Semantische Segmentierung der Umgebung auf Basis von 3D-Daten

Simon Kuhn

Wissenschaftliche Arbeit im Zuge des Fachwissenschaftlichen Seminares

Erstprüfer:	Prof. Dr. Christian Pfitzner
Betreuer:	Prof. Dr. Christian Pfitzner
Ausgabedatum:	23.03.2023
Abgabedatum:	31.08.2023

Inhaltsverzeichnis

1 Einleitung

1.1 Hintergrund und Motivation

In den letzten Jahren wurden im Bereich der Computer Vision enorme Fortschritte erzielt, was insbesondere im Bereich autonomer mobiler Plattformen von großer Bedeutung ist. Hierbei spielt die präzise und schnelle Erfassung und Interpretation der Umgebung eine zentrale Rolle, um eine zuverlässige Navigation zu ermöglichen. In diesem Zusammenhang hat die semantische Segmentierung der Umgebung auf Basis von 3D-Daten eine immer größere Bedeutung erlangt. Die semantische Segmentierung beschreibt ein Verfahren, welches eine automatische Klassifizierung von Objekten und Strukturen in der Umgebung auf Pixelebene bewirkt. Dabei wird jedem Pixel oder Voxel in einem 3D-Datensatz eine bestimmte semantische Bedeutung zugeordnet. In dieser Arbeit werden die Grundlagen der semantische Segmentierung der Umgebung auf Basis von 3D-Daten beleuchtet. Dabei sollen verschiedene Methoden und Ansätze für die semantischen Segmentierung, sowie bestehende Probleme, Potentiale und Anwendungsbereiche dargestellt und bewertet werden.[1]

1.2 Problemstellungen und aktueller Stand der Technik

Obwohl es im Bereich der Computer Vision beträchtliche Fortschritte gegeben hat, bestehen nach wie vor herausfordernde Aspekte. Eine zentrale Herausforderung besteht in der Komplexität der Umgebung. In einem 3D-Umfeld interagieren zahlreiche verschiedene Objekte und Strukturen miteinander und beeinflussen sich gegenseitig. Es ist schwierig, all diese Details genau zu erfassen und zu segmentieren, besonders wenn die Daten unvollständig oder fehlerhaft sind. Ein weiteres Problem ist die Notwendigkeit einer Segmentierung in Echtzeit in Bereichen der mobilen Plattformen. Einige Anwendungsbereiche, wie autonome Fahrzeuge, stellen dabei

besondere Anforderungen, wie eine präzise Echtzeit-Verarbeitung großer Datenmengen. Aktuelle Verfahren weißen noch immer eine begrenzte Genauigkeit bei der Segmentierung auf. Es gibt noch immer Schwierigkeiten bei der Unterscheidung zwischen ähnlichen Objekten, insbesondere wenn sie sich in Form oder Größe ähneln. Es ist schwierig, alle subtilen Unterschiede zu erfassen, die für eine präzise Segmentierung notwendig sind [2]. Ein großteil der Forschung im Bereich der Semantischen Segmentierung zielt auf Verbesserungen in diesen Bereichen ab. Dabei gelten besonders Deep-learning Methoden und Convolutional Neural Networks (CNNs) als vielversprechende Ansätze, um die Komplexität der Umgebung besser zu erfassen und zu bewältigen [3, 4].

2 Sensoren zur Erfassung von 3D-Daten

2.1 LiDAR-Sensoren

LiDAR-Sensoren (Light Detection and Ranging) stellen eine weit verbreitete Technologie zur Erfassung von 3D-Daten dar. Diese senden einen Laserstrahl aus, welcher von Objekten in der Umgebung reflektiert wird. Die Distanz des reflektierenden Objektes kann dabei über Time of Flight (TOF) oder über die Phase der reflektierten Lichtwelle gemessen werden. Während TOF-LiDAR die Distanz zum Objekt über eine Messung der Laufzeit des Laserstrahls bestimmt, erfolgt die Entfernungsmessung beim phasenbasierten LiDAR über die Auswertung der Phasenverschiebung der vom Objekt reflektierten Lichtwelle. Beide Methoden können hochgenaue Entfernungen zu den reflektierenden Objekten erfassen. Sie können detaillierte 3D-Punktwolken erzeugen, welche die Geometrie und räumliche Verteilung von Objekten in der Umgebung darstellen. Zusätzlich lassen sich LiDAR-Sensoren in Scanning-LiDAR und Non-Scanning-LiDAR untergliedern. Non-Scanning-LiDAR nutzt dabei einen statischen Laserstrahl, während Scanning-LiDAR einen sich bewegenden Laserstrahl nutzt und somit einen größeren Arbeitsbereich abdecken kann. [5]

2.2 Tiefenkameras

Unter dem Begriff Tiefenkamera lassen sich verschiedene Verfahren aufführen, welche unterschiedliche Funktionsweisen besitzen, um Tiefeninformationen einer Szene zu bestimmen. Im Bereich der semantische Segmentierung kommen besonders Kamerasysteme, die auf Stereo-Vision, Time-of-Flight oder Structured Light basieren zum Einsatz.

