Technische Hochschule Ingolstadt
Fakultät Elektro- und Informationstechnik

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 hat die Forschung im Bereich der autonomen Fahrzeuge und der Robotik enorme Fortschritte gemacht. Ein wichtiger Faktor für die Entwicklung dieser Technologien ist die Fähigkeit, die Umgebung ausreichend genau zu erkennen und zu verstehen. In diesem Zusammenhang hat die semantische Segmentierung der Umgebung auf Basis von 3D-Daten eine immer größere Bedeutung erlangt. Die semantische Segmentierung ist ein Verfahren zur automatischen Klassifizierung von Objekten und Strukturen in der Umgebung. Dabei werden jedem Pixel oder jedem Voxel in einem 3D-Modell eine bestimmte semantische Bedeutung zugeordnet, z.B. Straße, Gebäude, Bäume oder Fahrzeuge. Eine präzise und schnelle semantische Segmentierung ist eine wesentliche Voraussetzung für eine zuverlässige Navigation von mobilen Plattformen, wie autonomen Fahrzeugen oder Robotersystemen [1]. In dieser Arbeit wird die semantische Segmentierung der Umgebung auf Basis von 3D-Daten untersucht. Dabei sollen verschiedene Methoden und Ansätze für die semantischen Segmentierung, sowie bestehende Probleme dargestellt und bewertet werden.

1.2 Problemstellung und aktueller Stand

Trotz der Fortschritte im Bereich der Computer Vision gibt es noch immer einige Herausforderungen zu überwinden. Eines der Probleme besteht in der Komplexität der Umgebung. Ein 3D-Umfeld kann durch eine Vielzahl von verschiedenen Objekten und Strukturen, die miteinander interagieren und sich gegenseitig beeinflussen, besonders herausfordernd sein. 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 hohen Verarbeitungsgeschwindigkeit. Die Verarbeitung von großen Datenmengen erfordert eine erhebliche Rechenleistung, um eine schnelle und präzise Segmentierung der Umgebung zu ermöglichen. Dies kann für viele Anwendungen, insbesondere für sich schnell bewegende mobile Geräte, eine Herausforderung darstellen [2].

Hinzu kommt die begrenzte Genauigkeit der Segmentierungsverfahren. 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. Eine Vielzahl aktueller Entwicklungen beschäftigt sich mit der Verbesserung der Algorithmen. Dabei gelten besonders Deep-learning Methoden und Convolutional Neural Networks (CNNs) als vielversprechende Ansätze, um die Komplexität der Umgebung besser zu erfassen [3, 4].

2 Sensoren zur Erfassung von 3D-Daten

2.1 LiDAR-Sensoren

LiDAR-Sensoren, die auch unter dem Namen Light Detection and Ranging-Sensoren bekannt sind, stellen eine weit verbreitete Technologie zur Erfassung von 3D-Daten dar. Sie basieren auf dem Einsatz von Laserstrahlen, welche ausgesendet werden und von Objekten in der Umgebung reflektiert werden. Dabei kann zwischen Time of Flight (TOF) LiDAR und phasenbasiertem LiDAR unterschieden werden. Während TOF-LiDAR die Distanz über eine Messung der Laufzeit der lichtwelle bestimmt, erfolgt die Entfernungsmessung beim phasenbasierten LiDAR über die Auswertung der Phasenverschiebung der vom Objekt reflektierten Lichtwelle. Hierdurch können LiDAR-Sensoren hochgenaue Entfernungen zu den reflektierenden Objekten erfassen aus denen sich detaillierte 3D-Punktwolken erzeugen lassen, 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. [5]

2.2 Tiefenkameras

Tiefenkameras basieren auf verschiedene Verfahren, um Entfernungen zu messen. 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 diesem lassen sich Tiefeninformationen des betrachteten Objektes berechnen [6]. Bei Time of Flight Kameras wird ein moduliertes Lichtsignal im Infrarotbereich ausgesendet und von Objekten in der Umgebung reflektiert. Über die Phasenverschiebung der Infrarotwelle lässt sich die Entfernung des Objektes zur Kamera berechnen [7]. 3D-Kameras auf Basis von Structured Light projizieren ein spezielles 2D-Muster auf das zu betrachtende Objekt. Aus der Verzerrung dessen, lassen sich Tiefeninformationen berechnen [8]. Die meisten Tiefenkameras stellen dabei die Tiefeninformationen in einem Bild aus Graustufen dar. Zusätzlich gibt es auch RGB-D Kameras, welche zusätzlich zu einer Structured Light oder TOF-Kamera über eine RGB-Kamera verfügen. Diese haben häufig eine höhere räumliche Auflösung und sind in der Lage zusätzlich Farbinformationen aufzunehmen, besitzen jedoch einen deutlich kleineren Arbeitsbereich [9].

