umgesetzt. Die bei diesen Pflanzen vorkommenden Chlorophylle a und b absorbieren hauptsächlich Licht im blauen und roten Spektrum, wobei es somit zur entsprechenden Bildung eines charakteristischen Maximums im jeweiligen Bereich kommt. Im grünen Spektrum findet allerdings keine Absorption statt und somit kommt es zu einer Lücke in diesem Wellenlängenbereich, welche verantwortlich ist für die charakteristische grüne Farbe vieler Pflanzen, da Licht in dem entsprechenden Farbspektrum reflektiert wird. An der Photosynthese sind gebundene Chlorophylle, welche Unterschiede zu den freien Chlorophyllen aufweisen, und noch weitere lichtabsorbierende Stoffe beteiligt. Somit kommt es zu einem komplexeren Absorptionsspektrum [65]. Der Spektralbereich, in dem Licht von Pflanzen absorbiert und für die Photosynthese eingesetzt wird, nennt man photosynthetisch aktive Strahlung. Diese wird dargestellt wie in Abbildung 25 als sogenannte PAR-Kurve, wobei dabei die relative Quantenausbeute über die Wellenlänge aufgetragen wird. Der Bereich der photosynthetisch aktiven Strahlung liegt vor allem in dem

Wellenlängenbereich zwischen 400 und 700 nm. Außerhalb dieses Bereichs findet fast keine Lichtabsorption statt [66]. Es befindet sich ein Maximum im roten Spektralbereich bei einer Wellenlänge von ca. 620 nm und ein weiteres Maximum im blauen Bereich bei ca. 440 nm. Zusätzlich lässt sich ein Minimum im grünen Spektralbereich bei ca. 550 nm erkennen. Auf Grundlage dieser spezifischen biophysikalisch gegebenen Eigenschaften können nun Rückschlüsse auf die Vegetationsbedeckung innerhalb von Bildaufnahmen mit entsprechender spektraler Auflösung gezogen werden. Eine etablierte Möglichkeit diese Zusammenhänge zu verwenden, ist das Berechnen von Vegetationsindices. Dabei handelt es sich um spezifische Relationen zwischen den einzelnen Kanälen um pixelweise Rückschlüsse auf die Stärke der Vegetationsbedeckung zu erhalten. Ein häufig eingesetzter Vegetationsindex ist dabei der sogenannte Normalized Difference Vegetation Index (NDVI) [23]. Dieser Index beruht auf der Relation zwischen dem Wert des NIR-Kanals, welcher wegen der hohen Reflexion entsprechend groß ist und dem Wert des Rotkanals R, welcher aufgrund der hohen Absorption in diesem Spektralbereich wiederum sehr gering ausfällt. Der NDVI berechnet sich wie folgt:

$$NDVI = \frac{NIR - R}{NIR + R}$$

Durch Hinzufügen der Summe im Nenner erreicht man, dass sich der Wertebereich des NDVI zwischen -1 und 1 bewegt, wobei Werte größer als 0 auf eine Vegetationsbedeckung hindeuten [23]. Um aus einem solchen Indexbild eine Segmentierung zu erhalten, kann ein Schwellwert festgelegt werden, um das Bild in die Objektklassen Vegetation und Hintergrund zu unterteilen.

Obwohl durch den Einsatz von unüberwachter Domain Adaptation wie in Kapitel 5 beschrieben, der Annotationsaufwand der Zieldomäne umfangreich reduziert werden kann, besteht trotzdem noch die Notwendigkeit einer umfangreichen manuellen Annotation des Quelldomäne-Datensatzes. Durch den Einsatz von Vegetationsindices und der Wahl eines geeigneten Schwellwerts kann für Vegetation bereits eine Selektion infrage kommender Pixel in dem entsprechenden Datensatz getroffen werden. Der Nachteil dieser Methode ist allerdings, dass ausschließlich eine pixelweise Zuordnung ohne den Einbezug von Werten benachbarter Pixel getroffen wird. Diese räumlichen Informationen können

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

allerdings zu einer genaueren Klassifikation einzelnen Pixel beitragen, da auch Vegetation unterhalb des gewählten Schwellwerts vorkommen kann.

Im Folgenden wird eine Methode beschrieben zur Untersuchung und Verbesserung der indexbasierten Segmentierung von Vegetation auf dem Luftbilddatensatz Sommer. Dazu findet eine Übertragung von Informationen aus den Vegetationsindexbildern auf die entsprechenden RGB-Bilder statt, um unter Einsatz eines CNNs räumliche Verhältnisse zu erkennen und somit die Segmentierung auf Grundlage der Vegetationsindices zu verbessern und einen qualitativ hochwertigen Quelldomäne-Datensatz zu erstellen.

## 6.1 Methode

Die Methode basiert darauf, dass durch den Einsatz von CNNs, ohne zusätzlich manuell annotierte Daten, die Segmentierungsqualität von Vegetationsflächen durch das Berechnen von Vegetationsindices verbessert werden kann. Dazu wird das Resultat basierend auf einem einfachen Schwellwertverfahren in Verbindung mit Vegetationsindices zur Ermittlung von Vegetationsflächen in einen CNN-Segmentierungsalgorithmus integriert um diese Flächen zu korrigieren. Dazu werden drei Vegetationsindices als Grundlage für den Algorithmus angewandt, auf dem Luftbilddatensatz Sommer untersucht und mit dem jeweiligen Ergebnis verglichen. Die Methode besteht aus insgesamt drei Schritten:

Schritt 1: Erstellen eines Trainingsdatensatzes für das RGB-Segmentierungsnetzwerk

Schritt 2: Trainieren des RGB-Segmentierungsnetzwerks

Schritt 3: Erstellen des Quelldomäne-Datensatzes

Die Inhalte der einzelnen Schritte des Workflows werden im Folgenden erläutert.

### **Schritt 1: Erstellen eines Trainingsdatensatzes für das RGB-Segmentierungsnetzwerk**

Zunächst werden für die Erstellung der Trainingsdaten für das RGB-Segmentierungsnetzwerk entsprechende Vegetationsindexbilder erstellt. Dies wird auf Grundlage unterschiedlicher Vegetationsindices untersucht. Zum einen wird die Methode auf Grundlage der Werte des NDVI, wie oben bereits beschrieben, durchgeführt. Neben dem NDVI existieren noch weitere Vegetationsindices, welche durch eine komplexere und individuellere

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

Berechnung auch eine differenziertere Vegetationssegmentierung zulassen. Eine Variante ist der sogenannte Soil-Adjusted Vegetation Index (SAVI), welcher den Einfluss von Erde bei der Berechnung reduziert [67]. Dies wird erreicht durch das Hinzufügen einer Konstanten  $L$  und ergibt sich zu:

$$SAVI = \frac{NIR - R}{NIR + R + L} (1 + L)$$

Wobei  $L$  dabei manuell gewählt werden muss. Bei einer sehr geringen Vegetationsdichte wird  $L=1$  gesetzt, bei einer mittlerer Vegetationsdichte  $L=0,5$  und bei einer hohen Vegetationsdichte  $L=0,25$ . Im Folgenden wird bei der Berechnung  $L=0,5$  gesetzt. Ein weiterer häufig eingesetzter Vegetationsindex ist der Enhanced Vegetation Index (EVI), welcher neben dem roten Kanal auch noch den blauen Kanal verwendet um den Einfluss von Aerosolen zu minimieren [68]. Der EVI berechnet sich als:

$$EVI = G \times \frac{NIR - R}{NIR + C_1 \times R - C_2 \times B + L}$$

Mit  $G$  als Gain-Faktor von 2,5,  $C_1=6$  und  $C_2=7,5$  als Koeffizienten zur Anpassung des roten und blauen Kanals und  $L=1$  als Canopy-Background-Adjustment. Der Wertebereich ist sowohl beim SAVI wie auch beim EVI von der Normalisierung der Eingangsdaten abhängig. Nach der Berechnung des entsprechenden Vegetationsindexbildes für jedes der Trainingsdatenbilder, wird auf diese ein Schwellwert angewandt. Dieser sollte so gewählt werden, dass der Wert für die Precision bei der Klasse Vegetation möglichst hoch ist, bei ausreichendem Recall. D.h., es besteht nicht die Notwendigkeit ein bestmöglichstes Verhältnis zwischen Precision und Recall zu finden, wie es bei einem einfachen Schwellwertverfahren notwendig ist, sondern es sollte möglichst ein höherer Schwellwert  $SW$  gewählt werden um die Anzahl an Pixeln, welche fälschlicherweise als Vegetation gekennzeichnet werden zu minimieren. Dennoch müssen ausreichend Beispielpixel für Vegetation verbleiben. Das daraus resultierende Schwellwertbild wird nun als unvollständige Ground Truth für Vegetation  $uGT_{Vegetation}$  behandelt.

$$uGT_{Vegetation} = \text{Vegetationsindex} \geq SW$$

Dementsprechend ist der Precision-Wert für die Klasse Vegetation möglichst hoch und die unsicheren Vegetationspixel mit niedrigem Indexwert fallen in die Klasse Hintergrund. Im

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

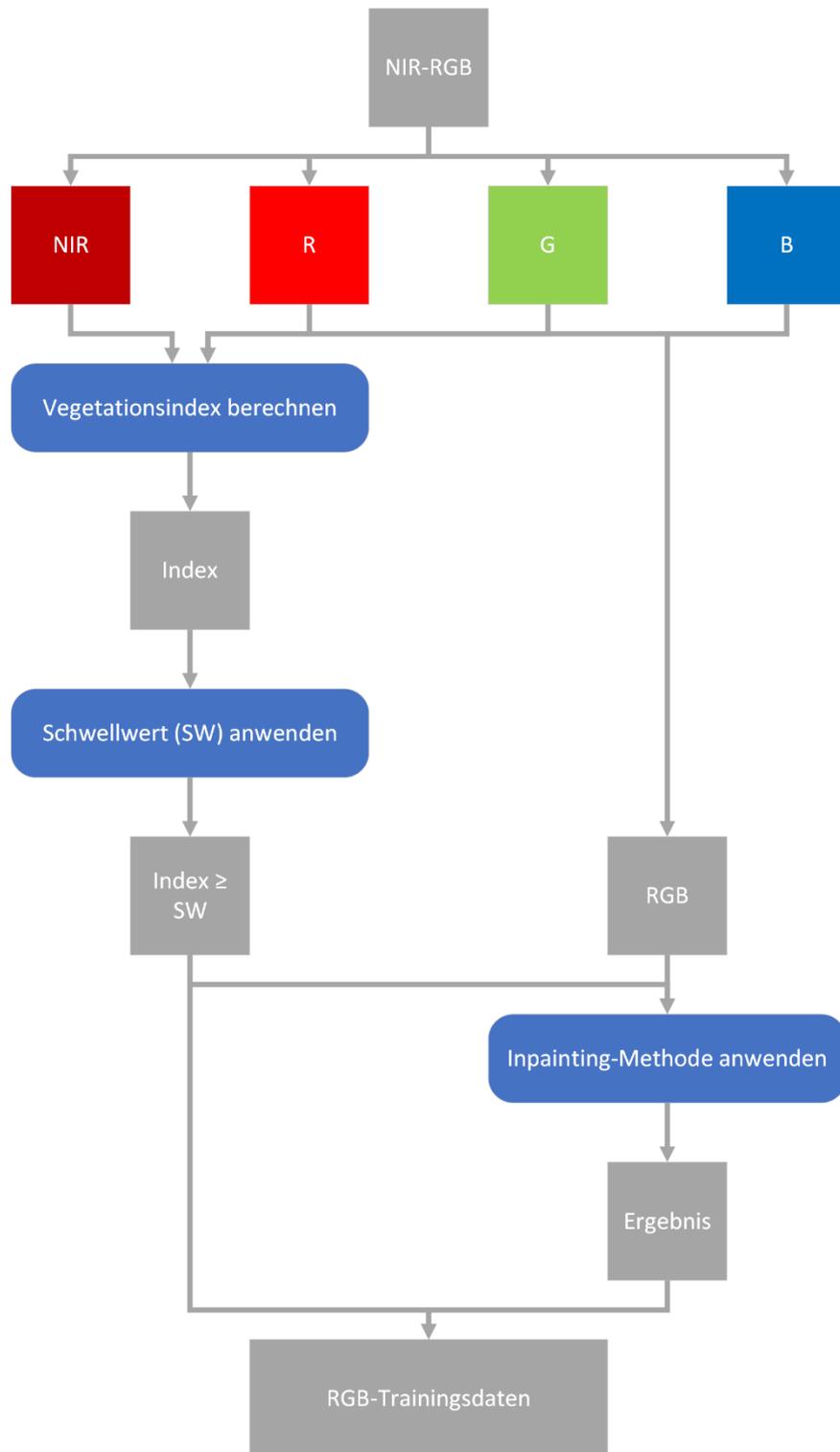


Abbildung 26: Erstellen eines Trainingsdatensatzes für das RGB-Segmentierungsnetzwerk.

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

Anschluss wird ein Median-Filter mit einer Größe von  $15 \times 15$  auf das Ergebnis angewandt. Nun müssen noch die RGB-Bilder für das RGB-Segmentierungsnetzwerk an die unvollständige Ground Truth angepasst werden. Dazu wird zunächst sämtliche Vegetation aus dem Bild entfernt mit einem Schwellwert von 0, um eine unvollständige Ground Truth für den Hintergrund  $uGT_{Hintergrund}$  zu erhalten mit möglichst wenig Vegetation:

$$uGT_{Hintergrund} = \text{Vegetationsindex} < 0$$

Ohne eine entsprechende Annotation bleiben nun im Folgenden die entsprechenden Pixel übrig, für welche der folgende Bereich des Vegetationsindexes gilt:

$$0 \leq \text{Vegetationsindex} < SW$$

Diese Pixel können ohne eine manuelle Annotation nicht mit ausreichend hoher Wahrscheinlichkeit den Klassen Vegetation oder Hintergrund zugeordnet werden. Deshalb werden diese im Anschluss entfernt und durch Pixel mit den Eigenschaften der Klasse Hintergrund ersetzt. Dies geschieht durch den Einsatz von Inpainting, wobei in diesem Fall Coherent Inpainting eingesetzt wurde [69]. Bei diesem Verfahren wird ein bestimmter Bereich eines Bildes entfernt und durch eine Textur ersetzt, welche den Eigenschaften des übrigen Bildes entspricht. Dieses Verfahren wird auf die RGB-Bilder angewandt, mit einem Füllen aller Pixel welche nicht  $uGT_{Hintergrund}$  entsprechen. Anschließend werden alle Pixel des Bildes, welche  $uGT_{Vegetation}$  entsprechen durch die Pixel des ursprünglichen RGB-Bildes ersetzt und man erhält das finale angepasste RGB-Bild. Der RGB-Segmentierungstrainingsdatensatz enthält nun die Klassen Vegetation mit Flächen, welche mit hoher Wahrscheinlichkeit entsprechend des Schwellwerts korrekt zugeordnet sind und die Klasse Hintergrund, welche zum einen Flächen mit der Bedingung  $0 < \text{Vegetationsindex}$  besitzt und den entsprechend berechneten Bereichen aus dem Inpainting-Verfahren. Der Ablauf ist in **Fehler! Verweisquelle konnte nicht gefunden werden.** dargestellt.

## Schritt 2: Trainieren des RGB-Segmentierungsnetzwerks

zweiten Schritt wird auf Grundlage des RGB-Segmentierungsnetzwerkdatensatzes das CNN für die Vegetationssegmentierung trainiert (siehe Abbildung 27). Als Segmentierungsnetzwerk wird die Architektur von U-Net eingesetzt [48]. Als Trainingsparameter

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs



Abbildung 27: Trainieren des RGB-Segmentierungsnetzwerks.

wurde eine Batch-Größe von 32 und der Adam-Optimierer [61] mit einer Lernrate von 0,001 verwendet.

### Schritt 3: Erstellen des Quelldomäne-Datensatzes

Im dritten Schritt wird das RGB-Segmentierungsnetzwerk, welches auf den bearbeiteten Bildern von Luftbilddatensatz Sommer trainiert wurde, auf die ursprünglichen Daten angewandt. Daraus resultiert nun ein Klassifikationsergebnis welches als Zwischenschritt für die Ground Truth verwendet werden kann. Da in Schritt 1 mit der Bestimmung eines Schwellwerts zur Erzeugung einer unvollständigen Ground Truth für Vegetation  $uGT_{Vegetation}$  gewählt wurde, welche auch Grundlage für das Training des RGB-Segmentierungsnetzwerk war, werden dementsprechend zusätzlich alle Pixel des Klassifikationsergebnisses, welche bei der Klasse Vegetation nicht  $uGT_{Vegetation}$  entsprechen durch diese ersetzt, da diese aufgrund des hoch gewählten Vegetationsindex mit großer Zuverlässigkeit als Vegetation angenommen werden können. Die Pixel der Klasse Hintergrund in diesem Zusammenhang bleiben unverändert. Somit stellt die finale Ground Truth eine Vereinigungsmenge des Klassifikationsergebnisses basierend auf den RGB-Daten und  $uGT_{Vegetation}$  in Hinblick auf die Klasse Vegetation dar. Anschließend stellen die NIR-RGB-Bilder und die entsprechende neu ermittelte Ground Truth den teilautomatisiert erstellten Quelldomäne-Datensatz dar (siehe Abbildung 28).

## Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

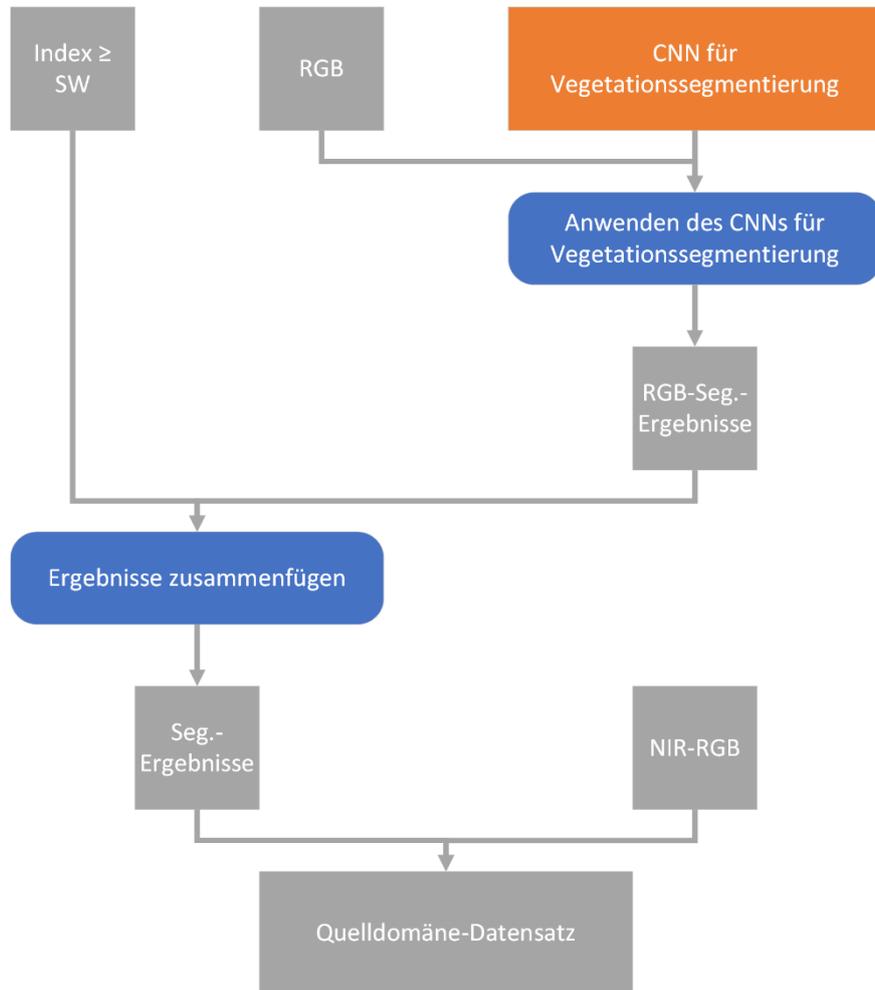


Abbildung 28: Erstellen des Quelldomäne-Datensatzes.

### 6.2 Ergebnisse und Diskussion

Die Methode wurde auf Grundlage der drei Vegetationsindices NDVI, SAVI und EVI durchgeführt und jeweils auf den Luftbilddatensatz Sommer angewandt. Dabei wurde zum Ermitteln eines Vergleichs jeweils zuerst auf Grundlage der manuellen Annotationsdaten des Datensatzes das bestmögliche Ergebnis mit einem einfachen Schwellwerts ermittelt. Als Referenz galt dabei der  $F_1$ -Wert. Dazu wurde der Schwellwert in Schritten von 0,01 von 0 bis 1 erhöht und jeweils auf Basis der manuellen Annotationen die  $F_1$ -Werte basierend auf dem jeweiligen Schwellwert berechnet. Dieser galt dabei im Folgenden als

## Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

Vergleichswert zu der CNN-basierten Methode. Diese wurde je Vegetationsindex auf Grundlage von sechs Schwellwertgrenzen in Schritten von 0,1 von 0 bis 0,5 durchgeführt.

### 6.2.1 Ergebnisse auf Grundlage des NDVI

Zunächst wurde nach dem besten  $F_1$ -Wert für eine Segmentierung auf Grundlage der Schwellwertmethode mittels NDVI gesucht. Dabei ergab sich der höchste Wert von 0,85 bei einem Schwellwert von 0,25 mit einem Wert von 0,74 für die IoU. Die Werte für die Precision und den Recall sind zusammen mit den Werten für die Methode in Abbildung 29 dargestellt. Für die CNN-Methode ergab sich der beste  $F_1$ -Wert mit 0,86 bei einem Schwellwert von 0,4 und einer IoU von 0,75. Die Ergebnisse der CNN-Methode für alle untersuchten Schwellwerte sind in Tabelle 11 zusammengefasst. Eine exemplarische Darstellung der Ergebnisse ist in Abbildung 30 gegeben.

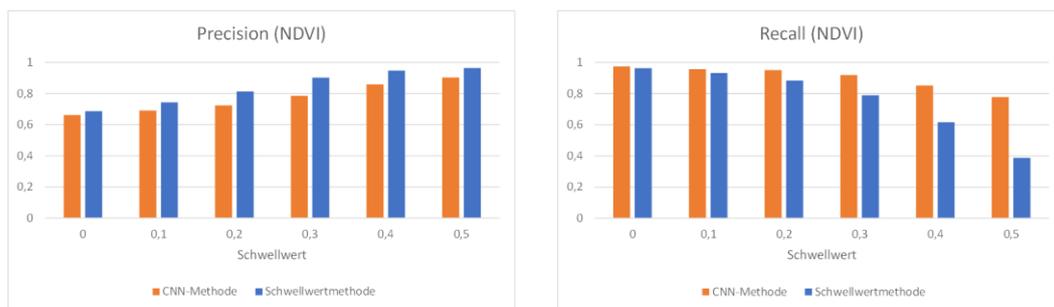


Abbildung 29: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem NDVI im Vergleich zum Schwellwertverfahren angegeben als  $F_1$ -Wert und der IoU.

Tabelle 11: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem NDVI angegeben als  $F_1$ -Wert und der IoU.

| Schwellwert | 0    | 0,1  | 0,2  | 0,3  | 0,4  | 0,5  |
|-------------|------|------|------|------|------|------|
| $F_1$ -Wert | 0,79 | 0,80 | 0,82 | 0,85 | 0,86 | 0,84 |
| IoU         | 0,65 | 0,67 | 0,70 | 0,74 | 0,75 | 0,72 |

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

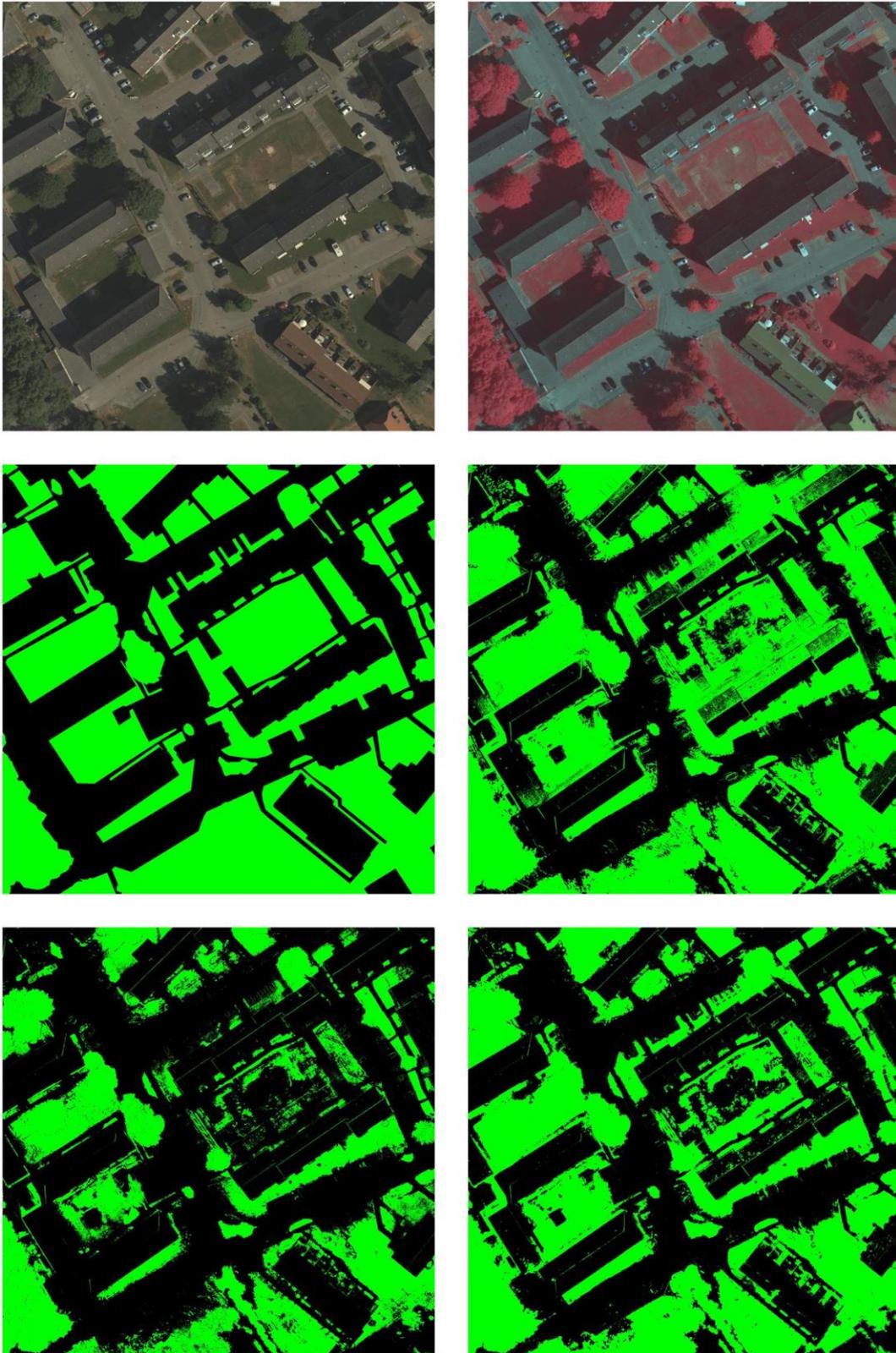


Abbildung 30: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von NDVI-Schwellwertverfahren bei 0,25. Unten links: Ergebnis von NDVI-Schwellwertverfahren bei 0,4. Unten rechts: Ergebnis von entsprechendem CNN-Verfahren.

### 6.2.2 Ergebnisse auf Grundlage des SAVI

Der beste Wert auf Grundlage der Schwellwertmethode ergab sich für das Anwenden des SAVI bei einem Wert von 0,06 mit einem  $F_1$ -Wert von 0,88 und einer IoU von 0,78. Bei Anwenden der CNN-Methode ergab sich der beste  $F_1$ -Wert von 0,91 bei einem Schwellwert von 0,1 und einer IoU von 0,83. Die jeweiligen Werte zu Precision und Recall der CNN-Methode und der Schwellwertmethode sind in Abbildung 31 dargestellt und die Ergebnisse für die Segmentierung basierend auf dem SAVI in Tabelle 12 zusammengefasst. Die Ergebnisse der Segmentierung sind beispielhaft in Abbildung 32 dargestellt.

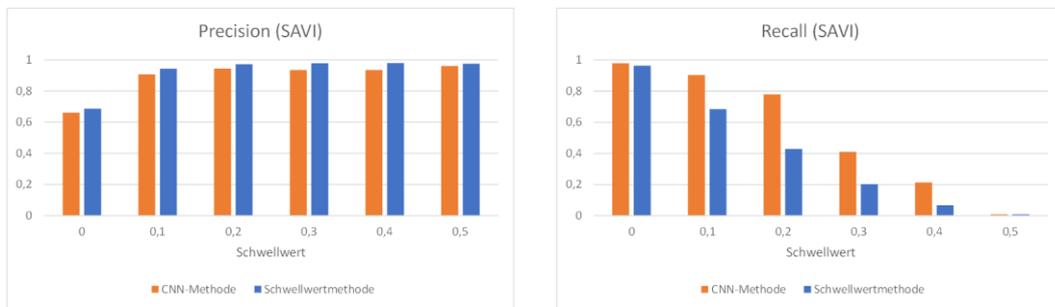


Abbildung 31: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem SAVI im Vergleich zum Schwellwertverfahren angegeben als  $F_1$ -Wert und der IoU.

Tabelle 12: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem SAVI angegeben als  $F_1$ -Wert und der IoU.

| Schwellwert                  | 0    | 0,1  | 0,2  | 0,3  | 0,4  | 0,5  |
|------------------------------|------|------|------|------|------|------|
| <b><math>F_1</math>-Wert</b> | 0,79 | 0,91 | 0,85 | 0,57 | 0,35 | 0,02 |
| <b>IoU</b>                   | 0,65 | 0,83 | 0,74 | 0,40 | 0,21 | 0,01 |

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

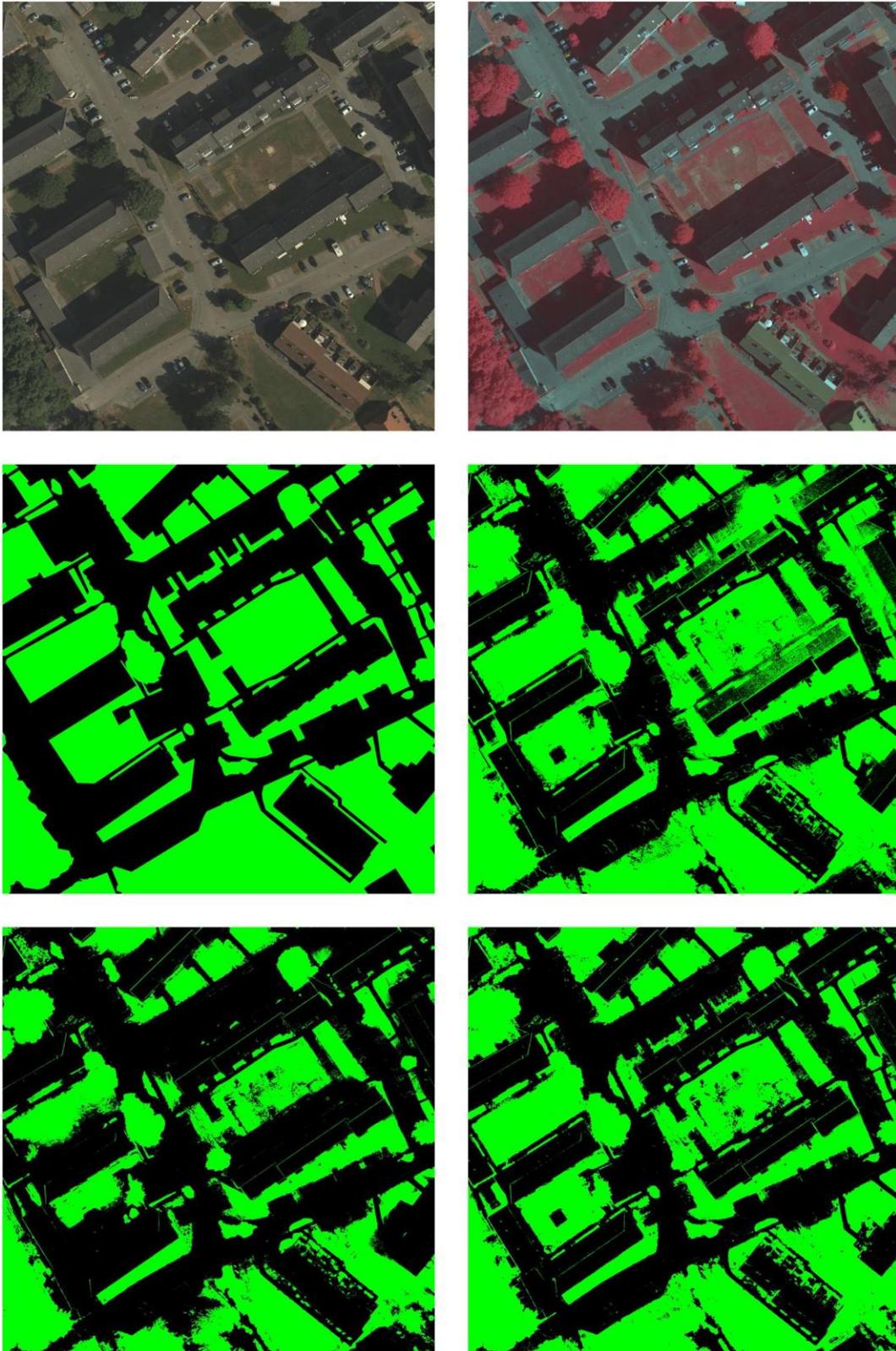


Abbildung 32: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von SAVI-Schwelwertverfahren bei 0,06. Unten links: Ergebnis von SAVI-Schwelwertverfahren bei 0,1. Unten recht: Ergebnis von entsprechendem CNN-Verfahren.

## Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

### 6.2.3 Ergebnisse auf Grundlage des EVI

Bei Anwenden des EVI ergab sich für die Schwellwertmethode ein Maximum bei 0,05 mit einem  $F_1$ -Wert von 0,87 und einer IoU von 0,77. Bei Anwenden der CNN-basierten Methode ergab sich der beste Wert bei 0,1 mit einem  $F_1$ -Wert von 0,90 und einer IoU von 0,81. Die Werte für die Precision und den Recall sind in Abbildung 33 dargestellt und in Tabelle 13 zusammengefasst. Eine grafische Darstellung der Segmentierungsergebnisse ist in Abbildung 34 gegeben.

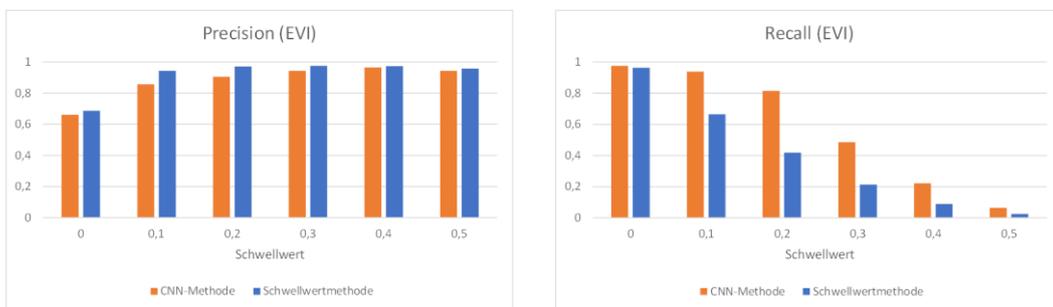


Abbildung 33: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem EVI im Vergleich zum Schwellwertverfahren angegeben als  $F_1$ -Wert und der IoU.