Kamerasysteme, die auf dem Prinzip der Stereo-Vision basieren, werden als Stereo Kameras bezeichnet. Bei diesen werden zwei räumlich getrennte Kameras verwendet, die gemeinsam Bilder von derselben Szene aus zwei leicht unterschiedlichen Perspektiven aufnehmen. Der dabei

entstehende horizontale Versatz der beiden Bilder wird als Disparität bezeichnet. Aus dieser lassen sich Tiefeninformationen des betrachteten Objektes durch Triangulation bestimmen [6].

Kameras die auf dem Time of Flight(TOF) Prinzip basieren, senden eine modulierte Lichtwelle im Infrarotbereich aus. Diese wird, wie beim LiDAR-Sensor, von Objekten in der Umgebung reflektiert und ermöglicht es Tiefeninformationen zu berechnen. Dies kann nur über die Laufzeit oder zusätzlich über die Phasenverschiebung der Infrarotwelle geschehen [7]. Diese können zusätzlich mit einer RGB-Kamera ausgestattet sein und sind dann unter dem Begriff RGB-D Kameras bekannt. Diese haben häufig eine höhere räumliche Auflösung und sind in der Lage zusätzlich Farbinformationen aufzunehmen, besitzen jedoch einen deutlich kleineren Arbeitsbereich [8]. Ein weiteres Verfahren basiert auf Structured Light. Bei diesem Verfahren wird ein spezielles 2D-Muster auf das zu betrachtende Objekt projiziert. Aus dessen Verzerrung lässt sich auf 3D-Information schließen [9].

2.3 Passive und aktive Sensoren

Die genannten Sensoren lassen sich zusätzlich in zwei Klassen unterteilen. Aktive Sensoren, wie LIDAR-Sensoren, senden selbst Energie in Form von einer Lichtwelle aus, um Informationen über die Umgebung zu sammeln. Im Gegensatz dazu arbeiten passive Sensoren, wie Stereokameras, ohne aktiv Energie auszusenden, sondern nutzen lediglich das natürliche Licht, das von der Umgebung reflektiert wird. Der Vorteil von aktiven Sensoren besteht darin, dass sie unabhängig von der Umgebungshelligkeit arbeiten und auch bei Dunkelheit eingesetzt werden können. Passive Sensoren hingegen können bei schlechten Lichtverhältnissen Schwierigkeiten haben, genaue Tiefeninformationen zu liefern. Es ist jedoch anzumerken, dass passive Sensoren in der Regel kostengünstiger sind und eine höhere räumliche Auflösung bieten können.

2.4 Auswahl von Sensoren für die semantische Segmentierung

Die Auswahl geeigneter Sensoren für die Semantische Segmentierung ist von deren Einsatzbereich abhängig. Dabei sind Faktoren wie die Umgebungsbedingungen, Budget und gewünschte Ergebnisse ausschlaggebend. Je nach Szenario sind werden unterschiedliche Anforderungen an die Genauigkeit, die räumliche Auflösung, die Reichweite, sowie die Echtzeitfähigkeit gestellt.

Anwendungen im Bereich der autonomen Fahrzeuge benötigen beispielsweise häufig Sensoren mit hoher Reichweite und Genauigkeit, während Anwendungen im Innenbereich möglicherweise Sensoren mit höherer räumlicher Auflösung benötigen. Kann es im geplanten Einsatzgebiet zu wechselnden Umgebungsbedingungen wie Wetter- oder Beleuchtungsveränderungen kommen, sind oft aktive Sensoren geeigneter, um unabhängig von äußeren Einflüssen funktionieren zu können. Durch ihre unterschiedlichen Eigenschaften können Sensoren besser für die Erkennung bestimmter Objekte geeignet sein. Außerdem ist das Budget bei der Sensorwahl zu berücksichtigen, da sich die Sensoren stark in ihren Kosten unterscheiden können.

3 Datengrundlage und Vorverarbeitung

3.1 3D-Datenformate und Datentypen

Bei der semantischen Segmentierung von 3D-Daten spielen die zugrunde liegenden Datenformate eine elementare Rolle. Diese bilden die Grundlage für die Erfassung, Speicherung und Verarbeitung von 3D-Daten, die für die semantische Segmentierung verwendet werden. In diesem Kapitel werden die beiden verbreitetsten 3D-Datenformate und Datentypen untersucht, die in der Forschung und Praxis eingesetz werden.

Ein wichtiges 3D-Datenformat ist das Punktewolkenformat, das unter anderem von LiDAR-Sensoren erzeugt wird. Eine Punktwolke ist eine Sammlungen von 3D-Punkten, die die Oberfläche von Objekten in der Umgebung darstellen. Jeder Punkt stellt dabei eine reflektierte Lichtwelle da. Sie können in verschiedenen Dateiformaten gespeichert werden, häufig geschieht dies im binären LAS-Format, das speziell für die Speicherung und den Austausch von LiDAR-Punktwolken entwickelt wurde. Diese Formate ermöglichen die Speicherung von großen Mengen an Punkten mit 3D-Koordinaten, Intensitätsinformationen und weiteren Attributen, die zur semantischen Segmentierung verwendet werden können [**.2020b**]. Punktwolken weißen dadurch eine stark variable Punktdichte auf. Dies lässt sich durch Faktoren, wie eine ungleichmäßige Abtastung des Raumes, der Verdeckung von Objekten und der relativen Ausrichtung des Objektes zum Sensor begründen. Das Punktwolkenformat hat aufgrund der großen Menge an Punkten, die durch einen LiDAR-Sensor erzeugt werden, einen sehr hohen Speicherbedarf, was zu einer rechenintensiven Verarbeitung der Daten führen kann [10].

