

Objektsegmentierung des Datensatzes von TDU-Aluam mit Hilfe von Mask-RCNN und YOLOCT.

Emrullah DERE e180501036@stud.tau.edu.tr
Hussain ZAKROUR e2000507063@stud.tau.edu.tr
Kaan ÖZTÜRK e190501049@stud.tau.edu.tr
Mürsel Erkin ÖZTÜRK e180501005@stud.tau.edu.tr

Türkisch-Deutsche Universität
Fakultät für Ingenieurwissenschaften

MEC319 - Mechatronisches Projekt
2022 WiSe
Gruppe 5

Dozent : Dr. Ing. Soner Emec
Projektbetreuer: BSc. Oğuzhan Memişoğlu

05.01.2023
Istanbul

Kurzfassung

Mask R-CNN ist ein Segmentierungsalgorithmus für Instanzobjekte, der vom Team von Facebook Artificial Intelligence Research (FAIR) entwickelt wurde. Ziel ist es, bestimmte Objekte im Bild zu erkennen und über jedem eine separate Maske zu erstellen. In den letzten Jahren hat die Verwendung der Objektsegmentierung in verschiedenen Bereichen zugenommen.

YOLACT (You only look at the coefficients) ist eine optimiertere Version zum Beispiel für die Segmentierung, die sich einen guten Ruf für ihre Geschwindigkeits- und Genauigkeitskompromisse gesichert hat. Es ist in der Lage, eine mittlere durchschnittliche Genauigkeit von 29,8 auf MS COCO bei 33 Bildern pro Sekunde zu erreichen, viel schneller als die anderen Frameworks der Konkurrenz.

Mask-R-CNN, YOLACT wurden hinsichtlich Effizienz und Benutzerfreundlichkeit verglichen und die optimale Methode für das Projekt ausgewählt. In diesem Projekt wird die Erkennung von belebten und unbelebten Objekten auf dem Campus der Türkisch-Deutschen Universität durchgeführt. Objekterkennungs- und Maskierungsprozesse werden mit der Mask-R-CNN-Struktur angewendet. Zusätzlich wurde ein Datensatz erstellt, um diese Struktur nutzbar zu machen.

Inhaltsverzeichnis

Abbildungsverzeichnis	4
Tabellenverzeichnis	4
1.Einleitung: Objektsegmentierung des Datensatzes von TDU-Aluam	5
1.1.Motivation der Instanzsegmentierung von Innenszenen	5
1.2.Bedarfsanalyse der Bildverarbeitung.....	6
1.3.Ziel des Projekts	6
2. Stand der Technik für Instanzsegmentierung von Innenszenen	7
2.1.Instanzsegmentierung	7
2.2.Ähnliche Projekte.....	7
2.2.1.Floor Segmentation	7
2.2.2.Maske R-CNN für Städtische Suche und Rettung.....	8
2.3.Verwendete Algorithmen zur Objektsegmentierung.....	9
2.3.1.Mask-RCNN	9
2.3.2.You Only Look at the Coefficients (YOLOCT)	10
2.4.Mögliche Anwendungsbereiche für TDU Campus	12
2.4.1.Autonomer Müllsammeln Roboter.....	12
2.4.2.Menschliche Wahrnehmung im TDU-Indoor	12
2.4.3.Kontrolle des Raucherbereichs für TDU-Campus.....	12
3.Arbeitspaketen	13
3.1.Arbeitspaketen für TDU-Aluam-Datensatz und Neuronale Netze	13
3.2.Gantt-Chart des Projekts	14
4.Lösungskonzept für den TDU-Aluam	15
5.Implementierung	18
5.1 Erstellen des Datensatzes	19
5.2 Training anhand Mask-RCNN	19
5.3 Training anhand YOLOCT	20
6. Testen und Evaluierung des Modells	22
7.Literaturverzeichnis	25

Abbildungsverzeichnis

Abb.1.Bildklassifikation, semantische Segmentierung und Instanz Segmentierung.	9
Abb.2.Mask RCNN Architektur.....	9
Abb.3.Yolact Architektur.....	10
Abb.4.Autonomer Müllsammeln Roboter	11
Abb.5.Menschliche Wahrnehmung	12
Abb.6.Masken von der CNC(i) ,Bohrmaschine(ii) Drehmaschine(iii) und Person(iiii)	20
Abb.7. ChatGPT.....	21
Abb.8.Ergebnisse des Models.....	23
Abb.9.Manuelle markiertes Bild	23
Abb.10.Ergebnisse des Models.....	23
Abb.11.Ergebnisse des Models.....	24
Abb.12.Manuelle markiertes Bild	24

Tabellenverzeichnis

Tab.1.Vergleich mit bestehenden Architekturen.....	11
Tab.2.Gantt-Chart	14
Tab.3.Lösungskonzeptskizze	15
Tab.4.Die Flussdiagramm des Algorithmus.....	18
Tab.5.Übersichtsplan von TDU-Aluam	22

Abkürzungsverzeichnis

CNN	Convolutional neural network
R-CNN	Region based Convolutional neural network
YOLACT	Algorithmus (You Look Only at the Coefficients)
TDU	Türkisch-Deutsche Universität

1.Einleitung

In den letzten Jahren hat sich die Entwicklung im Bereich der Objekterkennung und -segmentierung erheblich beschleunigt. Dank intelligenter Algorithmen können unzählige einzelne Objekte in einem Video oder einem Bild gefunden und klassifiziert werden. Obwohl dies für Maschinen anfänglich unglaublich schwierig war, gehört es heute zum Alltag. Sowohl die Objekterkennung als auch die Objektsegmentierung werden durch künstliche Intelligenz, maschinelles Lernen und Deep Learning unterstützt und sie sind in der Lage, Lösungen für lokale oder globale Probleme in verschiedenen Bereichen zu erstellen.

1.2. Bedarfsanalyse der Bildverarbeitung

Die Objektsegmentierung ist ein wichtiger Teil der Bildverarbeitung. Um die Objektsegmentierung gut zu verstehen, muss man wissen, warum Bildverarbeitung im Computer Vision Bereich benötigt wird. Die Bildverarbeitung ist auch ein wichtiger Bestandteil der künstlichen Intelligenz (KI). Durch das Verstehen und Verarbeiten von Bildern können KI-Systeme lernen, Objekte zu erkennen und Muster zu identifizieren. Dies hilft ihnen, bessere Entscheidungen zu treffen und Aufgaben effektiver auszuführen.