Tabelle 13: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem EVI angegeben als  $F_1$ -Wert und der IoU.

| Schwellwert | 0    | 0,1  | 0,2  | 0,3  | 0,4  | 0,5  |
|-------------|------|------|------|------|------|------|
| $F_1$ -Wert | 0,79 | 0,90 | 0,86 | 0,64 | 0,36 | 0,12 |
| IoU         | 0,65 | 0,81 | 0,75 | 0,47 | 0,22 | 0,06 |

Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

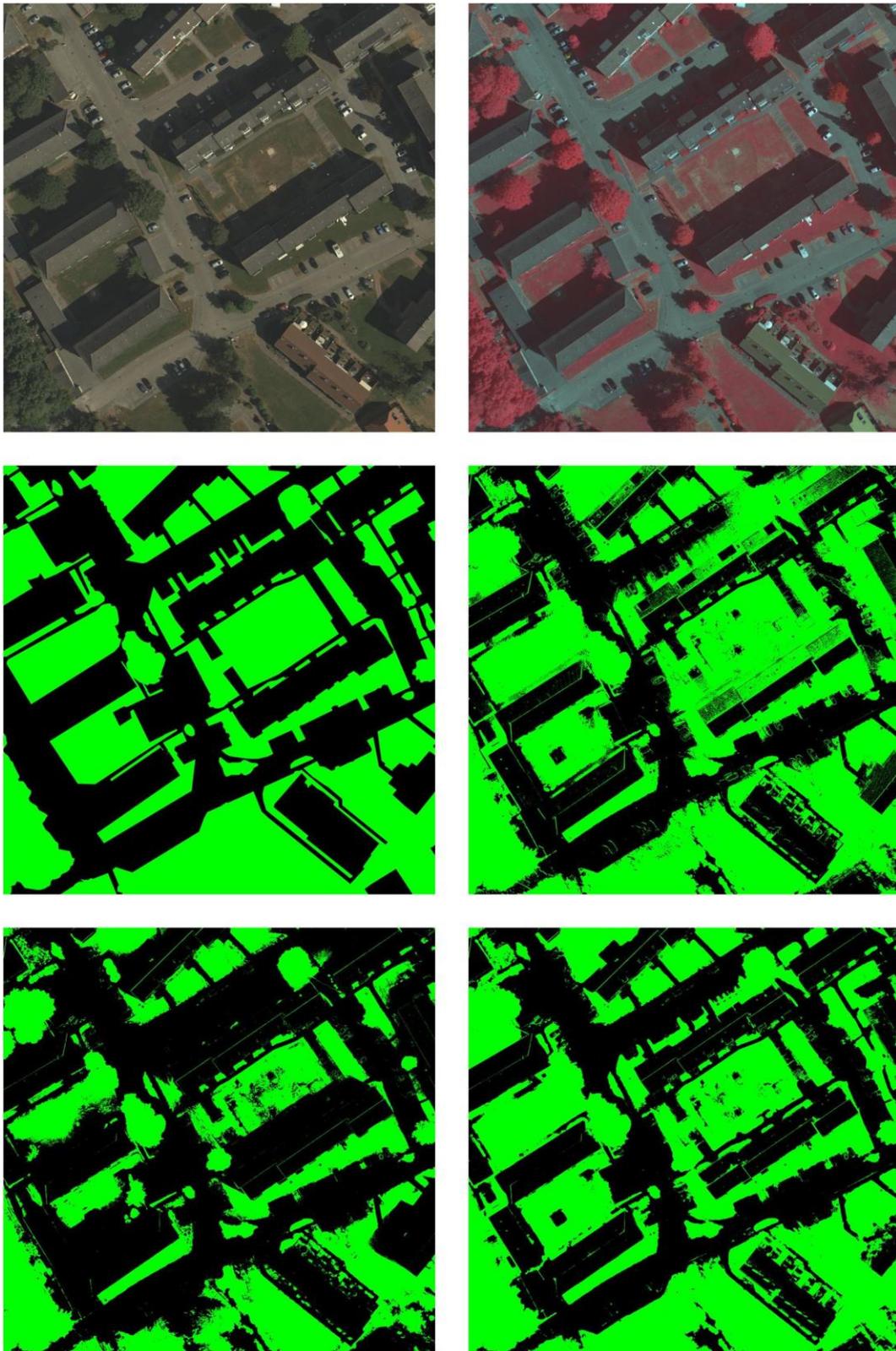


Abbildung 34: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von EVI-Schwellwertverfahren bei 0,05. Unten links: Ergebnis von EVI-Schwellwertverfahren bei 0,1. Unten rechts: Ergebnis von entsprechendem CNN-Verfahren.

## Erstellung eines Quelldomäne-Datensatzes auf Basis von Vegetationsindices und dem Einsatz von CNNs

Bei dem Vergleich zwischen den Precision-Werten der einzelnen Schwellwerte der Vegetationsindices und der hier vorgestellten CNN-basierten Methode angewandt auf den Luftbilddatensatz Sommer hat sich gezeigt, dass die Werte der Schwellwertmethode immer etwas höher sind als die der CNN-basierten Methode. Dies ist insofern nachvollziehbar, da diese bei der CNN-Methode als Ground Truth verwendet wurde. Es hat sich allerdings gezeigt, dass durch die Kombination von Vegetationsindices mit neuronalen Netzen zur semantischen Segmentierung die Qualität der automatisierten Annotation verbessert werden konnte. Der  $F_1$ -Wert konnte im Falle des NDVI um 1,2% verbessert werden, im Falle des SAVI um 3,4% und bei dem EVI um 3,5%. Den besten  $F_1$ -Wert von 0,91 wurde mit dem Anwenden des SAVI erreicht. Diese Verbesserung der Vegetationsindexwerte lässt sich darauf zurückführen, dass durch den Einsatz der CNNs eine bessere räumliche Erkennung der Vegetationsmuster möglich wurde und somit der entsprechende Recall-Wert gesteigert werden konnte. Dieser nimmt bei dem einfachen Anwenden der Schwellwertmethode deutlich schneller als bei der CNN-Methode ab und senkt somit den resultierenden  $F_1$ -Wert. Durch das Verwenden von CNNs in Kombination mit den Vegetationsindices konnte auch bei hohen Schwellwerten auf Grundlage der verbleibenden Pixel ein gutes Ergebnis erzielt werden, da durch das Erlernen der räumlichen Zusammenhänge im CNN-Klassifikationsschritt auch Vegetation mit niedrigeren Indexwerten gefunden werden konnte. Dies war in den Fällen von Vorteil bei welchen die spektralen Kriterien nicht für eine genaue Klassifikation ausreichten, die räumlichen Informationen dies aber ermöglichten. Dementsprechend zeigte sich die CNN-Methode als stabiler bezüglich eines guten Ausgleichs zwischen Precision und Recall-Werten. Aus diesem Ergebnis lässt sich weiterhin schlussfolgern, dass es möglich ist von einer einfachen spektralen Klassifikation basierend auf den Vegetationsindices räumliche Informationen für das Training eines CNNs zu gewinnen um eben diese indexbasierte Segmentierung des hier verwendeten Luftbilddatensatz Sommer als Quelldomäne zu verbessern.

## 7 Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

Im folgenden Kapitel wird eine Methode beschrieben für eine unüberwachte Domain Adaptation für die Segmentierung von Vegetation, basierend auf Cycle-Consistent Adversarial Networks. Der Fokus bei diesem Algorithmus ist es, die spezifischen spektralen Eigenschaften von Vegetation auf NIR-RGB-Bildern auch während der Transformation zwischen Quell- und Zieldomäne zu erhalten. Dazu werden bestimmte vegetationsindexbasierte Metriken in die Fehlerfunktion für das Training des Cycle-Consistent Adversarial Networks integriert. Die Einbindung bezüglich der Segmentierung erfolgt wie in Kapitel 5 beschrieben. Ausgewertet wird anschließend, inwiefern die Werte der jeweiligen Vegetationsindices nach dem Style-Transfer erhalten bleiben und ob sich dadurch die Ergebnisse bezüglich der Segmentierung von Vegetation verbessern lassen. Als Quelldomäne-Datensatz kommt der Luftbilddatensatz Sommer zum Einsatz und als Zieldomäne-Datensätze der Satellitenbilddatensatz Sommer und der Luftbilddatensatz Herbst (siehe Abbildung 35). Es wurden dabei alle vier vorhandenen Farbkanäle verwendet und die

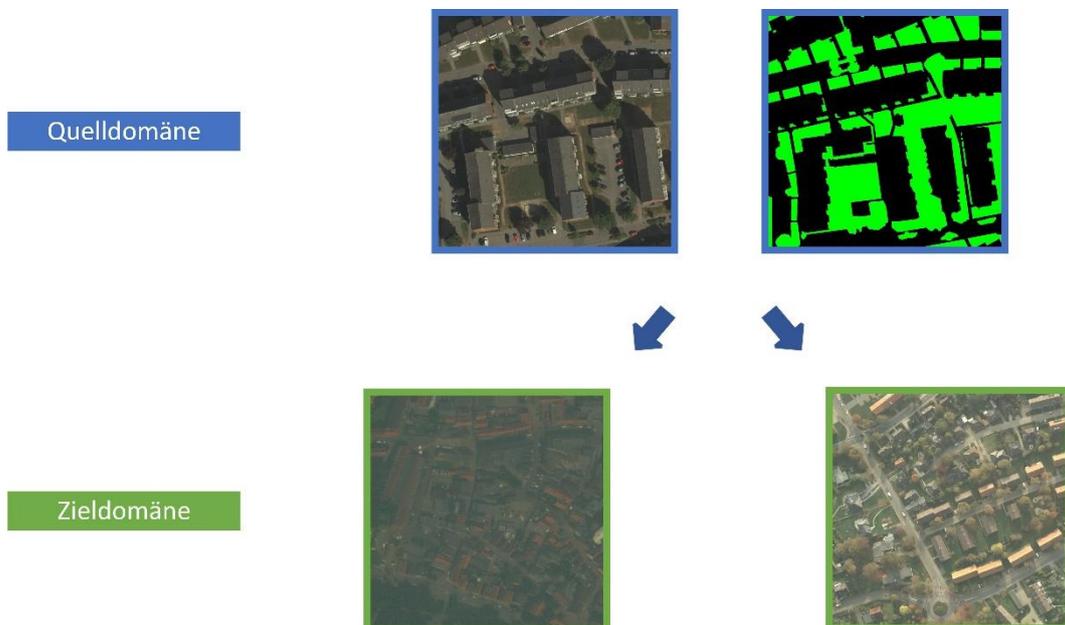


Abbildung 35: Bei dieser Auswertung kamen als Quelldomäne der Luftbilddatensatz Sommer und als Zieldomäne der Satellitenbilddatensatz Sommer und der Luftbilddatensatz Herbst zum Einsatz. Da es sich um unüberwachte Domain Adaptation handelt, waren für das Training der CNNs keine Annotationen in der Zieldomäne vorhanden.

entsprechende Objektklasse Vegetation. Ziel der Konzeption dieser Methode und der Durchführung der Analyse war es zu untersuchen, ob und inwiefern die vegetationspezifischen spektralen Eigenschaften während des Transformationsprozesses verändert werden und wie es möglich ist, diese durch biophysikalisch gegebene Zusammenhänge als Metrik in die Fehlerfunktion für das Training des Cycle-Consistent Adversarial Networks zu integrieren. Weiterhin wurde untersucht, ob sich durch diese Methode auch die Ergebnisse der Segmentierung von Vegetation im Rahmen der Domain Adaptation basierend auf einem CNN zur semantischen Segmentierung verbessern lassen.

## 7.1 Methode

Wie bereits in Kapitel 5 erläutert, stellt das Training von Cycle-Consistent Adversarial Networks für die Domain Adaptation bei Aufgaben der semantischen Segmentierung eine Herausforderung bezüglich der Stabilität des Trainings dar, da keine annotierten Daten verwendet werden. Dies kann dazu führen, dass die übersetzten Bilder zwar visuell betrachtet erfolgreich transformiert wurden, aber bestimmte Charakteristika, welche für eine anschließende semantische Segmentierung essenziell sind, nicht mehr im Bild als passende Information vorhanden sind. Wenn aber für einzelne Objektklassen bestimmte Zusammenhänge auf physikalisch gegebener spektraler Basis vorhanden sind, können diese wie in Kapitel 6 bereits beschrieben in den Lernprozess integriert werden. Für die Kategorie Vegetation zeichnet sich zuverlässig ein bestimmtes Muster des spektralen Verlaufs entsprechend der PAR-Kurve ab. Dabei sind die Relationen zwischen dem NIR-Kanal und den einzelnen Farbkanälen charakteristisch und werden auch in Form von Vegetationsindices verwendet. Das Verhältnis zwischen dem NIR-Kanal und dem Rot-Kanal, welcher bei Vegetation durch die hohe Reflexion im nahen Infraroten und die hohe Absorption im roten Spektrum einen sehr hohen Wert besitzt, wird durch den NDVI beschrieben [23]. Der Green Normalized Difference Vegetation Index (GNDVI) charakterisiert sich durch die Relation zwischen NIR-Kanal und Grün-Kanal, an dessen Stelle sich üblicherweise die Grünlücke befindet [70]. Dieser Vegetationsindex ist außerdem stark verbunden mit dem Normalized Difference Water Index (NDWI) [71]. Zusätzlich befindet sich noch ein spezifisches lokales Maximum im Bereich des blauen Spektrums aus welchem

## Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

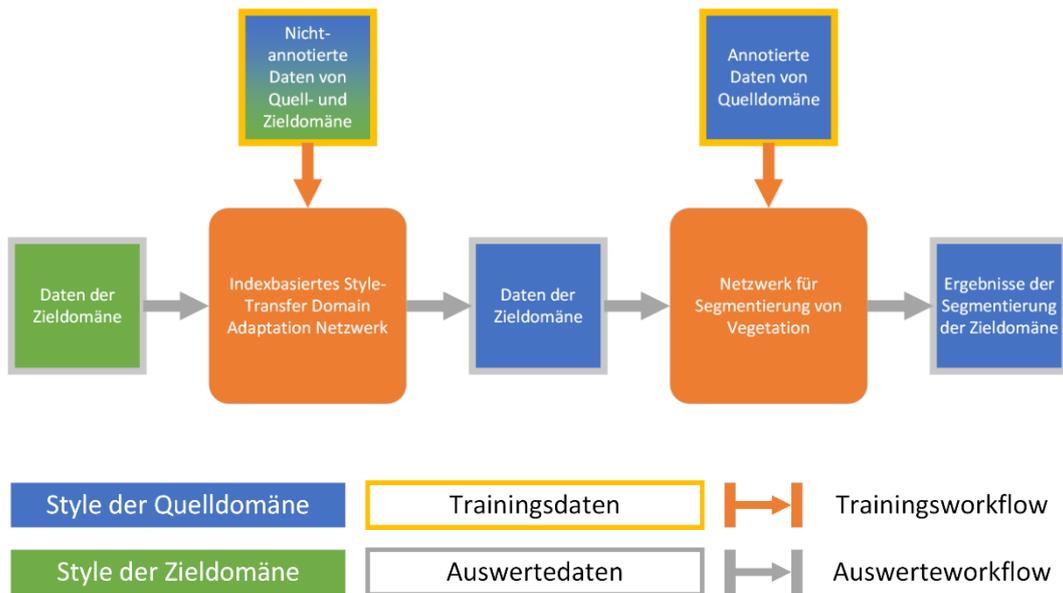


Abbildung 36: Schematische Darstellung des Trainings- und Auswerteworkflows der Methode für die Quell- und Zieldomäne.

basierend auf NIR-Kanal und Blau-Kanal der Blue Normalized Difference Vegetation Index (BNDVI) berechnet werden kann [72].

Die Methode besteht aus zwei Teilen, einem Cycle-Consistent Adversarial Network, welches für den Style-Transfer zwischen Quell- und Zieldomäne verantwortlich ist und einem CNN für die anschließende Segmentierung von Vegetation. Zunächst wird das CNN für die Segmentierung von Vegetation trainiert. Dabei kommt eine U-Net-Struktur wie bereits in Kapitel 6 zum Einsatz. Das Netz wurde basierend auf annotierten Daten der Quelldomäne für die zusammengefasste Objektklasse Vegetation (Bäume und niedrige Vegetation) und Hintergrund (alle weiteren Objektklassen) trainiert. Im Anschluss wird das Cycle-Consistent Adversarial Network basierend auf nicht-annotierten Daten der Quell- und Zieldomäne trainiert. Für die Auswertung werden Daten der Zieldomäne zunächst in den Style der Quelldomäne übersetzt und anschließend mit dem CNN zur Segmentierung von Vegetation ausgewertet. Eine Übersicht der Methode mit den einzelnen Schritten ist in Abbildung 36 dargestellt.