Neben dem Punktewolkenformat wird häufig auch das Voxel-Gitter Format für die semantische Segmentierung verwendet. Voxel sind volumetrische Elemente, die den Raum in einem dreidimensionalen Gitter unterteilen. Jedes Voxel enthält dabei Informationen über Materialeigenschaften, Farbe oder Textur des Objektes an diesem Punkt im Raum. Das Datenformat bietet

dabei eine effektive Möglichkeit, komplexe dreidimensionale Strukturen darzustellen und weiter zu analysieren. Die Voxeldichte im Raum kann dabei frei gewählt werden und bestimmt so die Auflösung und den Speicherbedarf des Datensatzes.

3.2 Vorverarbeitungsschritte wie Filterung, Normalenberechnung und Downsampling

Die Vorverarbeitung von 3D-Daten ist ein wichtiger Schritt in der semantischen Segmentierung, um die Qualität und Genauigkeit der Segmentierungsergebnisse zu verbessern. In diesem Kapitel werden verschiedene Vorverarbeitungsschritte wie Filterung, Normalenberechnung und Downsampling betrachtet, die oft in der Praxis angewendet werden.

Zu Beginn wird häufig eine Filterung des Datensatzes durchgeführt. Dadurch kann unerwünschtes Rauschen im Datensatz unterdrückt und dessen Qualität verbessert werden. Dies kann durch verschiedene Filtertechniken erfolgen, wie zum Beispiel durch Medianfilter, Gaußsche Filter [11] oder region growing und Bilateralfilter [12]. Durch die Filterung und die damit einhergehende Reduzierung des Rauschens kann die Qualität der Segmentierungsergebnisse verbessert werden.

Downsampling: Beim Prozess des Downsamplings wird die Anzahl an Punkten innerhalb einer Punktwolke reduziert, um die Verarbeitungsgeschwindigkeit und den Speicherbedarf zu reduzieren. Dies kann durch verschiedene Verfahren erfolgen. Eine sehr schnelle und recheneffiziente Methode ist hierbei das sogenannte Random-Downsampling. Dabei wird ein bestehender Datensatz um eine absolute oder relative Punktanzahl reduziert. Die wegfallenden Punkte werden dabei zufällig ausgewählt. Nachteil ist hierbei, dass sich ursprünglich ungleichmäßig abgetastete Bereich des Datensatzes noch vergrößern können, und dadurch markante Merkmale in der Punktwolke verloren gehen können [13]. Deshalb wird stattdessen häufig auf das Voxel-Grid-Downsampling zurückgegriffen. Dabei wird der Raum, in der sich die Punktwolke befindet, in gleichmäßige Voxel unterteilt. Daraufhin wird in jedem Voxel nur eine bestimmte Anzahl an Punkten übernommen, die anderen werden aus dem Datensatz entfernt. Welche Punkte übernommen werden kann dabei durch verschiedene Verfahren bestimmt werden. Danach wird eine bestimmte Anzahl an Punkten, etwa durch ihre nähe zum Mittelpunkt, ausgewählt und in das Voxel-Grid übernommen. Ziel des Voxel-Grid-Downsamplings ist es, die Datenmenge zu reduzie-

ren, während wichtige strukturelle und semantische Informationen erhalten bleiben.

Normalenberechnung: Die Berechnung von Normalen ist ein optionaler Schritt, um die geometrische Information der 3D-Daten zu erfassen. Dabei wird für jeden Punkt oder Voxel des Datensatzes ein Normalenvektor bestimmt. Normalen sind Vektoren, die senkrecht zur Oberfläche von Objekten in der Umgebung stehen und gibt so Aufschluss über die Orientierung der Oberflächen der Szene. SCHÄTZVERAHReN

3.3 Datenannotation und Ground Truth-Erstellung

Die Erstellung annotierter Daten und eines Ground Truths sind entscheidende Schritte bei der semantischen Segmentierung von 3D-Daten. Da die semantische Segmentierung auf Erweiterungen von neuronale Netze basiert, werden annotierte Datensätze benötigt um diese zu trainieren. Dabei erfolgt eine manuelle Klassifikation der Daten, indem semantische Labels oder Klasseninformationen den 3D-Daten zugeordnet werden. Die Qualität und Genauigkeit der Datenannotation sind von entscheidender Bedeutung für die Leistungsfähigkeit der Modelle für die semantische Segmentierung.

Die Erstellung des Ground Truths umfasst dabei die Erzeugung von referenzbasierten Segmentierungsergebnissen, die als Grundlage für das Training und die Evaluation von Segmentierungsmodellen dienen. Der Ground Truth kann sowohl manuell als auch automatisch erstellt werden, um die Zuverlässigkeit und Vergleichbarkeit der Segmentierungsergebnisse sicherzustellen und die Qualität der trainierten Modelle zu überprüfen. Dafür wird häufig der sogenannte IoU (Intersection over Union) verwendet. Dabei wird die Anteil der überlappenden Bereiche zwischen Ground Truth und Segmentierungsergebnis berechnet. Je höher der Wert, desto besser ist das Segmentierungsergebnis.