Die Bildverarbeitung kann auch verwendet werden, um Bilder zu komprimieren, damit sie auf Speichergeräten oder bei der Übertragung über Netzwerke weniger Platz beanspruchen. Dies ist wichtig, um Kosten zu senken und sicherzustellen, dass Daten schnell und effizient übertragen werden können. Bildverarbeitung kann auch für bedarfsoorientierte Objekterkennung- und Objektsegmentierungsoperationen verwendet werden,

1.3. Zielsetzung

Das Hauptziel dieses Projekts ist es, eine effiziente Deep-Learning-basierte Objektsegmentierung anhand der Mask-RCNN zu trainieren. Das trainierte Modell wird verwendet, um bestimmte Objekte in der Innen- und Außenumgebung der TDU zu segmentieren.

Das erstellte Modell wird in der Lage sein, die bekannten Objekte der Klassen in TDU zu erkennen, zum Beispiel Menschen, Tisch oder Stuhl. Es wird auch Objekte

erkennen können, die sich in der TDU-Outdoor Umgebung befinden, wie Menschen, Autos oder Bäume.

Die Dateibenennung erfolgt automatisch für bekannte Objekte und manuell für unbekannte Objekte.

2. Stand der Technik für Instanzsegmentierung von Innenszenen

2.1 Instanzsegmentierung

Die Instanzsegmentierung ist eine Computervisionsaufgabe zum Erkennen und Lokalisieren eines Objekts in einem Bild. Die Instanzsegmentierung ist eine natürliche Folge der semantischen Segmentierung und auch eine der größten Herausforderungen im Vergleich zu anderen Segmentierungstechniken. Das Ziel der Instanzsegmentierung besteht darin, eine Ansicht von Objekten derselben Klasse zu erhalten, die in verschiedene Instanzen unterteilt sind. Die Automatisierung dieses Prozesses ist nicht einfach, da die Anzahl der Instanzen nicht im Voraus bekannt ist und die Auswertung der erhaltenen Instanzen nicht wie bei der semantischen Segmentierung pixelbasiert erfolgt. Die Segmentierung von Bildinstanzen ist kein erforschtes Gebiet, aber das Interesse wird durch die Möglichkeit der Anwendung in der Praxis motiviert. Das Tagging von Instanzen liefert uns zusätzliche Informationen, um auf unbekannte Situationen zu schließen, Elemente derselben Klasse zu zählen und bestimmte Objekte zu erkennen, die bei Roboteraufgaben abgerufen werden sollen.

2.2 Ähnliche Projekte

2.2.1 Floor Segmentation

Ein auf Deep Learning basierendes Bodensegmentierungsmodell kann verwendet werden, um die Bodenpixel in einem Bild oder Video zu identifizieren und zu segmentieren und sie transparent zu machen. Dies kann für die Bereitstellung und das Testen neuer Bodendesigns in Echtzeit nützlich sein, da Designer sehen können, wie ein neues Bodendesign in einer realen Umgebung aussehen würde, ohne es physisch installieren zu müssen. Das Modell kann mit einem Datensatz von Bildern oder Videos mit beschrifteten Bodenpixeln trainiert und dann auf neue Bilder oder Videos

angewendet werden, um die Bodenpixel zu segmentieren und transparent zu machen. Techniken wie semantische Segmentierung, Objekterkennung und Instanzsegmentierung können verwendet werden, um das Modell zu trainieren. Dies kann für die Bereitstellung und das Testen neuer Bodendesigns in Echtzeit nützlich sein, da Designer sehen können, wie ein neues Bodendesign in einer realen Umgebung aussehen würde, ohne es physisch installieren zu müssen.

2.2.2 Maske R-CNN für Städtische Suche und Rettung

Die Instanzsegmentierung kann in dieser städtischen Such- und Rettungsmission auf verschiedene Weise verwendet werden. Eine Möglichkeit besteht darin, es auf den Bildern oder Punktwolken zu verwenden, die vom Sensor des Erkundungsroboters erfasst werden, um die Ziele zu erkennen und zu lokalisieren. Die Instanzsegmentierung ist eine Art der Objekterkennung und Bildsegmentierung, die einzelne Objekte innerhalb eines Bildes identifizieren und segmentieren kann, selbst wenn sie von derselben Klasse sind und sich überschneiden. Dies ist besonders nützlich bei Such- und Rettungsszenarien, bei denen sich möglicherweise mehrere Personen oder Objekte im selben Bereich befinden und einzeln identifiziert und lokalisiert werden müssen.

Eine andere Möglichkeit, die Instanzsegmentierung zu verwenden, sind die Sensordaten des Follower-Roboters. Beispielsweise kann der Folgeroboter eine Kamera verwenden, um Bilder des Bereichs aufzunehmen, zu dem er navigiert, und eine Instanzsegmentierung verwenden, um das Ziel, das er retten muss, innerhalb des Bilds zu identifizieren und zu lokalisieren. Diese Informationen können dann verwendet werden, um einen Weg zu planen und zum Ziel zu navigieren.

Der Erkundungsroboter, auch als Suchroboter bekannt, ist dafür verantwortlich, das Gebiet nach Zielen abzusuchen und deren Standorte an den Folgeroboter zu übertragen. Der Erkundungsroboter verwendet einen Sensor wie eine Kamera oder LIDAR, um die Ziele zu erkennen und ihren Standort zu bestimmen. Die Position der Ziele wird dann als tf-Frame gesendet, bei dem es sich um eine Datenstruktur handelt, die im Robot Operating System (ROS) zum Koordinieren der Positionen mehrerer Objekte in der Umgebung eines Roboters verwendet wird.

Der Follower-Roboter, auch als Rettungsroboter bekannt, lauscht auf die gesendeten tf-Frames vom Explorer-Roboter. Sobald es den Standort eines Ziels erhält, verwendet es seine eigenen Sensoren, wie z. B. einen Roboterarm oder Greifer, um zum Ziel zu navigieren und die Rettungsmission durchzuführen. Es kann

Pfadplanungsalgorithmen verwenden, um zum Ziel zu navigieren.

2.3 Verwendete Algorithmen zur Objektsegmentierung

2.3.1 Mask-RCNN

Mask R-CNN ist ein tiefes neuronales Netzwerk, das darauf abzielt, das Problem der Instanzsegmentierung beim maschinellen Lernen oder Computer Vision zu lösen. Mit anderen Worten, es kann verschiedene Objekte in einem Bild oder Video trennen.