Das Training des Cycle-Consistent Adversarial Networks basiert grundlegend auf der Zusammenstellung von Fehlerfunktionen für den Style-Transfer-Prozess, welche in Kapitel 5 zum Einsatz kamen. Die Fehlerfunktion setzt sich aus den beiden Termen für den Adversarial-Anteil mit  $L_{GAN}(G, D_T, S, T)$  und  $L_{GAN}(F, D_S, T, S)$  sowie dem Cycle-Consistent-

Fehler  $L_{cyc}(G, F)$  und dem Fehler bei Anwenden eines Generators auf Bilder, welche bereits der jeweiligen Domäne entsprechen  $L_{Identität}(G, F)$  zusammen. Dies ergibt sich zu der Form:

$$L(G, F, D_S, D_T) = L_{GAN}(G, D_T, S, T) + L_{GAN}(F, D_S, T, S) + L_{cyc}(G, F) + L_{Identität}(G, F)$$

Dabei wird versucht, für das Training der Generatoren und Diskriminatoren den folgenden Ausdruck zu optimieren:

$$G^*, F^* = \min_G \max_D L_{GAN}(G, F, D_S, D_T)$$

Bei dieser Methode wird nun diese Gesamtfehlerfunktion um weitere vegetationspezifische Terme erweitert. Ziel dieser Anpassung ist es, dass die charakteristische spektrale Form von Vegetation, welche in Kapitel 6 beschrieben ist, auch nach Anwenden der Style-Transfer-Methode in den Bildern vorhanden ist. Dazu werden Terme entsprechend dem NDVI, dem GNDVI und dem BNDVI mit

$$Index(VIS) = \frac{NIR - VIS}{NIR + VIS}$$

hinzugefügt, wobei  $VIS \in R, G, B$  den einzelnen RGB-Kanälen entspricht. Außerdem wurde dem Nenner ein kleiner Wert hinzugefügt, um die numerische Stabilität zu gewährleisten. Dementsprechend wird während des Trainings der Wert für  $Index(VIS)$  des Originalbildes kontinuierlich mit der jeweiligen übersetzten Variante verglichen. Dabei gilt für die Transformation von der Quell- in die Zieldomäne:

$$L_{Index(VIS)} = |Index(VIS)(s) - Index(VIS)(G(s))|^2$$

Für die Transformation der Bilder von der Ziel- in die Quelldomäne gilt dementsprechend die Bedingung:

$$L_{Index(VIS)} = |Index(VIS)(t) - Index(VIS)(F(t))|^2$$

Die resultierende gesamte Fehlerfunktion setzt sich nun aus den ursprünglichen Termen für das Cycle-Consistent Adversarial Network und den hier vorgestellten vegetationspezifischen Anteilen zusammen:

$$L(G, F, D_S, D_T) = L_{GAN}(G, D_T, S, T) + L_{GAN}(F, D_S, T, S) + L_{cyc}(G, F) + L_{Identität}(G, F) \\ + L_{Index(R)}(G, F) + L_{Index(G)}(G, F) + L_{Index(B)}(G, F)$$

Für das Training wurde der Adam-Optimierer [61] verwendet und es kam eine Batch-Größe von 16 zum Einsatz.

## 7.2 Ergebnisse und Diskussion

Die Methode wurde im Rahmen von zwei Szenarien untersucht und ausgewertet. In Szenario 1 wurde ein Style-Transfer von Luftbilddatensatz Sommer auf Satellitenbilddatensatz Sommer durchgeführt. Da die Bilder beider Datensätze zwar in unterschiedlichen Jahren aber zur selben Jahreszeit aufgezeichnet wurden, waren die Vegetationscharakteristika ähnlich. In Szenario 2 kam als Zieldomäne der Luftbilddatensatz Herbst zum Einsatz.

### 7.2.1 Style-Transfer zwischen den Datensätzen

Zunächst wurde der Style-Transfer zwischen den Datensätzen qualitativ untersucht. In Abbildung 37 sind für Szenario 1 sowohl der Transfer von der Quelldomäne zur Zieldomäne wie auch umgekehrt dargestellt. Man kann auf visueller Basis feststellen, dass der Transfer bei diesem Szenario funktioniert hat und entsprechende Charakteristika übernommen wurden. Für den Fall des Szenario 2 lässt sich nach dem Training des Cycle-Consistent Adversarial Networks ebenfalls qualitativ feststellen, dass die Generatoren für eine entsprechende Bild-zu-Bild-Übersetzung verwendet werden können. Eine beispielhafte Darstellung ist in Abbildung 38 gegeben.

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

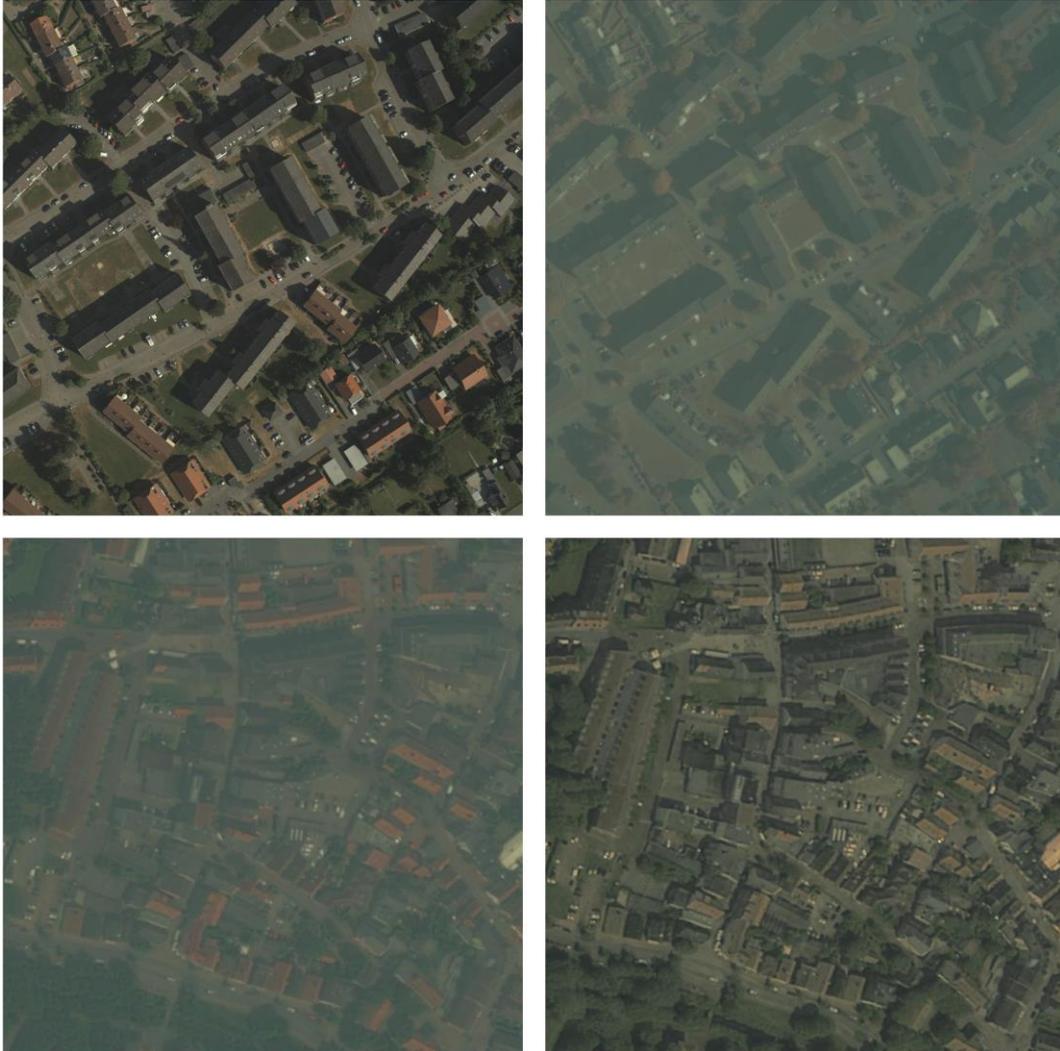


Abbildung 37: Oben links: RGB-Bild von Luftbilddatensatz Sommer. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer nach dem Style-Transfer. Unten links: RGB-Bild von Satellitenbilddatensatz Sommer. Unten rechts: RGB-Bild von Satellitenbilddatensatz Sommer nach dem Style-Transfer.

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks



Abbildung 38: Oben links: RGB-Bild von Luftbilddatensatz Sommer. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer nach dem Style-Transfer. Unten links: RGB-Bild von Luftbilddatensatz Herbst. Unten rechts: RGB-Bild von Luftbilddatensatz Herbst nach dem Style-Transfer.

## 7.2.2 Transfer spektraler Eigenschaften

Im Folgenden soll nun untersucht werden inwiefern die in beiden Szenarien generierten Bilder auch tatsächlich ihre Charakteristika bezüglich Vegetation behalten haben. Dazu wird in einem ersten Schritt überprüft, ob die entsprechenden Vegetationsindexwerte erhalten blieben. Es wurde ausgewertet, inwiefern sich der Einsatz der zusätzlichen Terme der Fehlerfunktion auf die Erhaltung der Vegetationsindexwerte ausgewirkt hat. Dazu wurden zunächst für die Evaluationsdaten der Zieldomänen Indexbilder für jeden der drei Vegetationsindices (NDVI, GNDVI, BNDVI) errechnet und diese mit den Vegetationsindexbilder der entsprechenden Bild-zu-Bild-Übersetzung von der Zieldomäne zur Quelldomäne verglichen. Dazu wurde jeweils die mittlere quadratische Abweichung (MQA) errechnet.

Für das Szenario 1 sind die Ergebnisse in Tabelle 14 zusammengefasst und in Abbildung 39 für den NDVI, in Abbildung 40 für den GNDVI und in Abbildung 41 für den BNDVI dargestellt. Dabei lässt sich feststellen, dass sich für die Bild-zu-Bild-Übersetzung durch den Einsatz von  $L_{Index}$  für die angepassten Bilder der Zieldomäne ergibt, dass für die Ergebnisse der Berechnung des NDVI der Wert von 5,10% auf 3,74%, für den GNDVI von 4,90% auf 4,24% und für den BNDVI von 5,10% auf 3,74% verbessert werden konnten.

*Tabelle 14: Abweichung der Vegetationsindices vom Ursprungsbild als mittlere quadratische Abweichung (MQA) angegeben in Prozent mit dem Satellitenbilddatensatz Sommer als Zieldomäne.*

| <b>MQA [%]</b> | <b>ohne <math>L_{Index}</math></b> | <b>mit <math>L_{Index}</math></b> |
|----------------|------------------------------------|-----------------------------------|
| NDVI           | 5,10                               | 3,74                              |
| GNDVI          | 4,90                               | 4,24                              |
| BNDVI          | 5,10                               | 3,74                              |

Für das Szenario 2 sind die Ergebnisse in Tabelle 15 zusammengefasst und in Abbildung 42 für den NDVI, in Abbildung 43 für den GNDVI und in Abbildung 44 für den BNDVI dargestellt. In diesem Szenario kann man sehen, dass sich für die Bild-zu-Bild-Übersetzung durch den Einsatz von  $L_{Index}$  für die angepassten Bilder der Zieldomäne im Fall des Luftbilddatensatz Herbst ergibt, dass für die Berechnung des NDVI der Wert von 12,96% auf

## Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

5,74%, für den GNDVI von 13,82% auf 5,66% und für den BNDVI von 17,35% auf 6,32% verbessert werden konnten.

*Tabelle 15: Abweichung der Vegetationsindices vom Ursprungsbild als mittlere quadratische Abweichung (MQA) angegeben in Prozent mit dem Luftbilddatensatz Herbst als Zieldomäne.*

| <b>MQA [%]</b> | <b>ohne <math>L_{Index}</math></b> | <b>mit <math>L_{Index}</math></b> |
|----------------|------------------------------------|-----------------------------------|
| NDVI           | 12,96                              | 5,74                              |
| GNDVI          | 13,82                              | 5,66                              |
| BNDVI          | 17,35                              | 6,32                              |

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

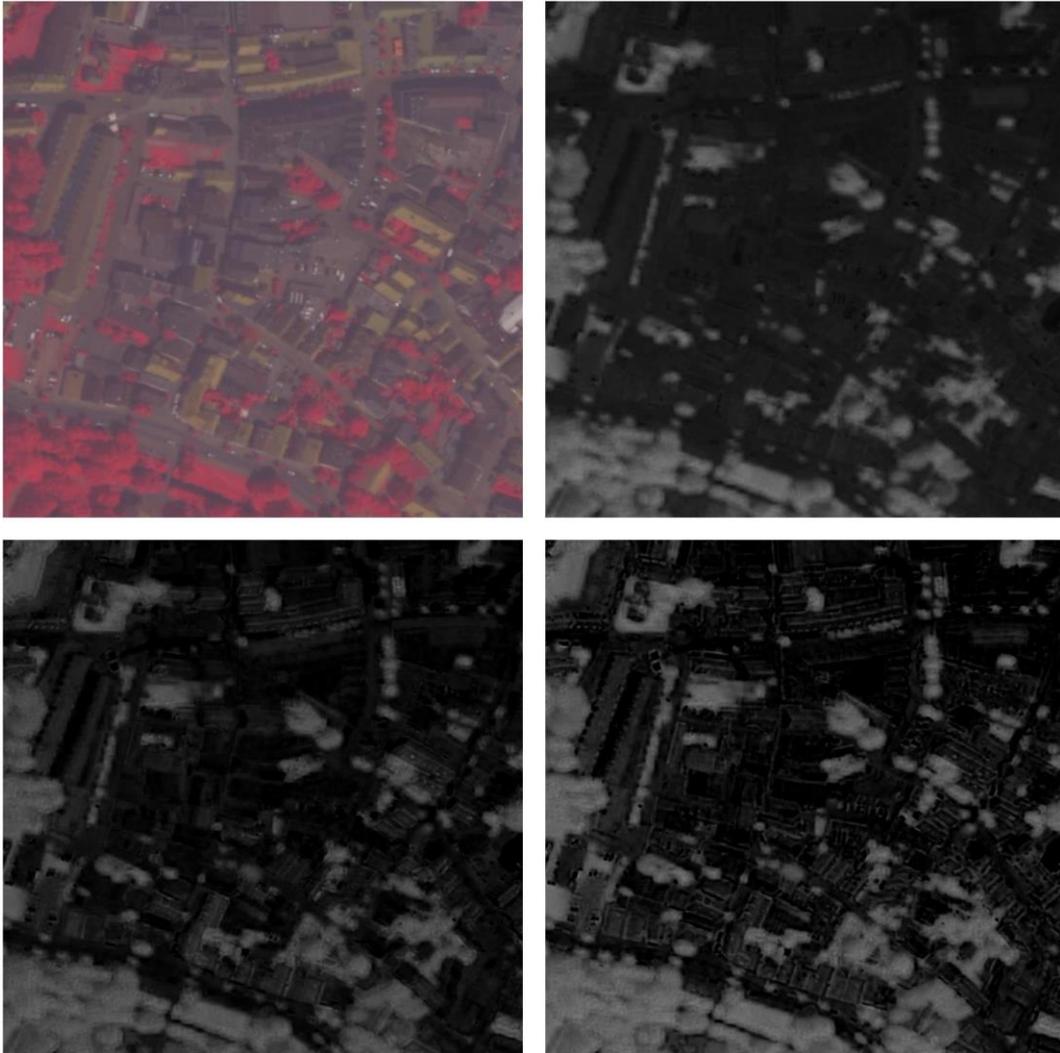


Abbildung 39: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: NDVI der ursprünglichen Bildes. Unten links: NDVI des angepassten Bildes ohne  $L_{index}$ . Unten rechts: NDVI des angepassten Bildes mit  $L_{index}$ .

## Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

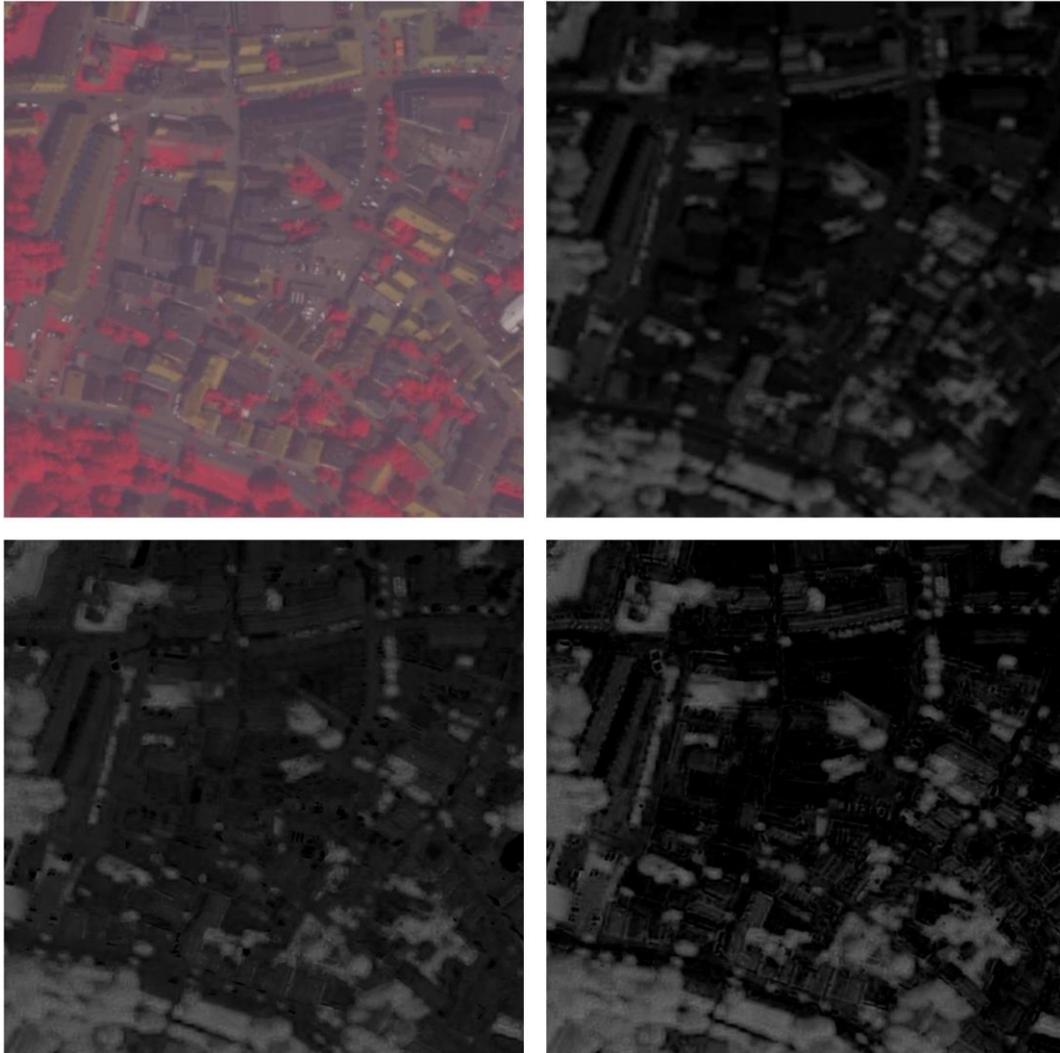


Abbildung 40: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: GNDVI der ursprünglichen Bildes. Unten links: GNDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: GNDVI des angepassten Bildes mit  $L_{Index}$ .