4 Grundlegende Verfahren der semantischen Segmentierung

Ziel der Semantischen Segmentierung von 3D-Daten ist es, jedem Punkt im Raum einer bestimmten Kategorie zuzuordnen und dadurch Bereiche des Bildes in klassifizierte Objekte zu unterteilen. Hierfür gibt es verschiedene Ansätze, die auf erweiterten neuronalen Netzen basieren.

4.0.1 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) sind eine erweiterte Art von künstlichen neuronalen Netzen, die speziell für die Verarbeitung von Bildern entwickelt wurden. Sie bestehen aus mehreren Schichten, darunter Convolutional Layers, Pooling Layers und Fully Connected Layers, die miteinander verbunden sind. Die Architektur der CNNs ist dabei nicht vorgegeben, folgt aber in der Praxis immer einer ähnlichen Vorlage. Ein Input-Layer beinhaltet die Pixelwerte des Bildes. Darauf folgen in der Regel ein oder zwei Convolutional-Layers, woraufhin sich ein Pooling-Layer anreiht. Die Kombination aus Convolutional- und Pooling-Layer kann dabei je nach Komplexität beliebig oft im Netz vorkommen. Am Ende folgt ein Fully-Connected-Layer an den der Output anknüpft. Dieser gibt das Klassifizierungsergebnis des betrachteten Bildes durch das Netzwerk aus. Die Convolutional-Layers können vereinfacht als Filter-Layer betrachtet werden und extrahieren markante Merkmale aus den Eingabebildern, um die Klassifikation zu erleichtern. Dabei wird jeder Bereich des Bildes pro Layer mit einem Kernel gefaltet und erzeugen eine 2D-Aktivierungskarte. Die kernels besitzen dabei häufig kleine räumliche Dimensionalitäten, erstrecken sich aber über die Gesamte Tiefe des Eingangsbildes. Die Anschließend folgenden Pooling-Layers dienen dazu, die Dimensionen der Ausgabe des Convolutional-Layers zu reduzieren und somit die Rechnenkomplexität des Modells zu verringern. Besonders häufig

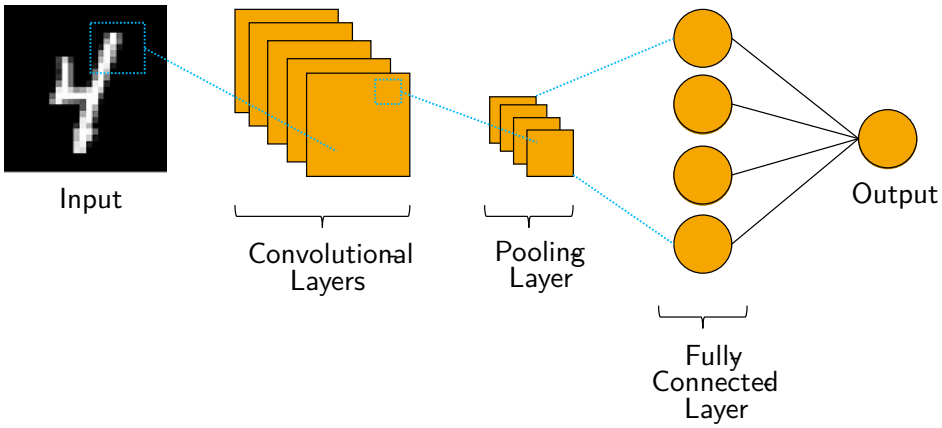

Abbildung 4.1: Aufbau eines Convolutional Neural Networks

kommen dabei sogenannte Max-Pooling-Layers zum Einsatz. Bei diesen wird ein Kernel einer beliebigen Größe, ohne zu überlappen, über die Aktivierungskarte geschoben. Dabei wird nur der größte Aktivierungswert eines Fensterbereiches übernommen und so die Dimensionalität der Aktivierungskarte stark reduziert. Die Fully Connected Layers am Ende des Netzes verarbeiten schließlich die extrahierten Aktivierungen und versuchen daraus Klassifizierungsergebnisse zu gewinnen. Der Hauptanwendungsbereich von CNNs liegt dabei in der Klassifikation von Bildern in vorbestimmte Kategorien. [14].

4.0.2 Fully Convolutional Networks (FCNs)

Fully Convolutional Networks (FCNs) sind eine Weiterentwicklung von CNNs, die speziell für die Aufgabe der semantischen Segmentierung von Bildern entwickelt wurden. Im Gegensatz zu herkömmlichen CNNs, die für die Klassifizierung und Erkennung von Objekten in Bildern ausgelegt sind, können FCNs jedes Pixel eines Eingabebildes klassifizieren und somit die räumlichen Informationen der klassifizierten Bilder beibehalten. FCNs verwenden dabei ausschließlich Convolutional-Layers und Pooling-Layers, nicht aber Fully-Connected-Layers. Dies ermöglicht es eine Merkmalskarte des Eingabebildes zu erzeugen, auf der jedes Pixel einer bestimmten Klasse zugeordnet wird. Durch die Verwendung von FCNs können somit komplizierte Zusammenhänge innerhalb von Bildern auf der Ebene der Pixel identifiziert werden, was für Anwendungen wie die autonome Navigation oder Objekterkennung von großer Bedeutung ist. [15] Bekannte und erfolgreiche Verfahren wie das U-Net, verwenden dabei häufig eine Encode-Decoder-Architektur,