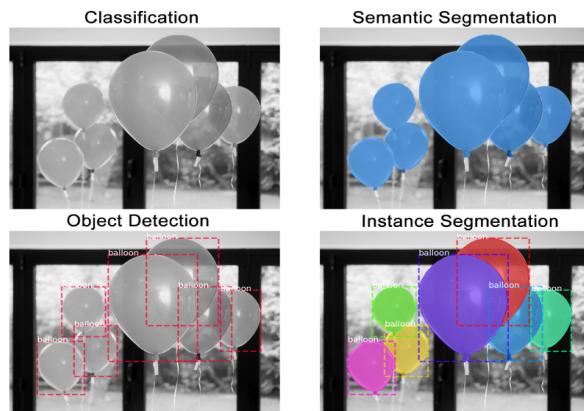


Abb.1. Bildklassifikation, semantische Segmentierung, Objekterkennung und Instanz Segmentierung.

In der Abbildung 5 sieht man vier Bilder. Bei Bild (a), der Bildklassifikation, wird nur ausgesagt, dass sich in diesem Bild eine unbenannte Anzahl von Ballon befindet. Bei zweiten Bild, der semantischen Segmentierung, wird zwischen Hintergrund und Ballons mit Maske unterschieden. Bei dritten Bild, der Objekte Lokalisation, wird die Position des Ballons markiert. Jede Objektklasse bekommt eine Farbe. Bei vierten Bild, der Instanz Segmentierung wird nicht nur zwischen Objektklassen unterschieden, sondern auch unter Instanzen. Die sieben Ballons haben verschiedene Farben.

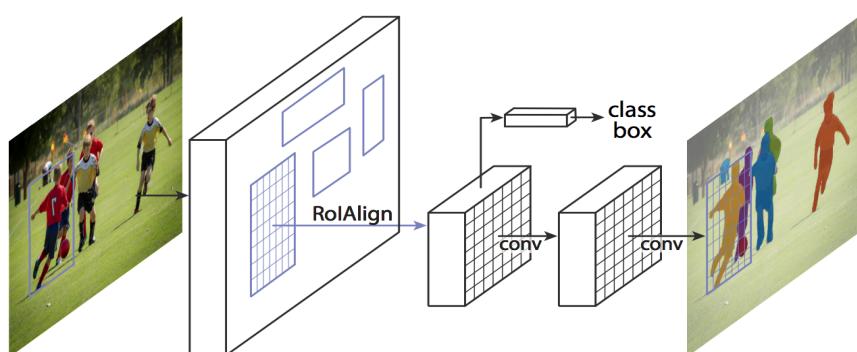


Abb.2 Mask RCNN Architektur.

Mask R-CNN ist ein zweistufiges Framework: Die erste Stufe scannt das Bild und generiert Vorschläge. Und die zweite Stufe klassifiziert die Vorschläge und generiert Begrenzungsrahmen und Masken.

Faster R-CNN ist ein beliebtes Framework zur Objekterkennung und Mask R-CNN erweitert es unter anderem um Instanzsegmentierung.

2.3.2. You Only Look at the Coefficients (YOLACT)

Die Netzwerke, die gute Ergebnisse bei der Instanzsegmentierung liefern, sind FCIS, Mask-R CNN, RetinaMask, PA-Net usw. Diese Frameworks funktionieren zwar relativ gut, aber die daraus gewonnenen Inferenzen können aufgrund der Rechenkomplexität, die mit der Erstellung solcher Systeme verbunden ist, nicht in „Echtzeit“ verwendet werden. Die schiere Anzahl von Parametern macht es für diese Netzwerke unmöglich auf Maschinen mit geringerer Rechenleistung auszuführen. Die Aufgabe erfordert daher eine andere Architektur, die in der Lage ist, Berechnungen in „Echtzeit“ durchzuführen.

YOLACT (You only look at the coefficients) ist eine optimiertere Version zum Beispiel für die Segmentierung, die sich einen guten Ruf für ihre Geschwindigkeits- und Genauigkeitskompromisse gesichert hat. Es ist in der Lage, eine mittlere durchschnittliche Genauigkeit (mAP) von 29,8 auf MS COCO bei 33 Bildern pro Sekunde zu erreichen, viel schneller als die anderen Frameworks der Konkurrenz.

YOLACT verwendet ResNet-101 mit FPN (Feature Pyramid Networks), das hilft, Pyramiden von Feature-Maps aus hochauflösenden Bildern zu erstellen, anstelle des traditionellen Pyramid-of-Image-Ansatzes, wodurch Zeit und Anforderungen an Rechenkapazitäten reduziert werden reduziert. Der Rechenaufwand wird weiter reduziert, indem nur auf höherer Ebene extrahierte Feature-Layer verwendet werden, die einen höheren semantischen Wert haben.

Die Architektur weist einen Top-Down-Pfad auf, der die Konstruktion der Schichten mit höherer Auflösung aus diesen semantisch reichen Schichten ermöglicht, um die Erkennungsfähigkeiten für die Positionen der Objekte nach dem Upsampling und Downsampling aufrechtzuerhalten, gibt es seitliche Verbindungen zwischen den rekonstruierten Schichten und den entsprechenden Funktionskarten.

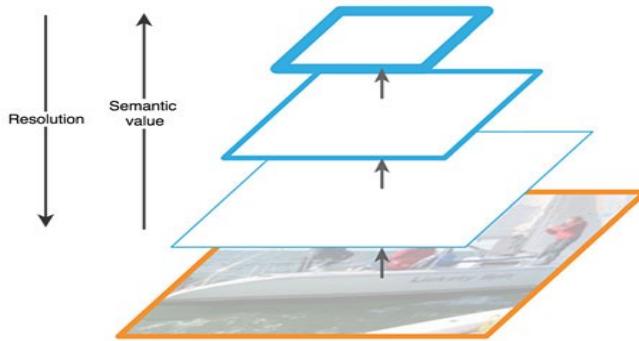
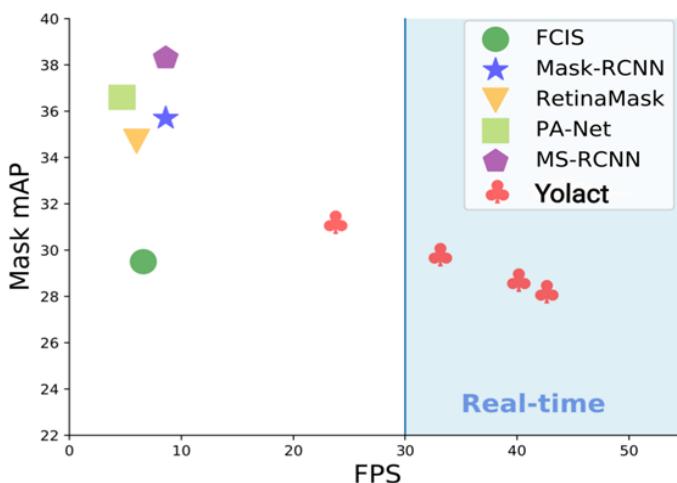


Abb.3 Yolact Architektur

- Die Auflösung und der semantische Wert sind beide in der gegebenen Architektur umgekehrt proportional.