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

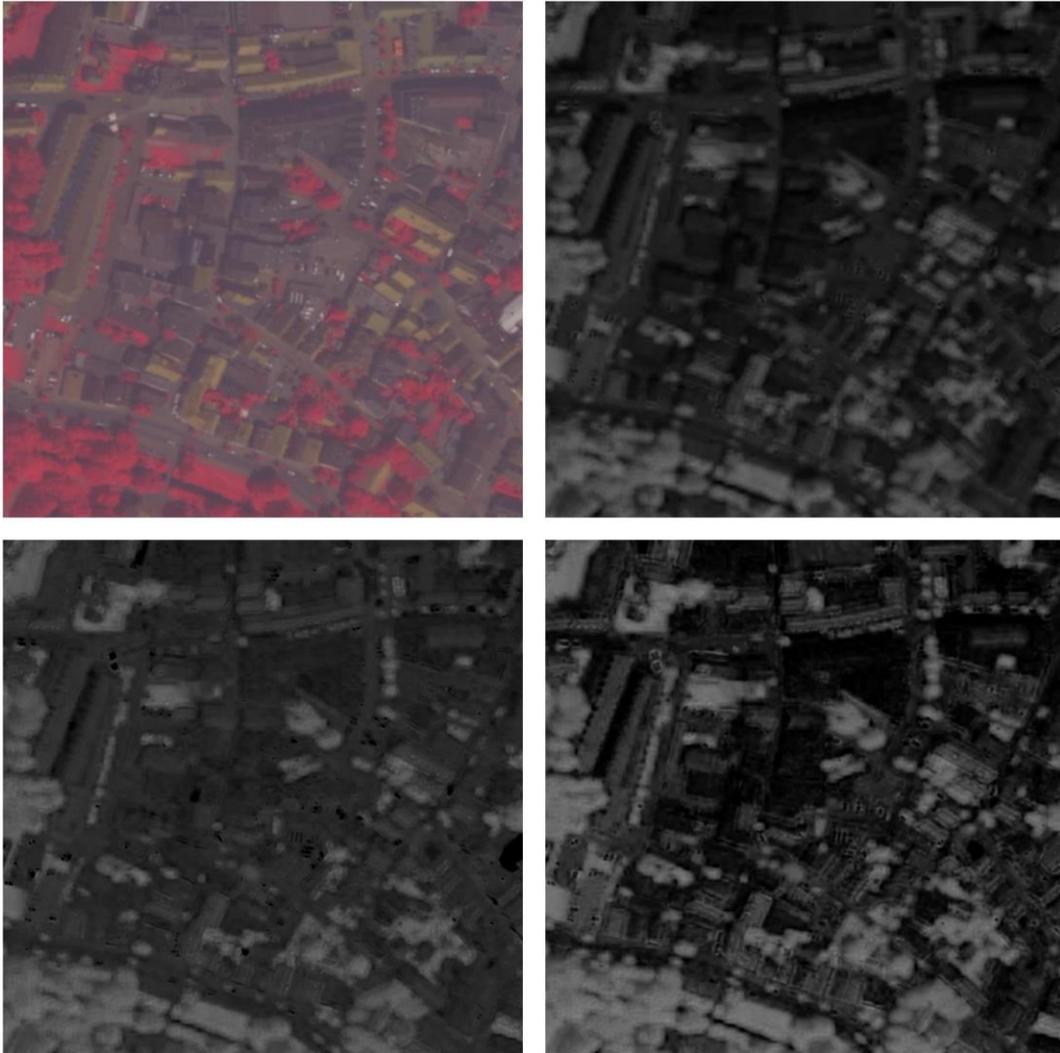


Abbildung 41: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: BNDVI der ursprünglichen Bildes. Unten links: BNDVI des angepassten Bildes ohne  $L_{index}$ . Unten rechts: BNDVI des angepassten Bildes mit  $L_{index}$ .

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

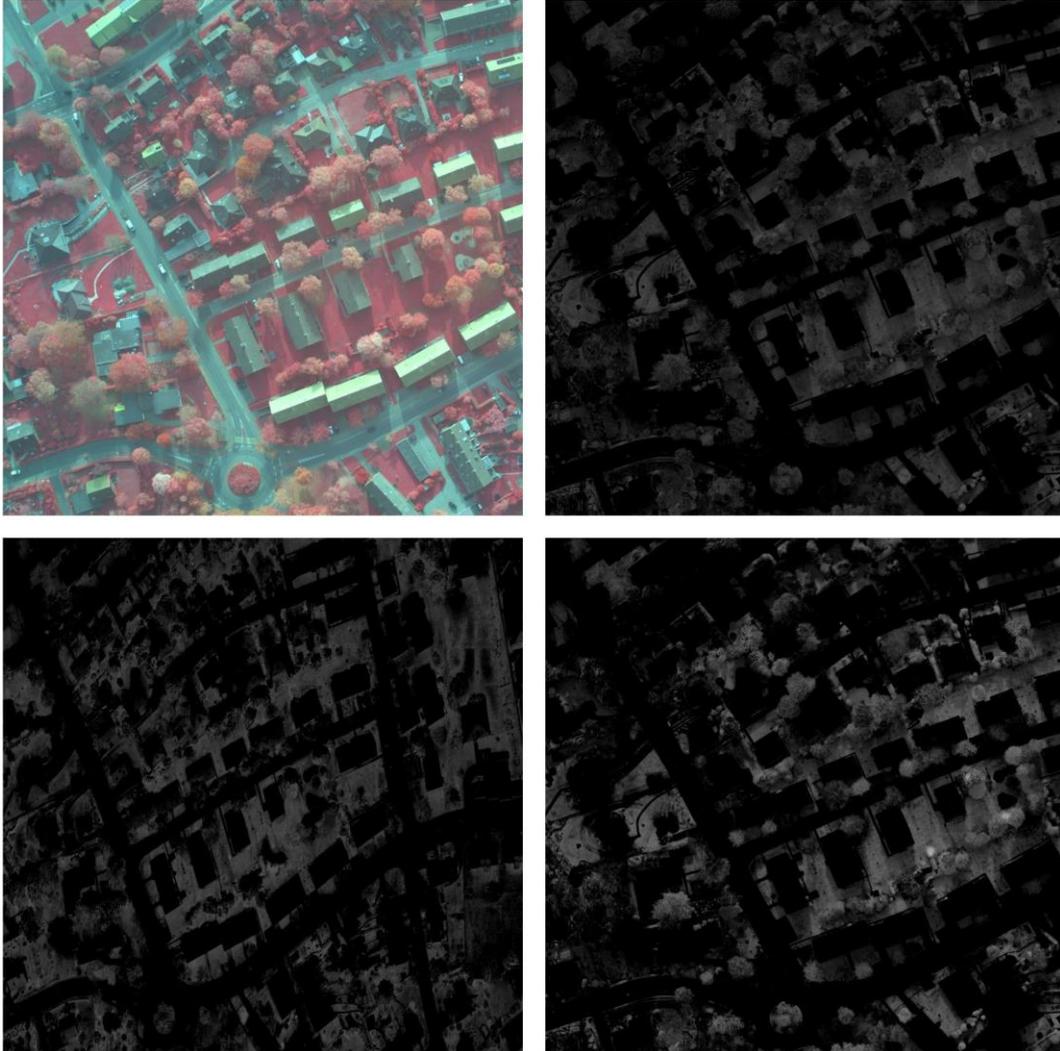


Abbildung 42: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: NDVI der ursprünglichen Bildes. Unten links: NDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: NDVI des angepassten Bildes mit  $L_{Index}$ .

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

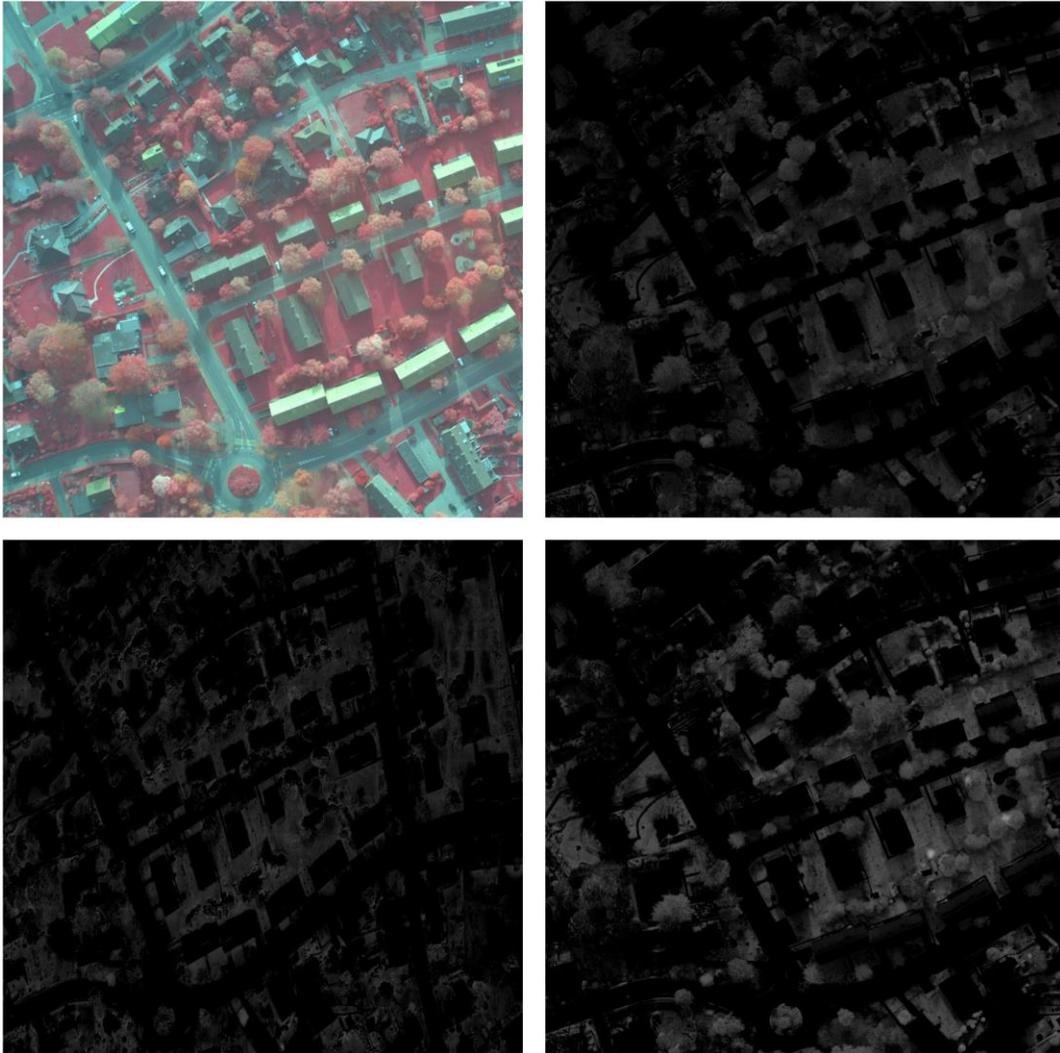


Abbildung 43: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: GNDVI der ursprünglichen Bildes. Unten links: GNDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: GNDVI des angepassten Bildes mit  $L_{Index}$ .

Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

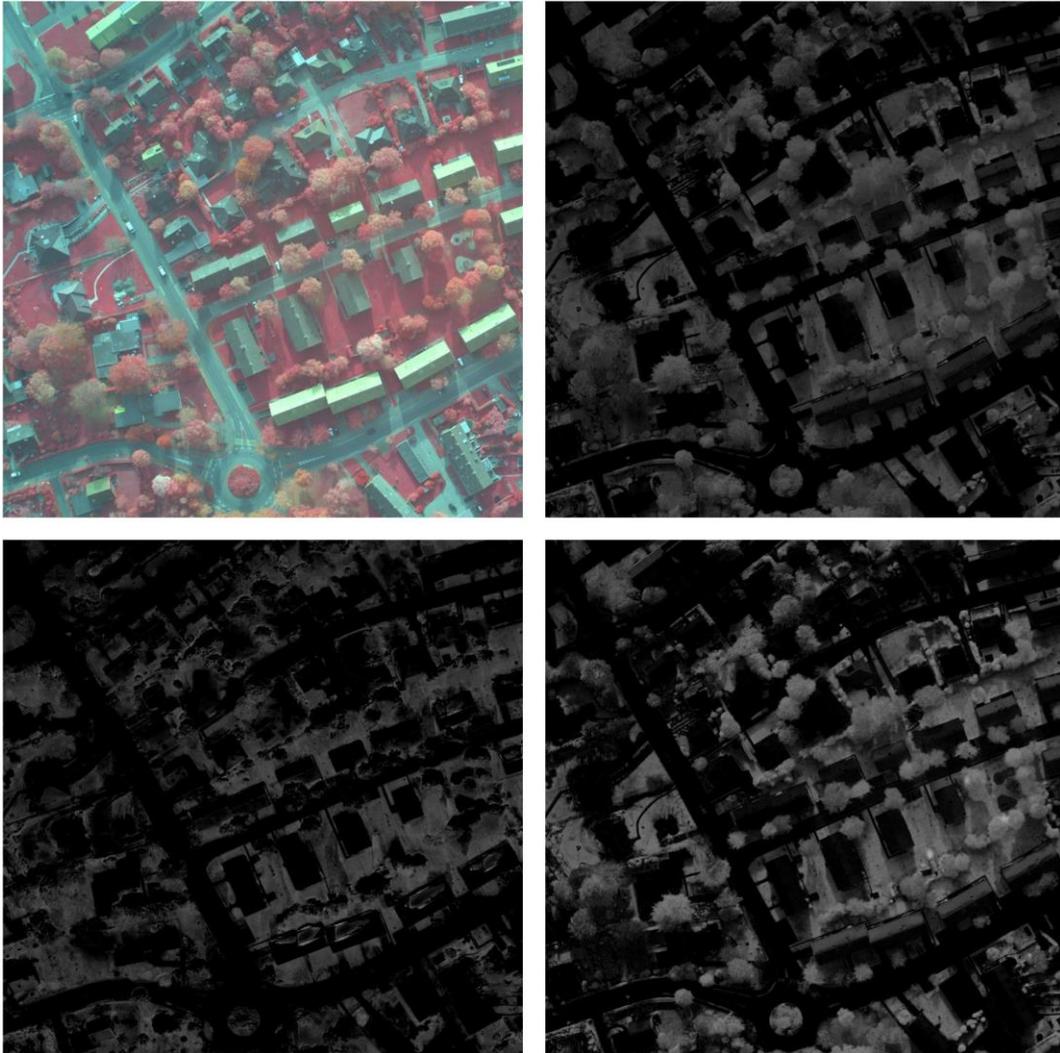


Abbildung 44: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: BNDVI der ursprünglichen Bildes. Unten links: BNDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: BNDVI des angepassten Bildes mit  $L_{Index}$ .

### 7.2.3 Segmentierung von Vegetation

Im nächsten Schritt wurde untersucht, ob sich durch den Einsatz der zusätzlichen Terme innerhalb der Fehlerfunktion während des Trainings des Cycle-Consistent Adversarial Networks eine Verbesserung der Segmentierungsergebnisse von Vegetation beobachten lässt. Dazu wurde der  $F_1$ -Wert und die IoU für den Fall ermittelt, dass keine Anpassung der Bilder stattgefunden hat, dass ein Style-Transfer durchgeführt wurde ohne ein Training basierend auf  $L_{Index}$  und entsprechend ein Training mit  $L_{Index}$  stattgefunden hat. Für das Szenario 1 sind die Ergebnisse in Tabelle 16 zusammengefasst und in Abbildung 45 dargestellt. Es lässt sich beobachten, dass für den Fall, dass keine Anpassung stattfindet, die Ergebnisse der Segmentierung sehr niedrig ausfallen. Durch einen Style-Transfer ohne  $L_{Index}$  kann im Vergleich zum Fall ohne Anpassung der  $F_1$ -Wert von 0,42 auf 0,85 und die IoU von 0,09 auf 0,74 verbessert werden. Durch den Einsatz von  $L_{Index}$  während des Trainings des Cycle-Consistent Adversarial Networks können diese Werte mit einem  $F_1$ -Wert von 0,87 und einer IoU von 0,77 noch weiter verbessert werden. Für das Szenario 2 sind die Ergebnisse in Tabelle 17 zusammengefasst und in Abbildung 46 dargestellt. In dem Fall ohne Anpassung ergibt sich ein  $F_1$ -Wert von 0,90 und eine IoU von 0,80. Für den Fall des Trainings des Cycle-Consistent Adversarial Networks ohne  $L_{Index}$  und eine anschließende Domain Adaptation ergeben sich dieselben Werte. Durch den Einsatz von  $L_{Index}$  während des Trainings können diese Werte mit einem  $F_1$ -Wert von 0,92 und einer IoU von 0,84 zusätzlich verbessert werden.