2.3 Passive und aktive Sensoren

Grundsätzlich lassen sich Sensoren zur Gewinnnung von 3D-Daten in zwei Klassen unterscheiden. Aktive Sensoren wie LIDAR-Sensoren senden selbst Energie in Form von Laser- oder Lichtwellen aus, um Informationen über das Objekt zu sammeln. Die reflektierten Signale werden von der Sensor-Einheit aufgenommen und zur Berechnung von Tiefeninformationen verwendet. Im Gegensatz dazu erfordern passive Sensoren wie Stereokameras keine aktive Energiequelle, sondern nutzen 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

ie Wahl der geeigneten Sensoren für die semantische Segmentierung hängt von verschiedenen Faktoren ab, wie den Anforderungen der Anwendung, den Umgebungsbedingungen, dem Budget und den gewünschten Ergebnissen. Aspekte wie die benötigte Genauigkeit, räumliche Auflösung, Reichweite, Echtzeitfähigkeit und Umgebungsbedingungen sollten bei der Auswahl von Sensoren berücksichtigt werden. Zum Beispiel benötigen Anwendungen im Bereich der autonomen Fahrzeuge möglicherweise Sensoren mit hoher Reichweite und Genauigkeit, während Anwendungen im Innenbereich möglicherweise Sensoren mit höherer räumlicher Auflösung und Echtzeitfähigkeit benötigen. Die Umgebungsbedingungen, wie schlechte Beleuchtungsbedingungen oder komplexe Geometrien, können ebenfalls die Leistung von Sensoren beeinflussen und die Wahl von geeigneten Sensoren beeinflussen. Das Budget ist ebenfalls ein wichtiger Faktor bei der Auswahl von Sensoren, da verschiedene Sensoren unterschiedliche Kosten haben können. Schließlich sollten auch die gewünschten Ergebnisse der semantischen Segmentierung berücksichtigt werden, da verschiedene Sensoren besser geeignet sein können, um bestimmte Objekte oder Strukturen in der Umgebung zu segmentieren. Zum Beispiel können Lidar-Sensoren aufgrund ihrer präzisen Tiefeninformationen und Reichweite gut geeignet sein, um Objekte wie Straßen, Gebäude oder Bäume zu segmentieren, während Kameras oder Tiefenkameras besser für die Segmentierung von Fußgängern oder Fahrzeugen geeignet sein können.

3 Datengrundlage und Vorverarbeitung

3.1 3D-Datenformate und Datentypen

Bei der semantischen Segmentierung von 3D-Daten spielen die zugrunde liegenden 3D-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 häufig von LiDAR-Sensoren erzeugt wird. Punktewolken sind Sammlungen von 3D-Punkten, die die Oberfläche von Objekten in der Umgebung darstellen. Sie können in verschiedenen Dateiformaten gespeichert werden, wie beispielsweise dem ASCII-Format oder dem binären LAS-Format (LASer File Format), das speziell für LiDAR-Daten 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. Im Vergleich zu Bildern weißen Punktwolken 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 [10]. Das Punktwolkenformat erzeugt dabei große Datenmengen, was zu einer rechenintensiven Verarbeitung der Daten führen kann.

Neben Punktewolken werden auch 3D-Gitter oder Voxel-Daten oft 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. Voxel-Daten können in verschiedenen Formaten gespeichert werden, wie zum Beispiel das binäre OctoMap-Format oder dem ASCII-Format. Sie bieten 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.

Filterung von 3D-Daten: Die Filterung von 3D-Daten beinhaltet das Entfernen von unerwünschtem Rauschen oder Ausreißern, um die Qualität der Daten zu verbessern. Dies kann durch verschiedene Filtertechniken erfolgen, wie zum Beispiel Medianfilter, Gaußscher Filter [11] oder region growing und Bilateralfilter [12]. Diese Filter können angewendet werden, um Rau-

schen oder Ausreißer in den 3D-Punktwolken oder Voxel-Daten zu reduzieren und somit die Qualität der Daten für die semantische Segmentierung zu verbessern.