Vergleich mit bestehenden Architekturen:

YOLACT hat eine Reihe von Vorteilen gegenüber den bestehenden Architekturen, einer der wichtigsten wird die Geschwindigkeit der Vorhersagen sein. YOLACT ist die einzige bestehende Architektur, die tatsächlich "Echtzeit"-Inferenz liefern kann. Obwohl YOLACT in Bezug auf Arbeitsgeschwindigkeit und Genauigkeit besser ist, gibt es viele Fehler aufgrund von Ressourcenmangel, die Version ist nicht aktuell und macht es unmöglich zu arbeiten. Die folgende Abbildung vergleicht die Leistung verschiedener Architekturen mehrerer Variationen des YOLACT:



Tab.1 Vergleich mit bestehenden Architekturen

2.4. Mögliche Anwendungsbereiche für TDU Campus

Das von uns vorbereitete Projekt ist offen für eine Adaption für den Einsatz in verschiedenen Bereichen als Konzept. Zusätzlich zu verschiedenen Anwendungen,

die durch Anpassung an Systeme wie Kameras gemacht werden können, kann es auf Roboter und ähnliche Hardware mit kompatibler Software angewendet werden.

2.4.1.Autonomer Müllsammeln Roboter

Wenn es auf Campus Basis gedacht wird, ist der Datensatz bereits fertig. Daten werden hinzugefügt, um Müll zu erkennen. Einem richtig konstruierten Roboter wird beigebracht, Müll zu erkennen und ihn in seinem Reservoir zu sammeln, indem er ihn vom Boden aufhebt. dann kann mit unserem programm zur erkennung ein Müllsammeln Roboter erstellt werden. Durch die Entwicklung und Modifikation des Datensatzes kann dieser Roboter überall eingesetzt werden. Es kann sogar für das Recycling geeignet gemacht werden, indem es Schulungen zu dem Objekt gibt, das es erkennt. oder es kann ein Roboter erworben werden, der die erforderlichen Produkte in die verschiedenen verbleibenden einführt und einsammelt.



Abb.4: Autonomer Müllsammeln Roboter



Abb.5: Menschliche Wahrnehmung

2.4.2.Menschliche Wahrnehmung im TDU-Indoor

Es gibt mehrere Projekte, die durchgeführt werden können, indem Personen von Sicherheitskameras erkannt werden. Warnsysteme können entwickelt werden, indem Kameras erkannt werden, wenn Personen verbote Bereiche betreten. Oder die Lebenssicherheit von Menschen in einem Gefahrenbereich kann erhöht werden. Durch die Berechnung der Personenzahl mit einem Software-Plug-in kann die Dichte in bestimmten Bereichen berechnet werden. Auf einem Campus mit mehr als einer Cafeteria kann beispielsweise die Berechnung, welche Cafeteria überfüllt ist und welche schneller essen kann, die Pausen Effizienz erhöhen, indem es Menschen mit begrenzten Pausenzeiten ermöglicht wird, ihre Mahlzeiten schneller zu bekommen.

2.4.3.Kontrolle des Raucherbereichs für TDU-Campus

Ein mit Überwachungskameras integriertes Projekt ist möglich, um in Bereichen mit Rauchverbot eine komfortable Kontrolle zu ermöglichen und einen Verstoß gegen das Verbot zu verhindern. Es ist möglich, das Projekt mit einem Paket zum Leben zu erwecken, das die Zigaretten Erkennung in den Datensatz und die Integration vorhandener Überwachungskameras ermöglicht.

3.Arbeitspaketen

3.1.Arbeitspaketen für Datensatz und Neuronale Netze

Arbeitspakete	Input	Funktion	Output	Verantwortlicher	Abgabetermin
1. Theoretischer Teil Objektsegmentierung mit bestimmten Algorithmen	der Literatur recherchieren, z.B. Google Scholar	Recherche und Überprüfung verschiedener Forschungsergebnisse und Studien. Allgemeines Wissen aus ähnlichen Projekten.	Allgemeine Informationen zur Objektsegmentierung	Emrullah Dere	21.10.2022
2.1 Untersuchung des entsprechenden Datensatzes.	Recherche verschiedener wissenschaftlicher Artikel.	-Untersuchung und Vergleich verschiedener Datensätze in ähnlichen Forschungen und Projekten.	Den passenden Datensatz finden.	Emrullah Dere	27.10.2022
2.2. Herunterladen vor trainierter Gewichte	-COCO Dataset	-Herunterladen von trainierten Gewichten herunterladen. Diese Gewichte werden von einem Modell erhalten, das mit dem MS COCO-Datensatz trainiert wurde.	Finden des Datensatzes für bekannte Objekte	Emrullah Dere	18.11.2022
2.3 Aufteilung des fertigen Datensatzes.	-Untersuchung der stückweisen Verarbeitung des Datensatzes.	-Reservieren von 70 % des Datensatzes für das Training und 30 % für Tests. Ein kleiner Anteil von 70 % wird zur Validierung des Train-Datensatzes verwendet.	Aufteilen des Datensatzes in drei Bereiche wie Trainieren, Validieren und Testen.	Emrullah Dere	25.11.2022

Arbeitspakete	Input	Funktion	Output	Verantwortlicher	Abgabetermin
3.1. Aufnahme der Fotos und Videos von der TDU	Aufnahme der Fotos und Videos	-Auf einer bestimmten Strecke und aus verschiedenen Blickwinkeln wurden Aufnahmen innerhalb und außerhalb der Universität gemacht.	Fotos und Videos der zu segmentierten Objekte	Hussain Zakrour	27.10.2022
3.2 Aufbereitung des Datensatzes	-VIA (VGG Image Annotator)	-Image Annotator ist eine einfache und eigenständige manuelle Anmerkung Software für Bilder.	Finden des Datensatzes für unbekannte Objekte.	Hussain Zakrour	22.11.2022
3.3 Aufteilung des fertigen Datensatzes.	-Untersuchung der stückweisen Verarbeitung des Datensatzes.	-Reservieren von 70 % des Datensatzes für das Training und 30 % für Tests. Ein kleiner Anteil von 70 % wird zur Validierung des Train-Datensatzes verwendet.	Aufteilen des Datensatzes in drei Bereiche wie Trainieren, Validieren und Testen.	Hussain Zakrour	25.11.2022
4.1. Recherchieren der neuronalen Netze	-der Literatur recherchieren z.B Google Scholar	Sammeln erforderliche Informationen, über neuronalen Netze	Lernen, wie die neuronalen Netze funktionieren	Kaan Öztürk	25.10.2022
4.2. Training mit neuronalen Netzen	-Mask RCNN	Verarbeitung der Bilden mit dem recherchierten Datensatz auf Mask RCNN Architektur	Bereit für die letzte Phase des Datensatz- und Netzwerk Trainings	Kaan Öztürk	20.11.2022
4.3 Einsatz von Google Collab	-Python -Coco Daten	Implementierung von Python-Codes und Coco Daten mittels Google Collab	Training abgeschlossen	Kaan Öztürk	28.12.2022