Tabelle 16: Ergebnisse der Segmentierung für den Satellitenbilddatensatz Sommer angegeben als  $F_1$ -Wert und IoU für die Fälle: Ohne Anpassung, Style-Transfer ohne  $L_{Index}$  und Style-Transfer mit  $L_{Index}$ .

| Metrik      | Ohne Anpassung | Ohne $L_{Index}$ | Mit $L_{Index}$ |
|-------------|----------------|------------------|-----------------|
| $F_1$ -Wert | 0,42           | 0,85             | 0,87            |
| IoU         | 0,09           | 0,74             | 0,77            |

Tabelle 17: Ergebnisse der Segmentierung für den Luftbilddatensatz Herbst angegeben als  $F_1$ -Wert und IoU für die Fälle: Ohne Anpassung, Style-Transfer ohne  $L_{Index}$  und Style-Transfer mit  $L_{Index}$ .

| Metrik      | Ohne Anpassung | Ohne $L_{Index}$ | Mit $L_{Index}$ |
|-------------|----------------|------------------|-----------------|
| $F_1$ -Wert | 0,90           | 0,90             | 0,92            |
| IoU         | 0,80           | 0,80             | 0,84            |

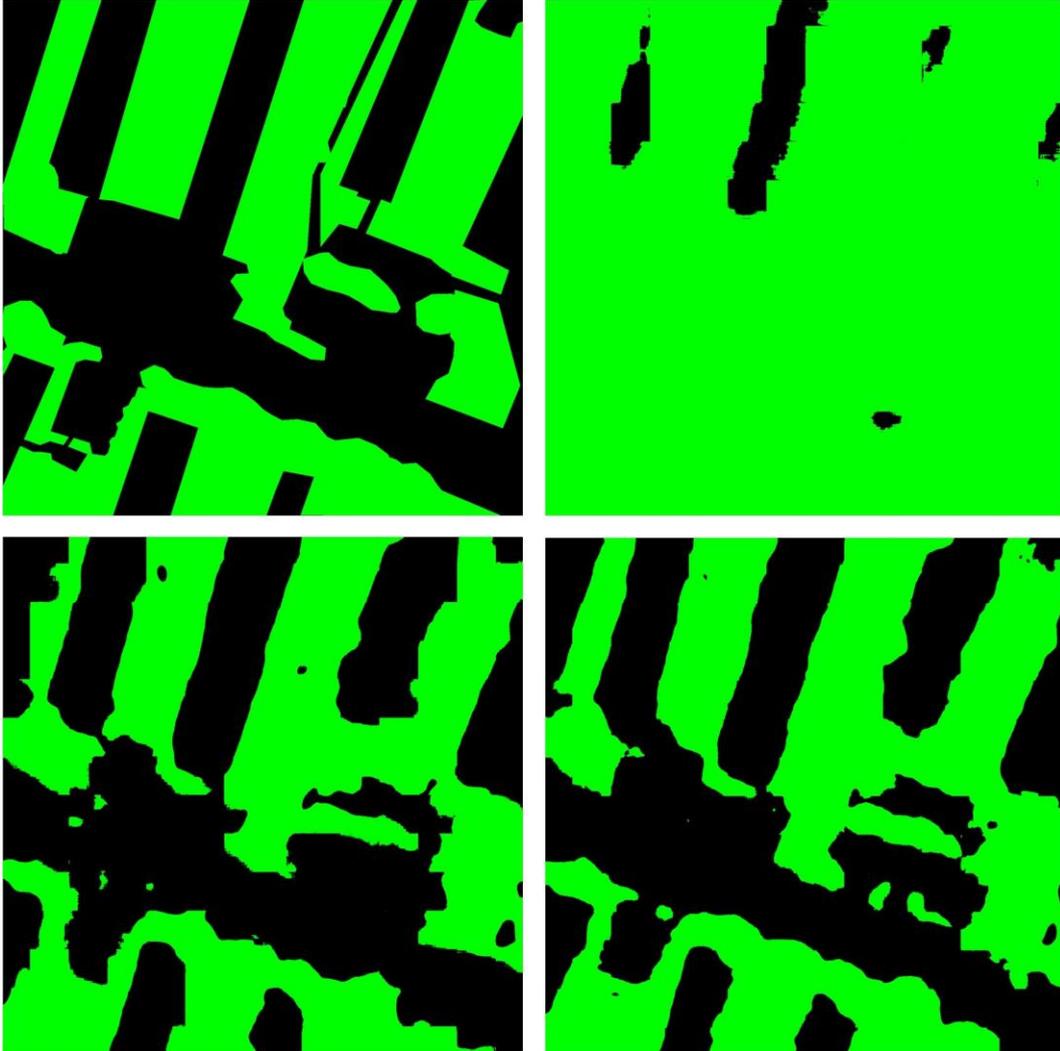


Abbildung 45: Oben links: Ground Truth des Satellitenbilddatensatz Sommer. Oben rechts: Ergebnisse der Segmentierung ohne Anpassung. Unten links: Ergebnisse der Segmentierung ohne  $L_{index}$ . Unten rechts: Ergebnisse der Segmentierung mit  $L_{index}$ .

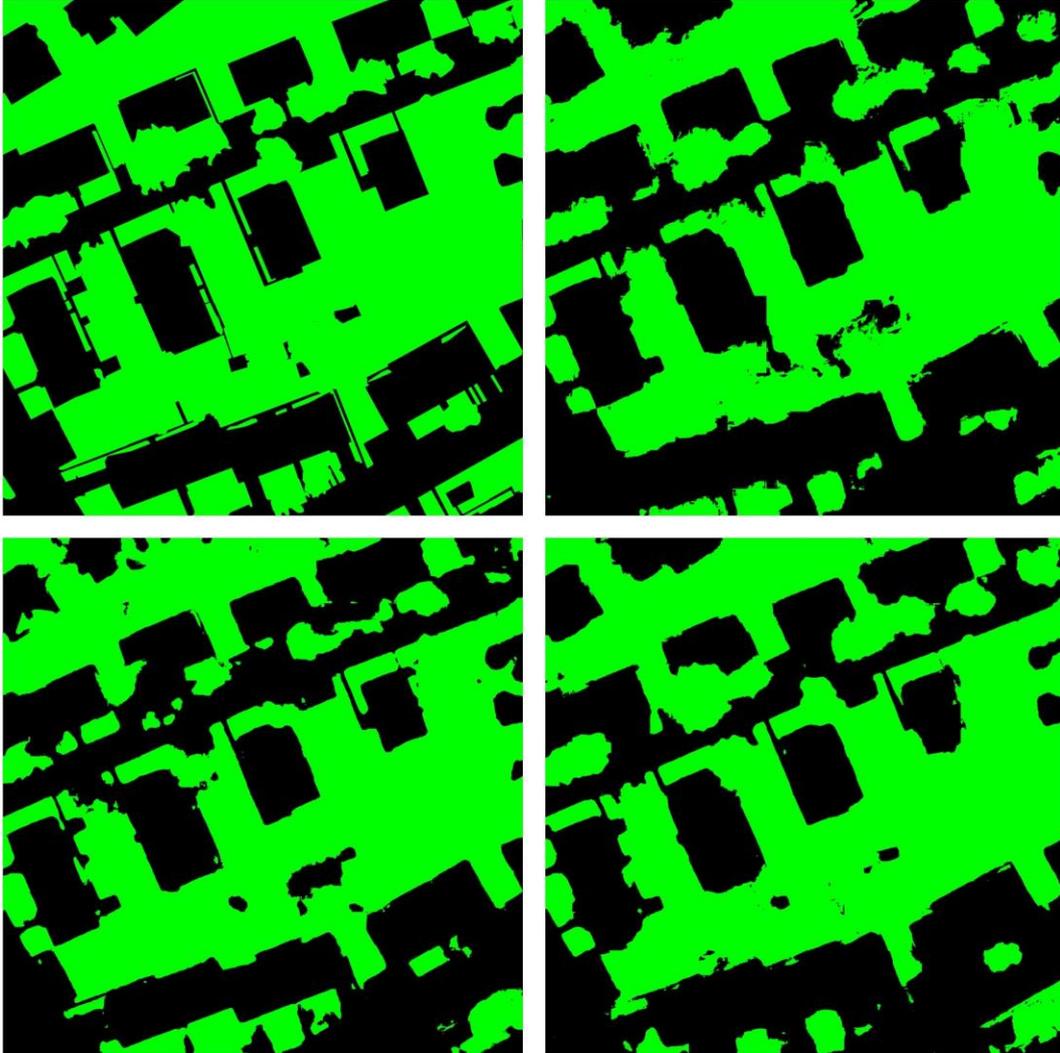


Abbildung 46: Oben links: Ground Truth des Luftbilddatensatz Herbst. Oben rechts: Ergebnisse der Segmentierung ohne Anpassung. Unten links: Ergebnisse der Segmentierung ohne  $L_{Index}$ . Unten rechts: Ergebnisse der Segmentierung mit  $L_{Index}$ .

Für beide Szenarien hat sich gezeigt, dass durch den Einsatz der zusätzlichen Terme der Fehlerfunktion die jeweiligen spektralen Charakteristika der Vegetationseigenschaften besser übertragen werden konnten als ohne. Dies hat sich deutlich bei der Berechnung der unterschiedliche Vegetationsindices gezeigt. Vor allem in Szenario 2 konnte man beobachten, dass ohne  $L_{Index}$  die Abweichung der Indexwerte verglichen zum Ursprungsbild deutlich zunimmt, während in dem Fall mit  $L_{Index}$  der Unterschied deutlich kleiner ausfällt. Somit konnte gezeigt werden, dass durch diese Methode die spektralen Eigenschaften von Vegetation gezielt während des Style-Transfer-Prozesses beibehalten werden können. In Bezug auf die Segmentierung von Vegetation hat sich für Szenario 1 gezeigt, dass bei dem Einsatz einer Zieldomäne basierend auf Bildern eines anderen Sensors wie in diesem Fall, die Segmentierung ohne einen Domain Adaptation-Prozess sehr schlechte Klassifikationsergebnisse liefert. Durch den Einsatz eines Cycle-Consistent Adversarial Networks kann dieses Ergebnis deutlich verbessert werden. Ein Training der Style-Transfer-Methode mit den zusätzlichen Termen innerhalb der Fehlerfunktion kann das Ergebnis nochmals um 2,35%, gemessen mit dem  $F_1$ -Wert, verbessern. Somit konnte gezeigt werden, dass das Segmentierungsnetzwerk neben räumlichen Merkmalen ebenfalls spezifische spektrale Informationen gelernt hat und für die Klassifikation von Vegetation verwendet. Bei der Auswertung der Ergebnisse für das Szenario 2 haben sich bereits gute Segmentierungsergebnisse bei der direkten Anwendung des Segmentierungsnetzwerks ergeben, welche durch den einfachen Style-Transfer basierend auf einem Cycle-Consistent Adversarial Network nicht verbessert werden konnten. Die bereits guten Ergebnisse auch ohne Anpassung lassen sich durch den Einsatz des NIR-Kanals zusätzlich zu RGB erklären, da das Netzwerk durch die spezifischen spektralen Eigenschaften von Vegetation bereits gut Zusammenhänge herstellen kann um diese Objektklasse von anderen Klassen zu unterscheiden. Durch den Einsatz der zusätzlichen Terme der Fehlerfunktion hat sich der  $F_1$ -Wert allerdings noch um 2,22% steigern können. Aus diesem Umstand lässt sich schlussfolgern, dass durch den Einsatz eines Cycle-Consistent Adversarial Networks zwar die räumlichen Charakteristika besser angepasst wurden, allerdings spektrale Informationen, welche für die Segmentierung von Vegetation relevant sind, im Falle des Trainings ohne  $L_{Index}$ , teilweise verloren gegangen sind. Dementsprechend konnte sich für diesen Fall keine Verbesserung der Ergebnisse trotz des Style-Transfers beobachten lassen. Bei

## Domain Adaptation für die Segmentierung von Vegetation mit Cycle-Consistent Adversarial Networks

dem Einsatz der zusätzlichen vegetationspezifischen Terme konnten durch ein Beibehalten der spektralen Zusammenhänge dementsprechend die Ergebnisse verbessert werden.

## 8 Zusammenfassung und Ausblick

In dieser Arbeit wurden Methoden zur Verbesserung von Domain Adaptation für die semantische Segmentierung mit Anwendung auf die Klassifikation von Bodenbedeckung bei Luft- und Satellitenbilddatensätzen in urbanen Gebieten vorgestellt. Dabei wurde ein Schwerpunkt der Untersuchungen auf die Klasse Vegetation gelegt. Dazu sind drei Datensätze erstellt worden, welche unterschiedliche Charakteristika bezüglich Helligkeit, Farbgebung, Schattenwurf, Bodenauflösung und Jahreszeit aufweisen. Das Ziel der Analyse und Entwicklung von geeigneten Algorithmen war es, die Klassifizierungsergebnisse zu verbessern und den manuellen Annotationsaufwand zu senken. Untersucht wurden sowohl Methoden für die überwachte Domain Adaptation (mit annotierten Daten in der Zieldomäne) wie auch die unüberwachte Domain Adaptation (ohne annotierte Daten in der Zieldomäne). Im Rahmen der überwachten Domain Adaptation wurde zunächst in Kapitel 4 ein Encoder-Decoder-basiertes Convolutional Neural Network, welches mit Skip-Connections in jeder Ebene verwunden war, auf dem umfangreichen ISPRS Potsdam-Datensatz für die semantische Segmentierung trainiert. Im Anschluss sollte dieses neuronale Netz an den Luftbilddatensatz Sommer als Zieldomäne angepasst werden. Da die Encoder-Decoder-Struktur bereits gute Ergebnisse auf dem Quelldomäne-Datensatz lieferte, war das Ziel, die große Menge an Gewichten in der vortrainierten Struktur nicht mehr signifikant durch Nachtraining zu verändern. Deshalb wurde für die Domain Adaptation eine für diesen Zweck verantwortliche Zwischenstruktur eingebaut, welche als Adaptionmodule bezeichnet werden. Diese Module befinden sich zwischen Encoder und Decoder anstelle der Skip-Connections und sollen die Ausgabe des Encoders dementsprechend anpassen, dass der Decoder zuverlässige Auswertergebnisse liefert. Das Training wurde anschließend in einem alternierenden Schema durchgeführt. Die Methode mit den Adaptionmodulen wurde mit der Fine-Tuning-Methode als Referenz verglichen und auf unterschiedliche Teildatensätzen für die Evaluierung angewandt. Es hat sich herausgestellt, dass sich vor allem bei dem Einsatz sehr kleiner annotierter Datensätze der Zieldomäne eine deutliche Verbesserung herausstellt. Damit konnte gezeigt werden, dass eine strukturelle Erweiterung der bereits trainierten Netzstruktur zu einem besseren Ergebnis führen kann als ein Nachtraining der Grundstruktur. Im nächsten Schritt wurde in Kapitel

5 vollständig auf annotierte Daten in der Zieldomäne für den Domain Adaptation-Schritt verzichtet und eine Methode basierend auf Cycle-Consistent Adversarial Networks untersucht. Dabei handelt es sich um ein neuronales Netz, welches für einen Style-Transfer zwischen zwei Datensätzen eingesetzt werden kann. Dazu werden im Rahmen dieser Methode zwei sogenannte Generatoren optimiert, welche einen entsprechenden Bild-zu-Bild-Transfer erlernen. Somit kann ein Bild des ersten Datensatzes auf Grundlage des einen Generators dementsprechend angepasst werden, dass es die Charakteristika des zweiten Datensatzes annimmt und mit dem anderen Generator entsprechend umgekehrt. Als Quelldomäne für das Training des neuronalen Netzes zur semantischen Segmentierung wurde nun der Luftbilddatensatz Sommer eingesetzt. Die Zieldomäne war bei dieser Untersuchung der Luftbilddatensatz Herbst. Beide Datensätze stammen zwar von demselben Sensor, weisen aber große Unterschiede in Helligkeit, Farbgebung, Schattenwurf und Jahreszeiten auf. Dementsprechend lieferte das neuronale Netz zur semantischen Segmentierung keine zuverlässigen Ergebnisse. Für das Training des Cycle-Consistent Adversarial Networks wurden von der Quelldomäne und Zieldomäne nur Bilder ohne ihre entsprechende Annotation verwendet. Beim Training wurde ein spezifisches Schema angewandt, welches ein kontinuierliches Anpassen des neuronalen Netzes zur semantischen Segmentierung vorsieht. Somit wurde während des Trainings des Cycle-Consistent Adversarial Networks immer der aktuelle Generator verwendet, welcher von der Quelldomäne in die Zieldomäne übersetzt, um auf die Zieldomäne angepasste Daten basierend auf der Quelldomäne zu erzeugen. Diese Daten wurden schließlich zusammen mit den bereits vorhandenen Annotationen der Quelldomäne dazu verwendet, um das Netzwerk zur semantischen Segmentierung zu trainieren. Durch das Verwenden unterschiedlicher Varianten dieses Generators wurde eine hohe Generalisierung der Auswertemethode erreicht. Dies ist insofern von Vorteil, da es in der Regel nicht möglich ist bei Cycle-Consistent Adversarial Networks eine messbare genau passende Transfervariante zu bestimmen. Im Vergleich zu den beiden Referenzmethoden, welche kein kontinuierliches Training vorsahen konnten die Klassifikationsergebnisse deutlich verbessert werden. In den beiden anschließenden Kapiteln wurde in dem Bereich der unüberwachten Domain Adaptation verstärkt auf die Segmentierung von Vegetation eingegangen. Dazu wurde zunächst in Kapitel 6 der Luftbilddatensatz Sommer als Quelldomäne untersucht und basierend auf spezifischen spektralen Zusammenhängen von Vegetation auf NIR-RGB-Bildern

teilautomatisiert annotiert. Dafür wurde als Grundlage die Segmentierung des Bildes mit einem Schwellwertverfahren bei dem Einsatz von Vegetationsindices verwendet. Diese Indices liefern eine Annotation des Bildes ohne weitere Parametereinstellungen. Allerdings werden dabei einige Pixel, welche auf Basis spektraler Zusammenhänge nicht richtig zugeordnet werden können, falsch annotiert. Deshalb wurde dafür zusätzlich ein Convolutional Neural Network daraufhin trainiert, räumliche Zusammenhänge auf Grundlage der spektralen Vorannotation zu erkennen und die Ergebnisse basierend auf diesem einfachen Schwellwertverfahren zu verbessern. Dies wurde auf Basis des Normalized Difference Vegetation Index (NDVI), des Soil-Adjusted Vegetation Index (SAVI) und des Enhanced Vegetation Index (EVI) durchgeführt und ausgewertet. Es hat sich gezeigt, dass der SAVI für den Luftbilddatensatz Sommer die besten teilautomatisierten Annotationsergebnisse liefert. In Kapitel 7 wurde nun untersucht, inwiefern sich gegebenenfalls durch den Style-Transfer basierend auf Cycle-Consistent Adversarial Networks, wie in Kapitel 5 beschrieben, die spektralen Zusammenhänge zwischen den einzelnen Kanälen ändern. Diese sind aufgrund der biophysikalischen Eigenschaften von Vegetation charakteristisch und können für eine Klassifikation basierend auf Convolutional Neural Networks mitverantwortlich sein. Als Quelldomäne wurde der Luftbilddatensatz Sommer verwendet, welcher bereits in Kapitel 6 hinsichtlich Vegetation untersucht wurde und als Zieldomäne kam zusätzlich zum Luftbilddatensatz Herbst der Satellitenbilddatensatz Sommer zum Einsatz. Dieser unterscheidet sich gegenüber der Quelldomäne hinsichtlich Helligkeit, Farbgebung, Schattenwurf und Bodenauflösung. Dazu wurde die Fehlerfunktion, auf dessen Grundlage das Cycle-Consistent Adversarial Network trainiert wird, um Terme erweitert, welche die spektralen Zusammenhänge zwischen dem NIR-Kanal und den einzelnen Kanälen von RGB jeweils berücksichtigen. Diese Methode wurde verglichen mit dem Training eines Cycle-Consistent Adversarial Networks ohne eine entsprechende Erweiterung der Fehlerfunktion. Es hat sich bei der Anpassung zum Satellitenbilddatensatz Sommer gezeigt, dass Cycle-Consistent Adversarial Networks im Rahmen von Domain Adaptation die Klassifikationsergebnisse allgemein deutlich verbessern können, wobei durch den Einsatz der zusätzlichen Terme der Fehlerfunktion das Ergebnis besser ausfiel. Bei der Untersuchung der Anpassung zum Luftbilddatensatz Herbst als Zieldomäne wiederum hat sich herausgestellt, dass auch ohne Anpassung bereits ein gutes Klassifikationsergebnis vorliegt, welches durch den Einsatz von Style-Transfer nicht verbessert werden konnte. Dies

lässt sich auf die Tatsache zurückführen, dass beide Datensätze auf demselben Sensor basieren, die Bodenauflösung gleich ist und der NIR-Kanal sehr charakteristische Zusammenhänge bezüglich Vegetation ermöglicht. Allerdings konnte wiederum durch das Einbinden der zusätzlichen Terme in die Fehlerfunktion eine Verbesserung erreicht werden. Daraus lässt sich schlussfolgern, dass durch einen Style-Transfer zwar die räumlichen Informationen gut angepasst werden, allerdings spektrale Zusammenhänge, welche für eine richtige Klassifikation entscheidend sind, verloren gehen. Durch den Einsatz der zusätzlichen Terme der Fehlerfunktion konnte diesem Umstand entgegengewirkt werden und das Gesamtergebnis verbessert werden.