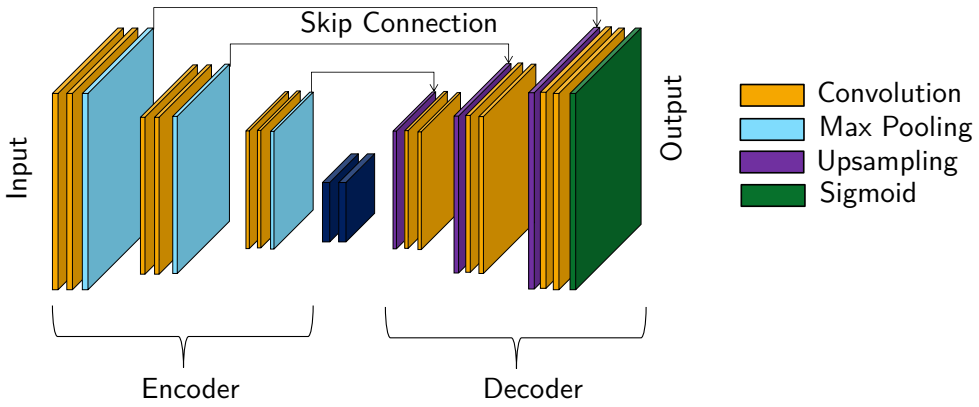

Abbildung 4.2: Encoder Decoder Architektur ähnlich U-Net

um das Klassifizierungsergebnis in der Dimensionalität des Eingangsbildes darzustellen.

4.0.3 Encoder-Decoder-Architekturen

Encoder-Decoder-Architekturen stellen eine spezielle Architektur von neuronalen Netzen dar. Der Name leitet sich aus deren Aufbau ab, welcher aus zwei Hauptkomponenten, einem Encoder und einem Decoder, besteht. Der Encoder verwendet typischerweise Convolutional und Pooling-Layers, um das Eingabebild schrittweise in eine kompakte, abstrakte Repräsentation zu komprimieren, die die Merkmale des Eingabebildes stark reduziert enthält. Dabei werden die Positionen der maximalen Aktivierungen der Ebene während des Max-Pooling Prozesses gespeichert und als Pooling-Indizes bezeichnet. Der Decoder verwendet häufig Deconvolutional-Neuronale-Networks, um das Ergebnis des Encoder-Netzwerkes wieder in die Dimensionalitäten des Eingangsbildes zurückzuführen. Das Upsampling erfolgt dabei nicht linear, sondern verwendet die Pooling-Indizes des zugehörigen Encoding-Schrittes, um die Aktivierungen an der richtigen Position des Bildes wiederherzustellen. Dies geschieht über sogenannte Skip-Verbindungen. Die Semantische Segmentierung erfolgt dabei am Ende des Decoding Netzwerkes. Dabei wird die Merkmalskarte des Decoders in eine Wahrscheinlichkeitsverteilung bezüglich der zu klassifizierenden Klassen umgewandelt. Die geschieht häufig über die Softmax-Funktion. [15]

4.0.4 Region-based Convolutional Neural Networks (R-CNNs)

Region-based Convolutional Neural Networks (R-CNNs) sind eine Weiterentwicklung von Convolutional Neural Networks. Im Gegensatz zu herkömmlichen CNNs, die eine feste Größe der

Eingabebilder erfordern, verwenden R-CNNs eine Region Proposal Technik, um Regions of Interest (ROI) innerhalb des Bildes zu detektieren. Anschließend wird nur auf die ausgewählten Bereiche eine Klassifizierung durchgeführt. R-CNNs erzielen dadurch eine höhere Genauigkeit als herkömmliche CNNs bei der Erkennung von Objekten in Bildern und werden daher häufig in der Robotik und im autonomen Fahren eingesetzt. Dies liegt daran, dass sie weniger Rechenleistung benötigen und gleichzeitig Störeinflüsse, die außerhalb der ROI liegen, keinen negativen Einfluss auf die Klassifizierung nehmen können. [**8237548**]

5 Anwendungszenarien der semantischen Segmentierung

5.1 Autonomes Fahren

Eines der wichtigsten Einsatzgebiete der semantischen Segmentierung ist der Bereich des autonomen Fahrens. Hierbei ist eine präzise und zuverlässige Wahrnehmung der Umgebung unumgänglich. Dabei ist eine der größten Herausforderungen die Verarbeitung der großen Datenmengen, die von den Sensoren erfasst werden. Die Verarbeitungsgeschwindigkeit muss dabei extrem hoch sein, um in Echtzeit auf die Umgebung reagieren zu können. Durch die semantische Segmentierung ist es dem Fahrzeug möglich, wichtige Informationen über Straßenverhältnisse, Verkehrszeichen, Fußgänger, Fahrzeuge und andere Hindernisse zu sammeln. Mithilfe einer präzise Klassifizierung jedes Pixels im Bild ist es dem Fahrzeug möglich, Objekte in der Umgebung kontextbezogen zu erkennen und situationsgerecht zu reagieren.