Arbeitspakete	Input	Funktion	Output	Verantwortliche r	Abgabetermin
5.1. Recherchieren der neuronalen Netze	-der Literatur recherchieren z.B Google Scholar	Sammeln erforderliche Informationen, über neuronalen Netze	Lernen, wie die neuronalen Netze funktionieren	Mürsel Erkin Öztürk	25.10.2022
5.2. Training mit neuronalen Netzen	-YOLOCT	Verarbeitung der Bilden mit dem recherchierten Datensatz auf YOLOCT Architektur	Bereit für die letzte Phase des Datensatz- und Netzwerktraining	Mürsel Erkin Öztürk	20.11.2022
5.3. Einsatz von Google Collab	-Python -Coco Daten	Implementierung von Python-Codes und Coco Daten mittels Google Collab	Training abgeschlossen	Mürsel Erkin Öztürk	28.12.2022
6.Test des gesamten Modells	-Test mit Fotos -Test mit Video	Tests, um zu überprüfen, ob das vorbereitete Programm ordnungsgemäß funktioniert.	Das Projekt ist bereit und funktioniert ordnungsgemäß	Mürsel Erkin Öztürk	01.01.2023

3.2.Gantt-chart des Projekts

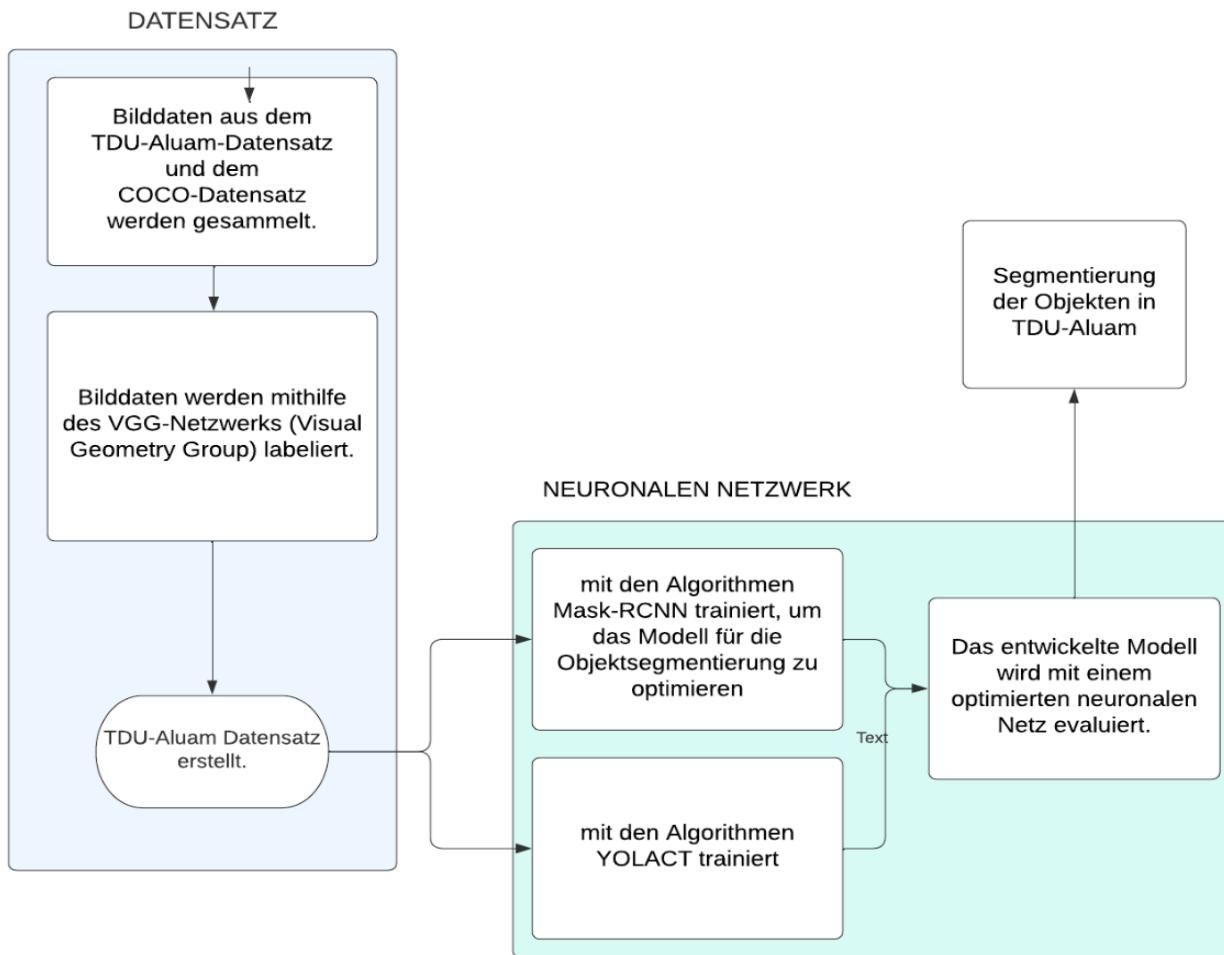
Tab.2 Gantt-Chart

Projektname Projektleitung Gruppe 5	Objektsegmentierung anhand Mask-RCNN im TDU Dr. Ing. Soner Imec			Startdatum Enddatum	19.10 5.1																		
		Aufgaben	Verantwortlich	Start	Ende	Tag	Status	17.10	22.10	27.10	1.11	6.11	11.11	16.11	21.11	26.11	2.12	7.12	12.12	17.12	22.12	27.12	1.1
1. Vorarbeiten	Emrullah Dere						Abgeschlossen																
1.1. Kick-off-Meeting		19.10	19.10		1		Abgeschlossen																
1.2. Ziele vereinbaren		19.10	21.10		2		Abgeschlossen																
2.Datensatz für bekannte Objekte	Emrullah Dere						Abgeschlossen																
2.1. Untersuchung des entsprechenden Datensatzes.		20.10	27.10		7		Abgeschlossen																
2.2. Aufbereitung des Datensatzes		28.10	18.11		30		Abgeschlossen																
2.3. Aufteilen des fertigen Datensatzes		19.11	25.11		6		Abgeschlossen																
3.Datensatz für unbekannte Objekte	Hussain Zakrour						Abgeschlossen																
3.1. Aufnahme der Fotos und Videos von der TDU		20.10	27.10		7		Abgeschlossen																
3.2. Aufbereitung des Datensatzes		28.10	22.11		34		Abgeschlossen																
3.3. Aufteilen des fertigen Datensatzes		22.11	25.11		3		Abgeschlossen																
4.Training mit Mask-RCNN	Kaan Öztürk						Abgeschlossen																
4.1. Recherchieren der Mask R-CNN		19.10	25.10		6		Abgeschlossen																
4.2. Training mit Mask R-CNN		26.10	22.11		27		Abgeschlossen																
4.3 Einsatz von Google Collab		23.11	14.12		21		Abgeschlossen																
5.Training mit YOLACT	Mürsel Öztürk						Abgeschlossen																
5.1. Recherchieren der YOLACT		19.10	25.10		6		Abgeschlossen																
5.2. Training mit YOLACT		26.10	22.11		27		Abgeschlossen																
5.3 Einsatz von Google Collab		23.11	14.12		21		Abgeschlossen																
6.Test des gesamten Modells	Mürsel Öztürk						Abgeschlossen																
6.1. Testen die Ergebnisse mit Fotos		15.12	20.12		5		Abgeschlossen																
6.2. Testen mit Ergebnissen mit Video		20.12	1.1		12		Abgeschlossen																
Meilensteine:																	M1	M2	M3	M4			
M1:Modellskizze und Lösungskonzept	M2:Bereitstellen von Datensätzen	M3:Analyse und Training	M4:Fertigstellung des Modells																				
Bestimmung des Datensatzes für bekannte Objekte. Notwendige Fotos für unbekannte Objekte machen und notwendige Klassen für verschiedene Datensätze erstellen.	Gruppieren der Datensätze, deren Maskierung und Benennung abgeschlossen sind, unter geeigneten Klassen und Bereitstellen für das Training.	Analyse und Training von Datensätzen in Mask-RCNN und YOLO Algorithmen in diesem Meilenstein.	Fertigstellung des getesteten Modells.																				