Die Beiträge dieser Arbeit im Rahmen der Domain Adaptation für die semantische Segmentierung können wie folgt zusammengefasst werden:

- Strukturelle Erweiterung und Analyse trainierter Netzstrukturen für die semantische Segmentierung mit Adaptionmodulen zur Verbesserung der überwachten Domain Adaptation und zur Senkung der Menge notwendiger Trainingsdaten
- Integration eines kontinuierlichen Trainingschemas zur Erhöhung der Generalisierung und entsprechender Verbesserung der unüberwachten Domain Adaptation mit Cycle-Consistent Adversarial Networks
- Analyse des Quelldomäne-Datensatzes auf Grundlage von Vegetationsindices hinsichtlich einer teilautomatisierten Annotation auf Grundlage spektraler Informationen
- Verbesserung der entsprechenden Schwellwertsegmentierung des Quelldomäne-Datensatzes basierend auf Vegetationsindices, durch Kombination mit einer passenden Vorverarbeitung und einer CNN-basierten Klassifikation zur Gewinnung räumlicher Zusammenhänge
- Erweiterung der Fehlerfunktion für das Cycle-Consistent Adversarial Network zur unüberwachten Domain Adaptation zur Beibehaltung biophysikalisch-spektraler Charakteristika von Vegetation bei einem Style-Transfer
- Verbesserung der Segmentierung von Vegetation auf Grundlage der erweiterten Fehlerfunktion und Analyse der Auswertergebnisse hinsichtlich der Lernweise von CNN-basierten Segmentierungsnetzwerken

Basierend auf diesen Erkenntnissen können weitere Untersuchungen hinsichtlich der unüberwachten Domain Adaptation für die semantische Segmentierung mit Anwendung auf die Klassifikation von Bodenbedeckung durchgeführt werden. Zum einen könnten entsprechend Kapitel 6 für die teilautomatisierte Annotation von Quelldomäne-Datensätzen weitere Vegetationsindices zum Einsatz kommen. Dies kann je nach Beschaffenheit des zu untersuchenden Bodens unterschiedlich von Vorteil sein. Im Weiteren können auch Indices zum Einsatz kommen, welche für die Klassifikation von Bildpunkten anderer Klassen als Vegetation gedacht sind. Somit kann die Gesamtannotation des Bildes um diese entsprechend erweitert werden. Dazu könnte beispielsweise eine indexbasierte Segmentierung von Wasser auf Grundlage des Normalized Difference Water Index (NDWI)

$$NDWI = \frac{G - NIR}{G + NIR}$$

durchgeführt werden, welcher den grünen Kanal und den nahen Infrarotkanal berücksichtigt [71]. Der mögliche Einsatz von Indices hängt dabei von der spektralen Art der verfügbaren Daten ab. Falls die Daten der Quell- und Zieldomäne entsprechende Kanäle vorweisen, kann dieses Vorgehen auch auf Indices wie den Normalized Difference Build-Up-Index (NDBI) erweitert werden

$$NDBI = \frac{SWIR - NIR}{SWIR + NIR}$$

welcher den kurzwelligen Infrarotkanal (SWIR) in Relation zum NIR-Kanal berücksichtigt [73]. Der Einsatz zusätzlicher Terme in der Fehlerfunktion wie in Kapitel 7 beschrieben kann für einen noch genaueren Style-Transfer auf Bilder mit mehr als vier Kanälen im Bereich der photosynthetisch aktiven Strahlung angewendet werden um die entsprechende charakteristische spektrale Form von Vegetation beizubehalten. Dabei kann wie entsprechend beschrieben jeder einzelne Kanal als Zusatzterm hinzugefügt werden oder eine Auswahl passender Kanäle, bei welchen sich bestimmte weitere Charakteristika herausstellen. Es wäre auch denkbar, dass entsprechende Kombinationen im Vorfeld generiert werden, welche in den Trainingsprozess eingefügt werden können.



## 9 Literaturverzeichnis

- [1] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth und B. Schiele, „The Cityscapes Dataset for Semantic Urban Scene Understanding,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [2] H. Shimodaira, „Improving predictive inference under covariate shift by weighting the log-likelihood function,“ *Journal of Statistical Planning and Inference*, Bd. 90, Nr. 2, pp. 227-244, 2000.
- [3] S. J. Pan und Q. Yang, „A Survey on Transfer Learning,“ *IEEE Transactions on Knowledge and Data Engineering*, Bd. 22, Nr. 10, pp. 1345-1359, 2010.
- [4] M. Wang und W. Deng, „Deep visual domain adaptation: A survey,“ *Neurocomputing*, Bd. 312, pp. 135-153, 2018.
- [5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville und Y. Bengio, „Generative Adversarial Nets,“ in *Advances in Neural Information Processing Systems*, 2014.
- [6] J.-Y. Zhu, T. Park, P. Isola und A. A. Efros, „Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks,“ in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [7] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros und T. Darrell, „CyCADA: Cycle-Consistent Adversarial Domain Adaptation,“ in *International Conference on Machine Learning (ICML)*, 2018.
- [8] Y.-C. Chen, Y.-Y. Lin, M.-H. Yang und J.-B. Huang, „CrDoCo: Pixel-Level Domain Transfer With Cross-Domain Consistency,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [9] B. Benjdira, Y. Bazi, A. Koubaa und K. Ouni, „Unsupervised Domain Adaptation Using Generative Adversarial Networks for Semantic Segmentation of Aerial Images,“ *Remote Sensing*, Bd. 11, Nr. 11, 2019.
- [10] B. Fang, R. Kou, L. Pan und P. Chen, „Category-Sensitive Domain Adaptation for Land Cover Mapping in Aerial Scenes,“ *Remote Sensing*, Bd. 11, Nr. 22, 2019.
- [11] D. Wittich und F. Rottensteiner, „Appearance based deep domain adaptation for the classification of aerial images,“ *ISPRS Journal of Photogrammetry and Remote Sensing*, Bd. 180, pp. 82-102, 2021.

- [12] H. E. Landsberg, *The Urban Climate*, Academic Press, 1981.
- [13] T. R. Oke, „The urban energy balance,“ *Progress in Physical Geography: Earth and Environment*, Bd. 12, Nr. 4, pp. 471-508, 1988.
- [14] P. Rajagopalan, K. C. Lim und E. Jamei, „Urban heat island and wind flow characteristics of a tropical city,“ *Solar Energy*, Bd. 107, pp. 159-170, 2014.
- [15] E. Alexandri und P. Jones, „Temperature decreases in an urban canyon due to green walls and green roofs in diverse climates,“ *Building and Environment*, Bd. 43, Nr. 4, pp. 480-493, 2008.
- [16] D. E. Bowler, L. Buyung-Ali, T. M. Knight und A. S. Pullin, „Urban greening to cool towns and cities: A systematic review of the empirical evidence,“ *Landscape and Urban Planning*, Bd. 97, Nr. 3, pp. 147-155, 2010.
- [17] A. Berland, S. A. Shiflett, W. D. Shuster, A. S. Garmestani, H. C. Goddard, D. L. Herrmann und M. E. Hopton, „The role of trees in urban stormwater management,“ *Landscape and Urban Planning*, Bd. 162, pp. 167-177, 2017.
- [18] K. P. Beckett, P. H. Freer-Smith und G. Taylor, „Urban woodlands: their role in reducing the effects of particulate pollution,“ *Environmental Pollution*, Bd. 99, Nr. 3, pp. 347-360, 1998.
- [19] J.-P. L. Savard, P. Clergeau und G. Mennechez, „Biodiversity concepts and urban ecosystems,“ *Landscape and Urban Planning*, Bd. 48, Nr. 3-4, pp. 131-142, 2000.
- [20] S. H. Faeth, C. Bang und S. Saari, „Urban biodiversity: patterns and mechanisms,“ *Annals of the New York Academy of Sciences*, Bd. 1223, Nr. 1, pp. 69-81, 2011.
- [21] V. V. Salomonson, W. L. Barnes, P. W. Maymon, H. E. Montgomery und H. Ostrow, „MODIS: Advanced facility instrument for studies of the Earth as a system,“ *IEEE Transactions on Geoscience and Remote Sensing*, Bd. 27, Nr. 2, pp. 145-153, 1989.
- [22] M. Immitzer, F. Vuolo und C. Atzberger, „First Experience with Sentinel-2 Data for Crop and Tree Species Classifications in Central Europe,“ *Remote Sensing*, Bd. 8, Nr. 3, 2016.
- [23] J. W. Rouse, R. H. Haas, J. A. Schell und D. W. Deering, „Monitoring vegetation systems in the Great Plains with ERTS,“ in *Goddard Space Flight Center 3d ERTS-1 Symp.*, 1974.
- [24] R. S. DeFries und J. R. G. Townshend, „NDVI-derived land cover classifications at a global scale,“ *International Journal of Remote Sensing*, Bd. 15, Nr. 17, pp. 3567-3586, 1994.

- [25] T. N. Carlson und D. A. Ripley, „On the relation between NDVI, fractional vegetation cover, and leaf area index,“ *Remote Sensing of Environment*, Bd. 62, Nr. 3, pp. 241-252, 1997.
- [26] N. Pettorelli, J. O. Vik, A. Mysterud, J.-M. Gaillard, C. J. Tucker und N. C. Stenseth, „Using the satellite-derived NDVI to assess ecological responses to environmental change,“ *Trends in ecology & evolution*, Bd. 20, Nr. 9, pp. 503-510, 2005.
- [27] T. Blaschke, „Object based image analysis for remote sensing,“ *ISPRS Journal of Photogrammetry and Remote Sensing*, Bd. 65, Nr. 1, pp. 2-16, 2010.
- [28] R. Pu, S. Landry und Q. Yu, „Object-based urban detailed land cover classification with high spatial resolution IKONOS imagery,“ *International Journal of Remote Sensing*, Bd. 32, Nr. 12, pp. 3285-3308, 2011.
- [29] R. Pu und S. Landry, „A comparative analysis of high spatial resolution IKONOS and WorldView-2 imagery for mapping urban tree species,“ *Remote Sensing of Environment*, Bd. 124, pp. 516-533, 2012.
- [30] X. Li und G. Shao, „Object-based urban vegetation mapping with high-resolution aerial photography as a single data source,“ *International Journal of Remote Sensing*, Bd. 34, Nr. 3, pp. 771-789, 2013.
- [31] T. Kattenborn, J. Leitloff, F. Schiefer und S. Hinz, „Review on Convolutional Neural Networks (CNN) in vegetation remote sensing,“ *ISPRS Journal of Photogrammetry and Remote Sensing*, Bd. 173, pp. 24-49, 2021.
- [32] T. Kattenborn, J. Eichel und F. E. Fassnacht, „Convolutional Neural Networks enable efficient, accurate and fine-grained segmentation of plant species and communities from high-resolution UAV imagery,“ *Scientific Reports*, Bd. 9, 2019.
- [33] F. Schenkel und W. Middelmann, „Domain Adaptation for Semantic Segmentation Using Convolutional Neural Networks,“ in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [34] F. Schenkel und W. Middelmann, „Domain Adaptation for Semantic Segmentation of Aerial Imagery Using Cycle-Consistent Adversarial Networks,“ in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2020.
- [35] F. Schenkel, S. Hinz und W. Middelmann, „Style transfer-based domain adaptation for vegetation segmentation with optical imagery,“ *Applied Optics*, Bd. 60, Nr. 22, pp. F109-F117, 2021.
- [36] R. Rojas, *Theorie der neuronalen Netze: Eine systematische Einführung*, Springer, 2013.

- [37] A. K. Jain, J. Mao und K. M. Mohiuddin, „Artificial neural networks: a tutorial,“ *Computer*, Bd. 29, Nr. 3, pp. 31-44, 1996.
- [38] W. S. McCulloch und W. Pitts, „A logical calculus of the ideas immanent in nervous activity,“ *The bulletin of mathematical biophysics*, Bd. 5, pp. 115-133, 1943.
- [39] D. E. Rumelhart, G. E. Hinton und R. J. Williams, „Learning representations by back-propagating errors,“ *Nature*, Bd. 323, pp. 533-536, 1986.
- [40] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard und L. D. Jackel, „Backpropagation Applied to Handwritten Zip Code Recognition,“ *Neural Computation*, Bd. 1, Nr. 4, pp. 541-551, 1989.
- [41] X. Glorot, A. Bordes und Y. Bengio, „Deep Sparse Rectifier Neural Networks,“ in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 2011.
- [42] K. Fukushima, „Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,“ *Biological Cybernetics*, Bd. 36, Nr. 4, pp. 193-202, 1980.
- [43] A. Krizhevsky, I. Sutskever und G. E. Hinton, „ImageNet classification with deep convolutional neural networks,“ *Communications of the ACM*, Bd. 60, Nr. 6, pp. 84-90, 2017.
- [44] K. Simonyan und A. Zisserman, „Very Deep Convolutional Networks for Large-Scale Image Recognition,“ in *International Conference on Learning Representations (ICLR)*, 2015.
- [45] K. He, X. Zhang, S. Ren und J. Sun, „Deep Residual Learning for Image Recognition,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [46] G. Huang, Z. Liu, L. van Der Maaten und K. Q. Weinberger, „Densely Connected Convolutional Networks,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [47] J. Long, E. Shelhamer und T. Darrell, „Fully Convolutional Networks for Semantic Segmentation,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [48] O. Ronneberger, P. Fischer und T. Brox, „U-net: Convolutional Networks for Biomedical Image Segmentation,“ in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

- [49] V. Badrinarayanan, A. Kendall und R. Cipolla, „Segnet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,“ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Bd. 39, Nr. 12, pp. 2481-2495, 2017.
- [50] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff und H. Adam, „Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation,“ in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- [51] S. Ren, K. He, R. Girshick und J. Sun, „Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,“ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Bd. 39, Nr. 6, pp. 1137-1149, 2017.
- [52] K. He, G. Gkioxari, P. Dollár und R. Girshick, „Mask R-CNN,“ in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [53] J. Redmon, S. Divvala, R. Girshick und A. Farhadi, „You Only Look Once: Unified, Real-Time Object Detection,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [54] Y. LeCun, Y. Bengio und G. Hinton, „Deep learning,“ *Nature*, Bd. 521, pp. 436-444, 2015.
- [55] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever und R. Salakhutdinov, „Dropout: A Simple Way to Prevent Neural Networks from Overfitting,“ *The Journal of Machine Learning Research*, Bd. 15, Nr. 1, pp. 1929-1958, 2014.
- [56] S. Ioffe und C. Szegedy, „Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,“ in *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.
- [57] G. Csurka, D. Larlus und F. Perronnin, „What is a good evaluation measure for semantic segmentation?,“ in *Proceedings of the British Machine Vision Conference (BMVC)*, 2013.
- [58] W. J. Carper, T. M. Lillesand und R. W. Kiefer, „The use of intensity-hue-saturation transformations for merging SPOT panchromatic and multispectral image data,“ *Photogrammetric Engineering & Remote Sensing*, Bd. 56, Nr. 4, pp. 459-467, 1990.
- [59] T.-M. Tu, P. S. Huang, C.-L. Hung und C.-P. Chang, „A fast intensity-hue-saturation fusion technique with spectral adjustment for IKONOS imagery,“ *IEEE Geoscience and Remote Sensing Letters*, Bd. 1, Nr. 4, pp. 309-312, 2004.
- [60] D.-A. Clevert, T. Unterthiner und S. Hochreiter, „Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs),“ in *International Conference on Learning Representations (ICLR)*, 2015.