5.2 Robotik in der Industrie

Die Semantische Segmentierung spielt eine wichtige Rolle in der Industrierobotik, insbesondere wenn es darum geht, stark variierende Produkte bezüglich ihres Formfaktors zu handhaben. Eine Herausforderung besteht darin, die Produkte zu erkennen und zu klassifizieren. Dies ermöglicht es dem Roboter zwischen unterschiedlichen Teilen zu unterscheiden und bauteilspezifisch zu reagieren. So kann beispielsweise eine optimale Griffposition des Bauteils ermittelt werden. Auch eine Manipulation komplexer Objekte kann dadurch ermöglicht werden.

5.3 Augmented Reality

Einsatzgebiete der Semantischen Segmentierung im Bereich von Augmented Reality (AR) sind vielfältig. Diese kann beispielsweise in der industriellen Fertigung eingesetzt werden, um komplexe Montage- und Wartungsprozesse zu unterstützen. Durch semantische Segmentierung können AR-Systeme bereits platzierte Teile und Komponenten identifizieren, Fehler erkennen und den Arbeitern visuelle Montageanleitungen in Echtzeit bereitstellen. Dies erhöht die Effizienz, Präzision und Sicherheit bei der Durchführung von Aufgaben. In der Architektur und im Bauwesen ermöglicht die semantische Segmentierung in Kombination mit AR eine Visualisierung von Gebäuden und Infrastrukturen. Architekten, Designer und Ingenieure können virtuelle Modelle in die reale Umgebung einfügen, um Entwürfe auf Kollisionen zu überprüfen. Die semantische Segmentierung ist dabei eine entscheidende Komponente, um die Umgebung zu analysieren, zu verstehen und interaktiv nutzbar zu machen. Dies ermöglicht es AR-Systemen, die Umgebung zu analysieren und virtuelle Inhalte realitätsnah zu platzieren und mit diesen zu interagieren.

5.4 Landwirtschaft

Auch in der Landwirtschaft hat die Semantische Segmentierung große Potentiale. Durch die präzise Klassifizierung von verschiedenen Pflanzenarten und Strukturen auf landwirtschaftlichen Flächen, können Landwirte wichtige Informationen gewinnen, um ihre Produktivität und Effizienz zu steigern. Mit Hilfe der semantischen Segmentierung können Unkraut und Nutzpflanzen unterschieden werden. Dies ermöglicht eine gezielte Anwendung von Düngemitteln und Pestiziden, sowie einer effizienten Bewässerung. Darüber hinaus kann die semantische Segmentierung auch bei der Erkennung von Schädlingen und Krankheiten helfen, um rechtzeitig Gegenmaßnahmen zu ergreifen. Durch die genaue Analyse und Segmentierung landwirtschaftlicher Flächen bietet die semantische Segmentierung ein wertvolles Werkzeug, um die Landwirtschaft nachhaltiger, produktiver und umweltfreundlicher zu gestalten. BILD

5.5 Medizin

Semantische Segmentierung spielt eine entscheidende Rolle im Bereich der Medizintechnik und hat das Potenzial, unterstützend bei medizinische Diagnosen und Behandlungen zu wirken. Durch

die präzise Segmentierung von anatomischen Strukturen und Geweberegionen in medizinischen Bildern wie CT-Scans oder MRT-Aufnahmen können Ärzte einen besseren Einblick in die Anatomie des Patienten erhalten. Beispielsweise können Organe für einen besseren Überblick automatisch eingefärbt werden, oder verdächtige Stellen in der Aufnahme markiert werden. So kann der untersuchende Arzt beispielsweise Verletzungen oder potentielle Tumore gezielt betrachten. Diese detaillierte Analyse ermöglicht eine präzisere Diagnosestellung und kann bei der Planung und Durchführung von chirurgischen Eingriffen, sowie der Überwachung des Fortschreitens von Krankheiten eingesetzt werden. Die semantische Segmentierung erleichtert auch die Erstellung von 3D-Modellen anatomischer Strukturen, die für die präoperative Planung und Simulation von komplexen Operationen verwendet werden können. Dabei können beispielsweise markante Stellen einer CT-Aufnahme bei einer endoskopischen Operation wieder erkannt werden und dem operierenden Arzt unterstützend eingeblendet werden. Dies kann unter anderem auch mithilfe von AR geschehen. BILD

6 State-of-the-Art Verfahren zur semantischen Segmentierung von 3D-Daten

6.0.1 PointNet

PointNet++ ist eine fortschrittliche Methode zur Verarbeitung von Punktwolken in der 3D-Bildverarbeitung, die speziell für die Aufgaben der Klassifikation, Segmentierung und Erkennung von Objekten entwickelt wurde. Es basiert auf der PointNet-Architektur und erweitert sie durch die Integration einer hierarchischen Struktur, um lokale und globale Kontextinformationen in Punktwolken zu erfassen. Das Verfahren arbeitet dabei direkt auf den Punktwolken-Daten, ohne diese in ein Voxelgitter oder eine andere Form zu überführen. Dadurch wird eine höhere Genauigkeit und Effizienz erreicht.