Normalenberechnung: Die Berechnung von Normalen ist ein wichtiger Schritt, um die geometrische Information der 3D-Daten zu erfassen. Normalen sind Vektoren, die senkrecht zur Oberfläche von Objekten in der Umgebung stehen und die Orientierung der Oberfläche angeben. Die Normalenberechnung kann auf Basis von Punktwolken durchgeführt werden und ermöglicht es, die Richtung und Orientierung der Objektoberflächen zu erfassen, was für die semantische Segmentierung von Objekten von Bedeutung ist.

Downsampling: Das Downsampling von 3D-Daten beinhaltet die Reduzierung der Datenmenge, um die Verarbeitungsgeschwindigkeit und den Speicherbedarf zu reduzieren. Dies kann durch verschiedene Techniken erfolgen. Häufig erfolgt dies in Form von Voxel-Grid-Downsampling. Dabei wird der Raum, in der sich die Punktwolke befindet, in gleichmäßige Voxel unterteilt, wobei für jedes Voxel der Schwerpunkt der darin enthaltenen Punkte berechnet wird. Danach wird eine bestimmte Anzahl an Punkten, häufig durch ihre nähe zum Schwerpunkt, ausgewählt und in das Voxel-Grid übernommen. Ziel des Downsamplings ist es, die Datenmenge zu reduzieren, während wichtige strukturelle und semantische Informationen erhalten bleiben.

4 Grundlagen der semantischen Segmentierung

4.1 Verfahren zur 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.

4.1.1 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) sind eine 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.[GRAFIK?] Die Convolutional Layers können vereinfacht als Filter-Layer betrachtet werden und extrahieren Merkmale aus den Eingabebildern. Dabei wird jeder Bereich des Bildes mit Kernels (Filtern) gefalten und erzeugen eine 2D-Aktivierungskarte. Die kernels besitzen dabei häufig kleine räumliche Dimensionalitäten, erstrecken sich aber über die Gesamte Tiefe des Eingangsbildes. Pooling Layers dienen dazu, die Dimensionen der Ausgabe des Convolutional-Layers zu reduzieren und somit die Rechnenkomplexität des Modells zu verringern. 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. [13].

4.1.2 Fully Convolutional Networks (FCNs)

Fully Convolutional Networks (FCNs) sind eine Weiterentwicklung von Convolutional Neural Networks (CNNs), die speziell für die Aufgabe der semantischen Segmentierung von Bildern entwickelt wurden. Im Gegensatz zu herkömmlichen CNNs, die für die Klassifizierung von Bildern ausgelegt sind, können FCNs die Pixel jedes Eingabebildes direkt klassifizieren und damit die räumliche Information 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. [14]

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

Region-based Convolutional Neural Networks (R-CNNs) sind eine Weiterentwicklung von Convolutional Neural Networks, die speziell für die Aufgabe der Objekterkennung in Bildern entwickelt wurden. 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 auf die ausgewählten Bereiche ein FCN angewendet, um diesen auf Pixelebene zu klassifizieren. 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. [15]

4.1.4 Encoder-Decoder-Architekturen:

Encoder-Decoder-Architekturen sind eine spezielle Art von neuronalen Netzen, die zur semantischen Segmentierung von Bildern eingesetzt werden. Der Name leitet sich von der Architektur ab, die aus zwei Hauptkomponenten besteht: dem Encoder und dem Decoder. Der Encoder besteht aus mehreren Faltungsschichten, die das Eingabebild schrittweise in eine kompakte, abstrakte Repräsentation komprimieren, die die Merkmale des Bildes stark reduziert enthält. Der Decoder besteht aus mehreren Deconvolution-Schichten, die diese kompakte Repräsentation schrittweise wieder in eine vollständige Bildgröße erweitern und dabei die semantische Information des Bildes beibehalten. Bei den Decoding-Schritten werden dabei die Informationen der zugehörigen Encoding Schritte verwendet, um die Segmentierungsinformationen strukturell einzuordnen (GRAFIK?) Encoder-Decoder-Architekturen sind sehr effektiv bei der Segmentierung von Bildern und haben in den letzten Jahren aufgrund ihrer hohen Genauigkeit und Effizienz an Bedeutung gewonnen.