- [61] D. P. Kingma und J. Ba, „Adam: A Method for Stochastic Optimization,“ in *International Conference on Learning Representations (ICLR)*, 2015.
- [62] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta und A. A. Bharath, „Generative Adversarial Networks: An Overview,“ *IEEE Signal Processing Magazine*, Bd. 35, Nr. 1, pp. 53-65, 2018.
- [63] P. Isola, J.-Y. Zhu, T. Zhou und A. A. Efros, „Image-to-Image Translation with Conditional Adversarial Networks,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [64] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan und D. Krishnan, „Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks,“ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [65] H.-W. Heldt und B. Piechulla, *Plant Biochemistry*, Academic Press, 2021.
- [66] K. J. McCree, „The action spectrum, absorptance and quantum yield of photosynthesis in crop plants,“ *Agricultural Meteorology*, Bd. 9, pp. 191-216, 1971.
- [67] A. R. Huete, „A soil-adjusted vegetation index (SAVI),“ *Remote Sensing of Environment*, Bd. 25, Nr. 3, pp. 295-309, 1988.
- [68] A. Huete, K. Didan, T. Miura, E. P. Rodriguez, X. Gao und L. G. Ferreira, „Overview of the radiometric and biophysical performance of the MODIS vegetation indices,“ *Remote Sensing of Environment*, Bd. 83, Nr. 1-2, pp. 195-213, 2002.
- [69] F. Bornemann und T. März, „Fast Image Inpainting Based on Coherence Transport,“ *Journal of Mathematical Imaging and Vision*, Bd. 28, pp. 259-278, 2007.
- [70] A. A. Gitelson, Y. J. Kaufman und M. N. Merzlyak, „Use of a green channel in remote sensing of global vegetation from EOS-MODIS,“ *Remote Sensing of Environment*, Bd. 58, Nr. 3, pp. 289-298, 1996.
- [71] S. K. McFeeters, „The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features,“ *International Journal of Remote Sensing*, Bd. 17, Nr. 7, pp. 1425-1432, 1996.
- [72] D. W. Hancock und C. T. Dougherty, „Relationships between Blue- and Red-based Vegetation Indices and Leaf Area and Yield of Alfalfa,“ *Crop Science*, Bd. 47, Nr. 6, pp. 2547-2556, 2007.

- [73] Y. Zha, J. Gao und S. Ni, „Use of normalized difference built-up index in automatically mapping urban areas from TM imagery,“ *International Journal of Remote Sensing*, Bd. 24, Nr. 3, pp. 583-594, 2003.



## 10 Abbildungsverzeichnis

|   |    |
|---|----|
| Abbildung 1: Die für den Bereich des Transferlernens relevanten Kriterien. Auf der Datenebene handelt es sich um die Quell- und Zieldomäne und auf der Anwendungsebene um die Quell- und Zielanwendung. ....  | 12 |
| Abbildung 2: Die möglichen Varianten des Transferlernens: Induktives, transduktives (Domain Adaptation) und unüberwachtes Transferlernen. ....  | 13 |
| Abbildung 3: Biochemische Prozesse führen bei Vegetation zu bestimmten Absorptions- und Reflexionscharakteristika und stellen sich dementsprechend in den einzelnen Farbkanäle von Sensoren dar und bilden ein entsprechendes spezifisches Spektrum im Pixelwertebereich. ....  | 17 |
| Abbildung 4: Blau hinterlegt: Methoden zur Domain Adaptation. Grau hinterlegt: Methode zur Quelldomäne-Datensatzerstellung. Roter Rahmen: Überwachtes Lernen (Annotationen in Zieldomäne vorhanden). Blauer Rahmen: Unüberwachtes Lernen (Annotationen in Zieldomäne nicht vorhanden) ....  | 20 |
| Abbildung 5: Schematische Darstellung eines neuronalen Netzes mit einer Eingabeschicht bestehend aus drei Neuronen auf der linken Seite, einer verdeckten Schicht mit fünf Neuronen in der Mitte und einer Ausgabeschicht, bestehend aus einem einzelnen Neuron auf der rechten Seite. Alle Neuronen einer jeweiligen Schicht sind mit allen Neuronen der vorherigen und darauffolgenden Schicht verbunden..... | 24 |
| Abbildung 6: Funktionsweise eines künstlichen Neurons mit mehreren Eingangssignalen $x_i$ , Gewichten $w_{ji}$ , einer Summenfunktion $\Sigma$ , der Summe der Netzeingabe $x_j$ , der Aktivierungsfunktion $\Theta$ und der entsprechenden Ausgabe des Neurons $y_j$ . ....  | 25 |
| Abbildung 7: Schematische Darstellung der Funktionsweise eine Convolutional-Schicht mit einer Filter-Größe von $3 \times 3$ . ....  | 30 |
| Abbildung 8: Schematische Darstellung der Funktionsweise eine Pooling-Schicht mit einer entsprechenden Fenster-Größe von $2 \times 2$ .....   | 31 |
| Abbildung 9: Schematische Darstellung des VGG16-Klassifikationsnetzwerks, bestehend aus fünf Convolutional-Blöcken mit Convolutional-Schichten und Pooling-Schichten zur Generierung von Features sowie drei vollverbundenen Schichten für die anschließende Klassifikation. ....   | 33 |
| Abbildung 10: Darstellung einer Konfusionsmatrix als Grundlage zur Ermittlung der Größen für die Metriken zur Auswertung der Ergebnisse. ....   | 35 |
| Abbildung 11: Schematische Darstellung der Schnittmenge und der Vereinigungsmenge von A und B. ....   | 36 |
| Abbildung 12: Oben links: RGB-Bild von ISPRS Potsdam-Datensatz. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer. Unten links: RGB-Bild von Luftbilddatensatz Herbst. Unten rechts: RGB-Bild von Satellitenbilddatensatz Sommer. ....   | 41 |
| Abbildung 13: Die Domain Adaptation erfolgt vom ISPRS-Datensatz als Quelldomäne zum Luftbilddatensatz Sommer als Zieldomäne. Da es sich um überwachtes Lernen handelt, ist für alle RGB-Bilder, welche im Trainingsvorgang verwendet werden auch eine entsprechende Annotation vorhanden.....   | 43 |

|  |   |
|--|---|
| Abbildung 14: Dargestellt ist die Encoder-Decoder-Struktur für die semantische Segmentierung, welche auf dem Quelldomäne-Datensatz trainiert wird.....   | 44  |
| Abbildung 15: Dargestellt ist die Ergänzung der Encoder-Decoder-Struktur um die für den Anpassungsprozess eingesetzten Adaptionenmodule. Dazu wird jede Skip-Connection, der auf der Quelldomäne trainierten Grundstruktur, durch ein entsprechendes Modul ersetzt. ....   | 45  |
| Abbildung 16: Dargestellt ist auf der linken Seite von oben nach unten: RGB-Bild, Ground Truth und Ergebnisse der semantischen Segmentierung des ISPRS Potsdam-Datensatzes und auf der rechten Seite von oben nach unten: RGB-Bild, Ground Truth und Ergebnisse der semantischen Segmentierung von Luftbilddatensatz Sommer..... | 50  |
| Abbildung 17: Die Domain Adaptation erfolgt vom Luftbilddatensatz Sommer als Quelldomäne zum Luftbilddatensatz Herbst als Zieldomäne. Da es sich um unüberwachtes Lernen handelt, ist für alle RGB-Bilder der Zieldomäne, welche im Trainingsvorgang verwendet werden keine Annotation vorhanden.....                            | 51  |
| Abbildung 18: Darstellung eines Generative Adversarial Networks mit einem Vektor $z$ , einem Generator $G$ , der Ausgabe des Generators $x'$ , einem Diskriminator $D$ und einem Datensatz $x$ .....   | 52  |
| Abbildung 19: Darstellung eines Conditional Adversarial Networks mit einem Vektor $z$ , einer Bedingung $c$ , einem Generator $G$ , der Ausgabe des Generators $x'$ , einem Diskriminator $D$ und einem Datensatz $x$ . ....   | 54  |
| Abbildung 20: Darstellung eines Cycle-Consistent Adversarial Networks mit den Generatoren $G$ und $F$ , den Diskriminatoren $D_x$ und $D_y$ und den Datensätzen $x$ und $y$ . ....   | 55  |
| Abbildung 21: Während des Trainings von $Seg_T$ kommen sowohl vom Generator $G$ erzeugte Bilder wie auch unveränderte Bilder der Quelldomäne zum Einsatz. ....   | 56  |
| Abbildung 22: Während des Trainings kommen mehrere Varianten des Generators $G$ zum Einsatz um das Segmentierungsnetzwerk $Seg_T$ zu trainieren. ....  | 57  |
| Abbildung 23: Oben links: RGB-Bild des Luftbilddatensatz Herbst. Oben rechts: Ground Truth, Mitte links: Ergebnis ohne Anpassung. Mitte rechts: Ergebnis der Style-Transfer-Methode. Unten links: Ergebnis der Fine-Tuning-Methode. Unten rechts: Ergebnis der kontinuierlichen Methode. ....                                    | 61  |
| Abbildung 24: Auf Grundlage von Vegetationsindices wird durch den Einsatz von CNNs zur semantischen Segmentierung eine teilautomatisierte Annotation ermöglicht. ....  | 63  |
| Abbildung 25: Darstellung der PAR-Kurve. Aufgetragen ist dabei schematisch die relative Quantenausbeute über die Wellenlänge in einem Bereich von 350 bis 750 nm. ....   | 64  |
| Abbildung 26: Erstellen eines Trainingsdatensatzes für das RGB-Segmentierungsnetzwerk. ....  | <b>Fehler! Textmarke nicht definiert.</b> |
| Abbildung 27: Trainieren des RGB-Segmentierungsnetzwerks. ....   | 70  |
| Abbildung 28: Erstellen des Quelldomäne-Datensatzes.....   | 71  |
| Abbildung 29: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem NDVI im Vergleich zum Schwellwertverfahren angegeben als $F_1$ -Wert und der IoU.....   | 72  |
| Abbildung 30: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von NDVI-Schwellwertverfahren bei 0,25. Unten links: Ergebnis von NDVI-  |   |

|   |    |
|---|----|
| Schwelldwertverfahren bei 0,4. Unten rechts: Ergebnis von entsprechendem CNN-Verfahren. ....  | 73 |
| Abbildung 31: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem SAVI im Vergleich zum Schwelldwertverfahren angegeben als $F_1$ -Wert und der IoU.....   | 74 |
| Abbildung 32: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von SAVI-Schwelldwertverfahren bei 0,06. Unten links: Ergebnis von SAVI-Schwelldwertverfahren bei 0,1. Unten recht: Ergebnis von entsprechendem CNN-Verfahren. ....       | 75 |
| Abbildung 33: Ergebnisse für die Precision und den Recall bei der Segmentierung von Vegetation basierend auf dem EVI im Vergleich zum Schwelldwertverfahren angegeben als $F_1$ -Wert und der IoU.....  | 76 |
| Abbildung 34: Oben links: RGB-Bild des Luftbilddatensatz Sommer. Oben rechts: entsprechendes NIR-RG-Bild. Mitte links: Ground Truth. Mitte rechts: Ergebnis von EVI-Schwelldwertverfahren bei 0,05. Unten links: Ergebnis von EVI-Schwelldwertverfahren bei 0,1. Unten rechts: Ergebnis von entsprechendem CNN-Verfahren.....         | 77 |
| Abbildung 35: Bei dieser Auswertung kamen als Quelldomäne der Luftbilddatensatz Sommer und als Zieldomäne der Satellitenbilddatensatz Sommer und der Luftbilddatensatz Herbst zum Einsatz. Da es sich um unüberwachte Domain Adaptation handelt, waren für das Training der CNNs keine Annotationen in der Zieldomäne vorhanden. .... | 79 |
| Abbildung 36: Schematische Darstellung des Trainings- und Auswerteworkflows der Methode für die Quell- und Zieldomäne. ....   | 81 |
| Abbildung 37: Oben links: RGB-Bild von Luftbilddatensatz Sommer. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer nach dem Style-Transfer. Unten links: RGB-Bild von Satellitenbilddatensatz Sommer. Unten rechts: RGB-Bild von Satellitenbilddatensatz Sommer nach dem Style-Transfer.....   | 84 |
| Abbildung 38: Oben links: RGB-Bild von Luftbilddatensatz Sommer. Oben rechts: RGB-Bild von Luftbilddatensatz Sommer nach dem Style-Transfer. Unten links: RGB-Bild von Luftbilddatensatz Herbst. Unten rechts: RGB-Bild von Luftbilddatensatz Herbst nach dem Style-Transfer. ....  | 85 |
| Abbildung 39: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: NDVI der ursprünglichen Bildes. Unten links: NDVI des angepassten Bildes ohne $L_{Index}$ . Unten rechts: NDVI des angepassten Bildes mit $L_{Index}$ . ....   | 88 |
| Abbildung 40: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: GNDVI der ursprünglichen Bildes. Unten links: GNDVI des angepassten Bildes ohne $L_{Index}$ . Unten rechts: GNDVI des angepassten Bildes mit $L_{Index}$ . ....  | 89 |
| Abbildung 41: Oben links: NIR-RG-Bild des Satellitenbilddatensatz Sommer. Oben rechts: BNDVI der ursprünglichen Bildes. Unten links: BNDVI des angepassten Bildes ohne $L_{Index}$ . Unten rechts: BNDVI des angepassten Bildes mit $L_{Index}$ . ....  | 90 |
| Abbildung 42: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: NDVI der ursprünglichen Bildes. Unten links: NDVI des angepassten Bildes ohne $L_{Index}$ . Unten rechts: NDVI des angepassten Bildes mit $L_{Index}$ . ....   | 91 |

Abbildung 43: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: GNDVI der ursprünglichen Bildes. Unten links: GNDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: GNDVI des angepassten Bildes mit  $L_{Index}$ . ..... 92

Abbildung 44: Oben links: NIR-RG-Bild des Luftbilddatensatz Herbst. Oben rechts: BNDVI der ursprünglichen Bildes. Unten links: BNDVI des angepassten Bildes ohne  $L_{Index}$ . Unten rechts: BNDVI des angepassten Bildes mit  $L_{Index}$ . ..... 93

Abbildung 45: Oben links: Grount Truth des Satellitenbilddatensatz Sommer. Oben rechts: Ergebnisse der Segmentierung ohne Anpassung. Unten links: Ergebnisse der Segmentierung ohne  $L_{Index}$ . Unten rechts: Ergebnisse der Segmentierung mit  $L_{Index}$ . ..... 95

Abbildung 46: Oben links: Grount Truth des Luftbilddatensatz Herbst. Oben rechts: Ergebnisse der Segmentierung ohne Anpassung. Unten links: Ergebnisse der Segmentierung ohne  $L_{Index}$ . Unten rechts: Ergebnisse der Segmentierung mit  $L_{Index}$ . ..... 96

## 11 Tabellenverzeichnis

|   |    |
|---|----|
| Tabelle 1: Übersicht über unterschiedliche Aktivierungsfunktionen die im Bereich neuronaler Netzte zum Einsatz kommen.....  | 28 |
| Tabelle 2: Aufbau eines VGG16-Klassifikationsnetzwerks.....   | 34 |
| Tabelle 3: Gemeinsamkeiten und Unterschiede zwischen Luftbilddatensatz Sommer und Luftbilddatensatz Herbst.....   | 39 |
| Tabelle 4: Die entsprechenden Wellenlängen der einzelnen Kanäle des WorldView-2-Sensors.....  | 40 |
| Tabelle 5: Gemeinsamkeiten und Unterschiede zwischen Luftbilddatensatz Sommer und Satellitenbilddatensatz Sommer.....   | 40 |
| Tabelle 6: Anzahl der Filter und die Größe der entsprechenden Feature-Maps bei den jeweiligen Adaptionmodulen.....  | 47 |
| Tabelle 7: Ergebnisse der Fine-Tuning-Methode und der Adaptionmodule-Methode als Gesamtgenauigkeit in Prozent für die unterschiedlichen Teildatensätze.....   | 49 |
| Tabelle 8: Ergebnisse der Fine-Tuning-Methode und der Adaptionmodule-Methode als durchschnittlicher $F_1$ -Wert für die unterschiedlichen Teildatensätze.....   | 49 |
| Tabelle 9: Ergebnisse der Style-Transfer-Methode, der Fine-Tuning-Methode und der kontinuierlichen Trainingsmethode als Gesamtgenauigkeit in Prozent.....   | 59 |
| Tabelle 10: Ergebnisse der Style-Transfer-Methode, der Fine-Tuning-Methode und der kontinuierlichen Trainingsmethode als $F_1$ -Wert.....   | 60 |
| Tabelle 11: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem NDVI angegeben als $F_1$ -Wert und der IoU.....   | 72 |
| Tabelle 12: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem SAVI angegeben als $F_1$ -Wert und der IoU.....   | 74 |
| Tabelle 13: Ergebnisse für die Segmentierung von Vegetation mit der CNN-Methode basierend auf dem EVI angegeben als $F_1$ -Wert und der IoU.....  | 76 |
| Tabelle 14: Abweichung der Vegetationsindices vom Ursprungsbild als mittlere quadratische Abweichung (MQA) angegeben in Prozent mit dem Satellitenbilddatensatz Sommer als Zieldomäne.....                                | 86 |
| Tabelle 15: Abweichung der Vegetationsindices vom Ursprungsbild als mittlere quadratische Abweichung (MQA) angegeben in Prozent mit dem Luftbilddatensatz Herbst als Zieldomäne.....                                      | 87 |
| Tabelle 16: Ergebnisse der Segmentierung für den Satellitenbilddatensatz Sommer angegeben als $F_1$ -Wert und IoU für die Fälle: Ohne Anpassung, Style-Transfer ohne $L_{Index}$ und Style-Transfer mit $L_{Index}$ ..... | 94 |
| Tabelle 17: Ergebnisse der Segmentierung für den Luftbilddatensatz Herbst angegeben als $F_1$ -Wert und IoU für die Fälle: Ohne Anpassung, Style-Transfer ohne $L_{Index}$ und Style-Transfer mit $L_{Index}$ .....       | 94 |