4.Lösungskonzept für den TDU-Aluam

Es geht in diesem Abschnitt um die Lösungskonzept unseres Projektes. Die verwendeten Algorithmen und Verfahren für die Objektsegmentierung- und Merkmalbestimmung Prozesse werden In diesem Kapitel genannt werden. Der gesamte Prozess kann folgendes durch fünf Aspekten zusammengefügt:

1. Bilddaten sammeln
2. Bilddaten werden mithilfe VGG (Bild-Kommentator) labeln.
3. Der erstellte Datensatz wird mit den Algorithmen Mask-RCNN und YOLACT trainiert.
4. Das entwickelte Modell wird mit einem optimierten neuronalen Netz evaluiert.
5. Baseline Modell von Objekten in TDU-Aluam wird erstellt.



Tab.3 Lösungskonzeptskizze

Bilddaten aus dem TDU-Aluam-Datensatz und dem COCO-Datensatz werden gesammelt und in verschiedene Klassen wie z.B. Personen, Maschine, Tisch, Stuhl usw. eingeteilt. Diese Klassifizierung ermöglicht es, die Bilder gezielt für die Ausbildung des Modells auszuwählen.

Bilddaten werden mithilfe des VGG-Netzwerks (Visual Geometry Group) beschriftet. Das VGG-Netzwerk ist ein tiefes konvolutionelles neuronales Netzwerk, das speziell für die Bilderkennung entwickelt wurde. Es besteht aus mehreren Schichten von konvolutionellen und pooling-Schichten, die zusammen arbeiten, um die Merkmale eines Bildes zu extrahieren und zu erkennen. In diesem Prozess wird das VGG-Netzwerk auf die gesammelten Bilddaten angewendet, um automatisch die Bilder zu kategorisieren und zu beschriften. Dazu wird das VGG-Netzwerk trainiert, um bestimmte Muster in den Bildern zu erkennen und zu klassifizieren. Beispielsweise kann es gelernt haben, dass bestimmte Merkmale wie Kanten, Farben und Texturen mit bestimmten Objekten assoziiert sind. Nach dem Training wird das VGG-Netzwerk verwendet, um die Bilder automatisch zu beschriften, indem es die höchstwahrscheinliche Klasse für jedes Bild vorschlägt. Dies reduziert den Aufwand des manuellen Labels erheblich und erhöht die Effizienz des Prozesses. Es ist zu beachten, dass das VGG-Netzwerk lediglich für die automatische Beschriftung der Bilder verwendet wird.

und die endgültige Entscheidung über die Korrektheit des Labels immer noch von menschlichen Annotatoren getroffen werden muss, da das VGG-Netzwerk nicht in der Lage ist, alle Feinheiten der Bilder zu erfassen und kann Fehler machen.

Der erstellte Datensatz, der aus den labelierten Bilddaten besteht, wird dann mit den Algorithmen Mask-RCNN und YOLACT trainiert, um das Modell für die Objekterkennung zu optimieren. Mask-RCNN ist ein Algorithmus zur Instanz Segmentierung, der es ermöglicht, die Position und die Form jedes Objekts in einem Bild zu identifizieren. Der Algorithmus verwendet ein Region-Based Convolutional Neural Network (R-CNN), das vorgeschlagene Regionen im Bild identifiziert und dann ein weiteres Netzwerk verwendet, um die Klasse und die Masken der Objekte innerhalb der Regionen vorherzusagen. Durch die Verwendung von Masken werden die Grenzen der Objekte genau definiert und es wird eine höhere Genauigkeit der Objekterkennung erreicht.

YOLACT ist ein Algorithmus zur Instanz Segmentierung, der es ermöglicht, die Position und die Form von Objekten in einem Bild zu identifizieren. Es verwendet ein You Only Look At Coefficients (YOLACT) Konzept, das es ermöglicht, die Regionen von Objekten zu identifizieren und zu segmentieren, indem es die wichtigsten Merkmale eines Bildes hervorhebt und die Regionen, die diese Merkmale enthalten, vorschlägt. Dieser Algorithmus ist besonders leistungsfähig bei der Erkennung von Objekten in Bildern mit hoher Dichte und ermöglicht eine schnellere Verarbeitung im Vergleich zu anderen Algorithmen.