4.2 Datenannotation und Ground Truth-Erstellung

Datenannotation und Ground Truth-Erstellung sind wichtige Schritte bei der semantischen Segmentierung von 3D-Daten, da sie die Trainingsdaten für maschinelles Lernen bereitstellen und Modelle für die Segmentierung von 3D-Daten trainieren und evaluieren. Bei der Datenannotation werden semantische Labels oder Klasseninformationen manuell oder automatisch zu den 3D-Daten hinzugefügt, um sie bestimmten Klassen oder Kategorien zuzuordnen. Die Qualität und Genauigkeit der Datenannotation sind entscheidend für die Leistungsfähigkeit von semantischen Segmentierungsalgorithmen. Die Ground Truth-Erstellung beinhaltet die Erstellung von referenzbasierten Segmentierungsergebnissen, die als Grundlage für das Training und die Evaluierung von semantischen Segmentierungsalgorithmen dienen. Die Ground Truth kann manuell oder automatisch erstellt werden, um die Zuverlässigkeit und Vergleichbarkeit von Segmentierungsergebnissen zu gewährleisten und die Qualität von trainierten Modellen zu überprüfen.

4.3 Evaluierung von Verfahren zur semantischen Segmentierung

Die Evaluierung von Verfahren zur semantischen Segmentierung erfolgt in der Regel anhand von Metriken wie der Ïntersection over Union"(IoU), auch "Jaccard Index"genannt. Dieser Wert gibt an, wie viel Prozent der vorhergesagten Pixel tatsächlich richtig klassifiziert wurden im Verhältnis zu den tatsächlich vorhandenen Pixeln. Weitere Metriken sind die "Pixelgenauigkeit"(Pixel Accuracy), die "Klassen-Genauigkeit"(Class Accuracy) und die "Mittlere-Klassen-Genauigkeit"(Mean Class Accuracy). Für die Evaluierung wird in der Regel ein Testdatensatz verwendet, der sowohl Bilder als auch Ground-Truth-Masken enthält. Anhand dieser Daten wird das Verfahren trainiert und anschließend auf dem Testdatensatz ausgewertet. Die Bewertung der Ergebnisse ermöglicht die Beurteilung der Leistung des Verfahrens und die Vergleichbarkeit mit anderen Ansätzen.

5 Anwendungszenarien der semantischen Segmentierung

5.1 Autonomes Fahren

Eine der vielversprechendsten Anwendungen der semantischen Segmentierung ist das autonome Fahren. Hierbei müssen komplexe Entscheidungen in Echtzeit getroffen werden, um eine sichere Navigation durch die Straßen zu gewährleisten. Mit Hilfe der semantischen Segmentierung können Verkehrsschilder, Straßenmarkierungen, Fußgänger und andere Fahrzeuge automatisch erkannt und identifiziert werden. Dies ermöglicht eine präzisere Steuerung des Fahrzeugs und trägt zu einer höheren Verkehrssicherheit bei.

5.2 Robotik in der Industrie

Die semantische Segmentierung findet auch in der Robotik in der Industrie Anwendung. Durch die Verwendung von Robotern können viele Produktionsprozesse automatisiert werden. Mit Hilfe der semantischen Segmentierung können Roboter ihre Umgebung besser verstehen und präziser auf Veränderungen in der Umgebung reagieren. Dadurch wird die Effizienz und Genauigkeit von Robotern in der Fertigung erhöht.

5.3 Augmented Reality

Augmented Reality ist eine Technologie, die in verschiedenen Bereichen eingesetzt wird, von Unterhaltung bis hin zur medizinischen Ausbildung. Die semantische Segmentierung ermöglicht es, virtuelle Objekte realistischer in die reale Welt zu integrieren. Durch die semantische Segmentierung können virtuelle Objekte auf der Grundlage der Umgebung automatisch positioniert und skaliert werden.

5.4 Stadtplanung

Die semantische Segmentierung wird auch in der Stadtplanung eingesetzt. Durch die automatische Identifizierung von Gebäuden, Straßen und Grünflächen kann die Stadtplanung schneller und präziser durchgeführt werden. Es ermöglicht auch eine schnellere und genauere Analyse von Verkehrsflüssen und städtischen Mustern.

5.5 Umweltüberwachung

Die semantische Segmentierung ist auch in der Umweltüberwachung nützlich. Sie ermöglicht es, bestimmte Objekte und Merkmale in der Umwelt automatisch zu identifizieren und zu überwa-

chen. Beispielsweise kann die semantische Segmentierung verwendet werden, um die Auswirkungen von Umweltverschmutzung auf Wälder oder Flüsse zu überwachen.