PointNet++ verwendet eine hierarchische Struktur, um die räumlichen Informationen der 3D-Daten effektiv zu erfassen. Es besteht aus einer Serie von PointNet-Modulen, die auf verschiedenen Ebenen der Punktwolke arbeiten. Jedes PointNet-Modul nimmt eine Teilmenge von Punkten als Eingabe und extrahiert lokale Merkmale durch mehrere Convolutional- und Pooling-Schichten. Durch die hierarchische Anordnung der Module können sowohl lokale als auch globale Merkmale erfasst werden, um eine genaue Segmentierung zu ermöglichen.

Ein wichtiger Schritt in der Funktionsweise von PointNet++ ist die Anwendung von Farb- oder Normalinformationen, um zusätzliche Merkmale zu extrahieren. Dadurch können beispielsweise Oberflächenmerkmale oder Orientierungsinformationen berücksichtigt werden, was zu einer verbesserten Segmentierungsgenauigkeit führt.

6.0.2 3D U-Net

3D U-Net ist ein leistungsstarkes Verfahren für die semantische Segmentierung von 3D-Daten. Es basiert auf der verarbeiten 2D U-Net-Architektur, die für die Bildsegmentierung entwickelt wurde. Um diese Architektur auf 3D-Daten anzuwenden, wurde sie entsprechend angepasst.

Die 3D U-Net-Architektur besteht aus einem Encoder-Decoder-Netzwerk mit Skip-Connections. Der Encoder-Teil nimmt das Eingabevolumen entgegen und besteht aus mehreren Convolutional-Layers, gefolgt von Pooling-Layern. Diese Schichten helfen dabei, wichtige Merkmale des Eingangsbildes zu extrahieren und die räumliche Auflösung zu reduzieren. Die Skip Connections werden zwischen den Encoder- und Decoder-Schichten erstellt, um beim Prozess des upsamplings eine positionsgetreue Wiederherstellung der Bildmerkmale zu gewährleisten. Der Decoder-Teil des 3D U-Net-Netzwerks besteht dabei aus Upsampling-Schichten, die die räumliche Auflösung erhöhen, gefolgt von Deconvolutional-Layers.

Ein wesentliches Merkmal von 3D U-Net ist seine Fähigkeit, volumetrische Kontextinformationen zu berücksichtigen. Durch die Verarbeitung von 3D-Volumendaten können komplexe räumliche Zusammenhänge erfasst und genutzt werden, um eine präzise Segmentierung zu erreichen. Dies ist insbesondere in medizinischen Anwendungen von großer Bedeutung, in denen die genaue Abgrenzung von Organen oder Tumoren entscheidend ist.

7 Herausforderungen und zukünftige Entwicklungen

7.1 Herausforderungen und Limitationen

Die semantische Segmentierung ist eine leistungsstarke Technik mit breitem Anwendungspotenzial, besitzt jedoch noch einige Herausforderungen und Limitationen. Eine der Hauptproblematiken besteht jedoch in der Notwendigkeit großer Datensätze für das Training von Segmentierungsmodellen. Das manuelle Labeln solcher Datensätze ist dabei sehr zeitaufwändig. Zudem kann es schwierig sein, einen ausreichend großen Datensatz für selten vorkommende Klassen oder bestimmte Anwendungsbereiche zu sammeln. Ein weiteres Problem ist die Verarbeitungsgeschwindigkeit bei der Echtzeitsegmentierung. Der Einsatz von komplexen Modellen und hochauflösenden Kamerasystemen erfordert leistungsstarke Hardware und effiziente Algorithmen, um die erforderliche Echtzeitverarbeitung zu gewährleisten. Eine weitere Limitation besteht in der Anfälligkeit gegenüber Variationen in Beleuchtung, Umwelteinflüssen, abweichenden Blickwinkeln und schwankender Bildqualität. Darüber hinaus kann die semantische Segmentierung in Bereichen mit starken Objektüberlappungen oder ähnlichen Texturen Schwierigkeiten haben, klare Grenzen zwischen den Objekten zu erkennen und korrekt zu segmentieren. Die größte Herausforderung im Bereich der semantischen Segmentierung bestehen darin, die Geschwindigkeit und Genauigkeit der Segmentierung zu verbessern.

7.2 Potenziale und Trends für zukünftige Entwicklungen

Der Einsatz von Deep-Learning Modellen, insbesondere in Kombination mit CNNs, hat in den letzten Jahren bereits für große Fortschritte im Bereich der semantischen Segmentierung gesorgt.

In der Zukunft können leistungsfähigere und effizientere Modelle erwartete werden, mithilfe deren die Segmentierungsgenauigkeit weiter ansteigen sollte. Dadurch wird neben der Verarbeitungsgeschwindigkeit der Modelle auch die Robustheit gegenüber gegenüber Störeinflussen steigen. Durch die steigende Geschwindigkeit der Modelle, könnten auch komplexere 3D-Datensätze für die Segmentierung verwendet werden, was zu präziseren Ergebnissen führen könnte.

Ein weiterer Bereich, der an Bedeutung gewinnt, ist die Echtzeitsegmentierung. Mit der steigenden Verfügbarkeit von leistungsstarken GPUs und der Optimierung von Deep-learning Modellen wird es möglich sein, semantische Segmentierung in Echtzeit auf hochauflösenden Bildern oder sogar in Echtzeit-Videoströmen durchzuführen. Dies eröffnet neue Anwendungsbereiche in Bereichen wie autonomes Fahren, Robotik, Augmented Reality und Überwachungssystemen.