Durch die Verwendung dieser Algorithmen wird der Datensatz trainiert, um das Modell für die Objekterkennung zu optimieren. Das Modell lernt, die Merkmale der Objekte in den Bildern zu erkennen und zu klassifizieren, und wird immer besser darin, neue Bilder richtig zu labeln, je mehr Bilder es sieht und trainiert wird.

Das entwickelte Modell wird dann mit einem optimierten neuronalen Netz evaluiert, um die Genauigkeit der Objekterkennung zu bestimmen. Dieser Schritt ist wichtig, um sicherzustellen, dass das Modell tatsächlich in der Lage ist, Objekte in Bildern zu erkennen und zu klassifizieren, und um zu bestimmen, ob es notwendig ist, weitere Optimierungen durchzuführen.

Es gibt verschiedene Metriken, die verwendet werden können, um die Leistung des Modells zu messen, wie z.B: Intersection over Union (IoU), Average Precision (AP) oder F1-Score. Der IoU misst die Übereinstimmung zwischen den vorhergesagten und den tatsächlichen Masken der Objekte und gibt einen Prozentsatz aus, der angibt, wie viel der Fläche der vorhergesagten Maske mit der Fläche der tatsächlichen Maske übereinstimmt. Der AP misst die Genauigkeit der Klassifizierung der Objekte und gibt einen Prozentsatz aus, der angibt, wie viele der vorhergesagten Objekte tatsächlich in den Bildern vorhanden sind. F1-Score ist eine Kombination aus precision and recall. Es gibt einen Durchschnittswert, der die Leistung des Modells bei der Erkennung der Objekte in Bezug auf die Genauigkeit und die Recall misst.

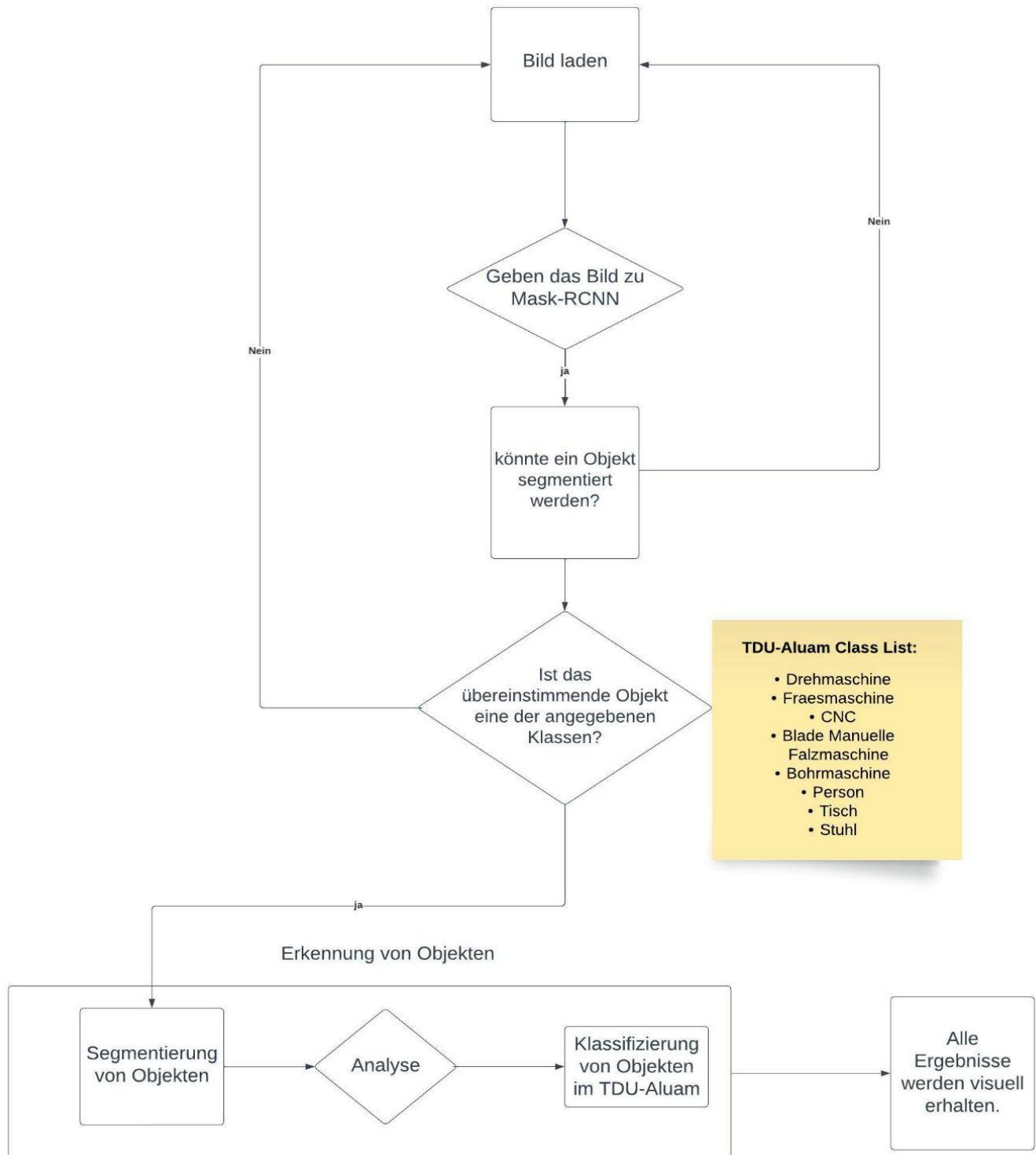
Es ist wichtig zu beachten, dass die Ergebnisse der Evaluation auf einer Test-Datenmenge basieren, die vom Trainings-Datensatz getrennt ist. Es gibt keine Garantie, dass das Modell auf neuen, ungesiehenen Daten genauso gut funktionieren

wird, aber es gibt eine gute Schätzung der Leistungsfähigkeit des Modells. Falls die Ergebnisse nicht zufriedenstellend sind, kann das Modell weiter optimiert werden, indem z.B. neue Daten hinzugefügt oder die Architektur des neuronalen Netzes geändert wird.

5.Implementierung

In diesem Kapitel werden die im Projekt verwendeten Algorithmen und die notwendigen Bibliotheken erwähnt.