6 State-of-the-Art Methoden zur semantischen Segmentierung auf Basis von 3D-Daten

6.1 Überblick über aktuelle Forschung und Entwicklungen

6.2 Bekannte Segmentierungslösungen und deren Funktionsweisen

6.2.1 PointNet

PointNet ist eine auf Punktwolken basierende Architektur für die semantische Segmentierung von 3D-Daten. Im Gegensatz zu anderen Segmentierungsmethoden, die auf Voxel-basierten Darstellungen oder Faltung von 3D-Daten basieren, behandelt PointNet jede Punktwolke als eine Menge von Punkten und verwendet eine Permutationsinvarianz, um diese Punkte zu analysieren. PointNet besteht aus einer einfachen Architektur, die eine eingebettete Vorverarbeitung und eine aufmerksame Pooling-Schicht umfasst. Es hat eine hohe Effizienz und ist sehr genau bei der Segmentierung von Punktwolken.

6.2.2 Voxelnet

VoxelNet ist eine Methode für die 3D-Objekterkennung und -Segmentierung, die eine Voxel-basierte Darstellung verwendet. Es wandelt 3D-Punktwolken in einen dreidimensionalen Voxel-Raum um und verwendet dann eine 3D-Faltungsarchitektur, um Features zu extrahieren. Das extrahierte Feature-Set wird dann für die Vorhersage von 3D-Objektboxen verwendet. VoxelNet hat eine höhere Genauigkeit als andere auf Punktwolken basierende Architekturen wie PointNet, benötigt jedoch auch mehr Rechenressourcen.

6.2.3 SSD (Single Shot MultiBox Detector)

SSD (Single Shot MultiBox Detector) ist eine Methode für die 2D-Objekterkennung und -Segmentierung, die eine Ein-Schritt-Detektionsarchitektur verwendet. Im Gegensatz zu anderen Architekturen, die eine räumliche Pyramiden-Struktur verwenden, verwendet SSD ein Feature-Extraktionsnetzwerk, das von mehreren Detektionsnetzwerken gefolgt wird, um die Objekte in verschiedenen Skalen zu erkennen. SSD ist sehr schnell und hat eine hohe Genauigkeit, jedoch ist es nicht so robust wie andere Methoden wie YOLO.

6.2.4 YOLO (You Only Look Once)

YOLO (You Only Look Once) ist eine Methode für die 2D-Objekterkennung und -Segmentierung, die ebenfalls eine Ein-Schritt-Detektionsarchitektur verwendet. YOLO verwendet ein vollständig konvolutionelles Netzwerk, um die Objekte in einer einzigen Vorwärtsbewegung zu erkennen und zu segmentieren. Es ist sehr schnell und hat eine hohe Genauigkeit bei der Objekterkennung, jedoch hat es Schwierigkeiten, kleine Objekte zu erkennen und zu segmentieren.

Ein bekanntes graphenbasiertes Verfahren für die semantische Segmentierung ist das Graph-CNN-Modell, das ich bereits in meiner vorherigen Antwort erwähnt habe. Es nutzt die Nachbarschaftsbeziehungen zwischen den Pixeln eines Bildes, um das Bild als Graph darzustellen und führt dann Faltungsoperationen auf diesem Graphen aus, um die semantische Information zu extrahieren. Ein weiteres Beispiel für ein graphenbasiertes Verfahren ist das DeepLab-Modell, das eine spezielle Form der Dilated-Konvolutionen auf Graphen anwendet, um die räumliche Auflösung der Feature-Maps zu erhalten.

6.3 Herausforderungen und Limitationen

Die semantische Segmentierung stellt verschiedene Herausforderungen dar, insbesondere im Hinblick auf die Komplexität der Umgebung und die Vielfalt der Objekte und Strukturen. Eine Herausforderung besteht darin, dass Objekte und Strukturen oft unterschiedliche Skalierungen, Formen und Orientierungen aufweisen, was eine präzise Klassifizierung erschwert.

Eine weitere Herausforderung besteht darin, dass die semantische Segmentierung oft in Echtzeit erfolgen muss, um eine zuverlässige Navigation von autonomen Fahrzeugen und Robotern zu ermöglichen. Dies erfordert eine hohe Rechenleistung und eine effiziente Implementierung der Verfahren.