Neben diesen technischen Trends wird die zunehmende Verfügbarkeit großer annotierter Datensätze und die Verbesserung der Labeling-Technologien voraussichtlich zu weiteren Fortschritten in der semantischen Segmentierung führen. Mehr Daten ermöglichen es, Modelle auf breiteren und vielfältigeren Datensätzen zu trainieren, was die Generalisierung und Anpassungsfähigkeit verbessert.

8 Zusammenfassung und Anwendung

In dieser Arbeit wurden die Grundlagen der semantischen Segmentierung erläutert. Hierfür gibt es verschiedene Verfahren, die auf erweiterten neuronalen Netzen basieren. Zu diesen Verfahren gehören Convolutional Neural Networks (CNNs), Fully Convolutional Networks (FCNs), Region-based Convolutional Neural Networks (R-CNNs) und Encoder-Decoder-Architekturen. CNNs sind speziell für die Verarbeitung von Bildern konzipiert und werden für die Klassifikation von Bildern in vorbestimmte Kategorien eingesetzt. FCNs wurden speziell für die semantische Segmentierung von Bildern entwickelt und können die Pixel jedes Eingabebildes direkt klassifizieren, wodurch die räumliche Information beibehalten wird. R-CNNs wurden für die Objekterkennung in Bildern entwickelt und verwenden eine Region Proposal Technik, um Regions of Interest (ROI) innerhalb des Bildes zu detektieren. Encoder-Decoder-Architekturen bestehen aus einem Encoder, der das Eingabebild in eine kompakte, abstrakte Repräsentation umwandelt, und einem Decoder, der aus dieser Repräsentation eine Semantikkarte erzeugt.

Literatur

[1] Kaihong Yang, Sheng Bi und Min Dong. „Lightningnet: Fast and Accurate Semantic Segmentation for Autonomous Driving Based on 3D LIDAR Point Cloud“. In: *2020 IEEE International Conference on Multimedia and Expo (ICME)*. 2020, S. 1–6. DOI: 10.1109/ICME46284.2020.9102769.

[2] Mohammad Hosein Hamian u. a. „Semantic Segmentation of Autonomous Driving Images by the Combination of Deep Learning and Classical Segmentation“. In: *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*. 2021, S. 1–6. DOI: 10.1109/CSICC52343.2021.9420573.

[3] Tuan Pham. „Semantic Road Segmentation using Deep Learning“. In: *2020 Applying New Technology in Green Buildings (ATiGB)*. 2021, S. 45–48. DOI: 10.1109/ATiGB50996.2021.9423307.

[4] Liuhao Ge u. a. „3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images“. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, S. 5679–5688. DOI: 10.1109/CVPR.2017.602.

[5] Jingyun Liu u. a. „TOF Lidar Development in Autonomous Vehicle“. In: *2018 IEEE 3rd Optoelectronics Global Conference (OGC)*. 2018, S. 185–190. DOI: 10.1109/OGC.2018.8529992.

[6] Emre DANDIL und Kerim Kürşat ÇEVİK. „Computer Vision Based Distance Measurement System using Stereo Camera View“. In: *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*. 2019, S. 1–4. DOI: 10.1109/ISMSIT.2019.8932817.

[7] Yosef Dalbah, Stephan Rohr und Friedrich M. Wahl. „Detection of dynamic objects for environment mapping by time-of-flight cameras“. In: *2014 IEEE International Conference on Image Processing (ICIP)*. 2014, S. 971–975. DOI: 10.1109/ICIP.2014.7025195.

[8] José Gomes da Silva Neto u. a. „Comparison of RGB-D sensors for 3D reconstruction“. In: *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*. 2020, S. 252–261. DOI: 10.1109/SVR51698.2020.00046.

[9] Inzamam Anwar und Sukhan Lee. „High performance stand-alone structured light 3D camera for smart manipulators“. In: *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. 2017, S. 192–195. DOI: 10.1109/URAI.2017.7992709.

[10] Yin Zhou und Oncel Tuzel. „VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection“. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, S. 4490–4499. DOI: 10.1109/CVPR.2018.00472.

[11] Karl Thurnhofer-Hemsi u. a. „Super-Resolution of 3D MRI Corrupted by Heavy Noise With the Median Filter Transform“. In: *2020 IEEE International Conference on Image Processing (ICIP)*. 2020, S. 3015–3019. DOI: 10.1109/ICIP40778.2020.9191237.

[12] Li Chen, Hui Lin und Shutao Li. „Depth image enhancement for Kinect using region growing and bilateral filter“. In: *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. 2012, S. 3070–3073.

[13] Bingjie Liu u. a. „Tree Species Classification of Backpack Laser Scanning Data Using the PointNet++ Point Cloud Deep Learning Method“. In: *Remote Sensing* 14.15 (2022), S. 3809. ISSN: 2072-4292. DOI: 10.3390/rs14153809. URL: https://www.mdpi.com/2072-4292/14/15/3809.

[14] Keiron O'Shea und Ryan Nash. *An Introduction to Convolutional Neural Networks*. URL: https://arxiv.org/pdf/1511.08458.pdf.

[15] Vijay Badrinarayanan, Alex Kendall und Roberto Cipolla. „SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (2017), S. 2481–2495. DOI: 10.1109/TPAMI.2016.2644615.