7 Herausforderungen und zukünftige Entwicklungen

7.1 Herausforderungen bei der semantischen Segmentierung von 3D-Daten

Die semantische Segmentierung von 3D-Daten ist eine komplexe Aufgabe in der Computer Vision, die noch zahlreiche Herausforderungen aufweist. Eine der größten Herausforderungen ist die Komplexität der 3D-Daten selbst. Im Gegensatz zu 2D-Bildern haben 3D-Daten zusätzliche Dimensionen, was bedeutet, dass sie viel größer sind und mehr Informationen enthalten. Dies erfordert leistungsfähigere Algorithmen und mehr Rechenleistung, um sie zu verarbeiten.

Ein weiteres Problem bei der semantischen Segmentierung von 3D-Daten ist die mangelnde Verfügbarkeit von geeigneten Datensätzen. Im Gegensatz zu 2D-Bildern gibt es nur wenige öffentlich zugängliche 3D-Datensätze, die ausreichend annotiert sind, um als Trainingsdaten für Algorithmen zu dienen. Dies erschwert die Entwicklung von Algorithmen und macht es schwierig, die Leistung der Modelle zu verbessern.

Zusätzlich erschwert die Vielfalt der 3D-Daten die semantische Segmentierung. Da 3D-Daten aus verschiedenen Quellen stammen können, können sie sehr unterschiedlich sein und unterschiedliche Formen, Größen und Auflösungen aufweisen. Die Herausforderung besteht darin, Algorithmen zu entwickeln, die in der Lage sind, diese Vielfalt zu bewältigen und trotzdem genaue Segmentierungsergebnisse zu liefern.

Ein weiteres Problem bei der semantischen Segmentierung von 3D-Daten ist die Berücksichtigung von Kontextinformationen. Da 3D-Daten in der Regel komplexe Szenen darstellen, ist es wichtig, den Kontext zu berücksichtigen, um genaue Segmentierungsergebnisse zu erzielen. Dies erfordert jedoch Algorithmen, die in der Lage sind, räumliche Zusammenhänge zwischen Objekten zu verstehen und zu modellieren.

Insgesamt gibt es noch viele Herausforderungen bei der semantischen Segmentierung von 3D-Daten. Die Entwicklung von leistungsfähigen Algorithmen, die in der Lage sind, die Komplexität der 3D-Daten zu bewältigen, die Verfügbarkeit von geeigneten Datensätzen zu verbessern und Kontextinformationen zu berücksichtigen, sind nur einige der Herausforderungen, die bewältigt werden müssen, um genaue und zuverlässige semantische Segmentierungsergebnisse zu erzielen.

7.2 Potenziale und Trends für zukünftige Entwicklungen

Die semantische Segmentierung von 3D-Daten ist eine komplexe Aufgabe, die mit mehreren Herausforderungen verbunden ist. Eine der wichtigsten Herausforderungen besteht darin, die hohe Dimensionalität der Daten zu bewältigen. 3D-Daten bestehen aus einer großen Anzahl von Punkten, die alle mit verschiedenen Merkmalen und Eigenschaften versehen sind. Um eine

semantische Segmentierung durchzuführen, müssen diese Merkmale erfasst und interpretiert werden, was einen hohen Rechenaufwand erfordert.

Eine weitere Herausforderung besteht darin, eine ausreichende Menge an gelabelten Daten zu sammeln, die für die Schulung von Algorithmen verwendet werden können. Es ist oft schwierig, genügend qualitativ hochwertige Daten zu sammeln, insbesondere wenn es um seltene oder komplexe Objekte geht.

Zusätzlich ist die Semantik von 3D-Daten oft mehrdeutig und kann von verschiedenen Betrachtungswinkeln abhängen. Beispielsweise können Teile eines Objekts aufgrund ihrer Perspektive oder Positionierung schwer voneinander zu unterscheiden sein. Es ist daher erforderlich, robuste Algorithmen zu entwickeln, die in der Lage sind, diese Herausforderungen zu bewältigen und semantische Segmentierungen von 3D-Daten mit hoher Genauigkeit durchzuführen.

8 Zusammenfassung und Anwendung

8.1 Zusammenfassung der Arbeit

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] 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.

[9] 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.

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

[14] Jonathan Long, Evan Shelhamer und Trevor Darrell. „Fully convolutional networks for semantic segmentation“. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, S. 3431–3440. DOI: 10.1109/CVPR.2015.7298965.

[15] Kaiming He u. a. „Mask R-CNN“. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, S. 2980–2988. DOI: 10.1109/ICCV.2017.322.