Tab.4 Die Flussdiagramm des Algorithmus



5.1 Erstellen des Datensatzes

Der MS COCO-Datensatz ist ein umfangreicher Datensatz zur Objekterkennung, -segmentierung und -kennzeichnung, der von Microsoft veröffentlicht wird. Ingenieure für maschinelles Lernen und maschinelles Sehen verwenden den COCO-Datensatz häufig für verschiedene Computer-Vision-Projekte. Bestimmte Klassen wurden aus dem Coco-Datensatz für bekannte Objekte in TDU

heruntergeladen.

Klassen für bekannte Objekte:

- Die Person (66808 Ergebnisse)
- Der Stuhl (13354 Ergebnisse)
- Der Tisch (12338 Ergebnisse)

Für unbekannte Objekte an der TDU wurden Aufnahmen innerhalb und außerhalb der Universität gemacht.

Klassen für unbekannte Objekte:

- Fräsmaschine
- Bohrmaschine
- CNC (NMV 106A)
- Drehmaschine
- Blade Manuelle Falzmaschine

Punkte, die beim Fotografieren zu beachten sind:

- Dynamisches / statisches Bild.
- Helle / dunkle Umgebung.
- Direkte Bewegung auf einer vorgegebenen Route / Blick nach links und rechts.

Technische Details der aufgenommenen Fotos:

- Bildauflösung: 640×480 (gleicher Wert wie Bilder im Coco-Datensatz).
- Sony IMX586 Sensor mit f/1.8 Blende.

5.2 Training anhand Mask-RCNN

Mask R-CNN ist ein tiefes neuronales Netzwerk, das darauf abzielt, das Problem der Probensegmentierung beim maschinellen Lernen oder Computer Vision zu lösen.

Der Mask-RCNN Algorithmus wird verwendet, um die Objekte im TDU-Aluam Datensatz zu maskieren.

Masken der einigen Objekten im TDU-Aluam:

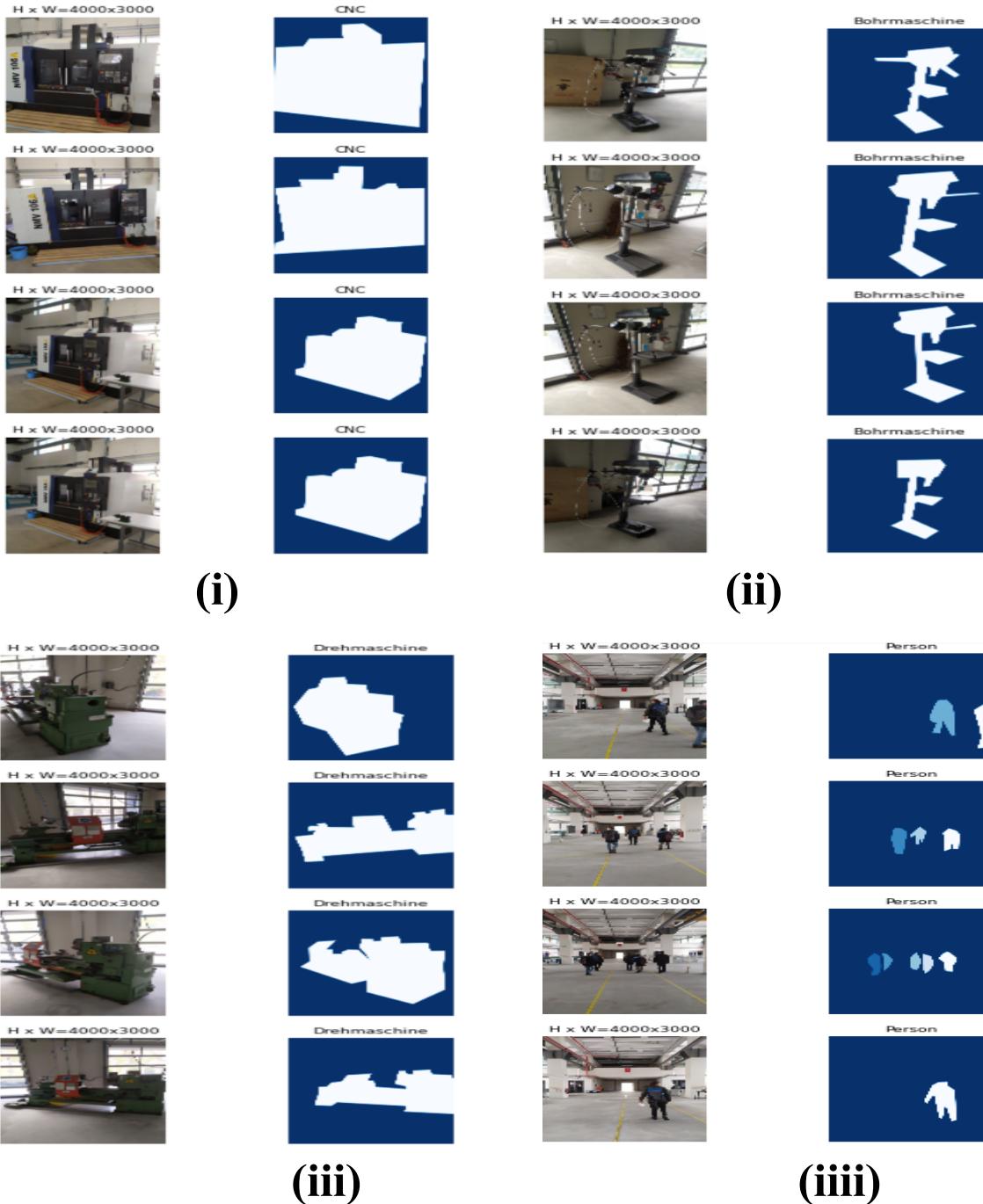


Abb.6 Masken von der CNC(i) ,Bohrmaschine(ii) Drehmaschine(iii) und Person(iiii)

Der entsprechende mask-rcnn-Code kann über den unten stehenden Github-Link aufgerufen werden:

<https://github.com/EmrullahDere/Objektsegmentierung-von-TDU-ALUAM-anhand-Neuronale-Netze-Gruppe5>

5.3 Training anhand YOLACT

Die richtigen Torchvision- und Torch-Pakete für die Verwendung von YOLACT gefunden. Die erforderlichen Git-Repos wurden dem Projekt hinzugefügt. DCNv2 kompiliert. Trainierte Gewichtssätze wurden gesenkt. Vorbereitete Datensätze wurden für das Training hinzugefügt. Nach Abschluss des Trainings soll es auf Fotos und Videos verwendet werden.

Das erste Problem bestand darin, die richtigen Versionen von Torch und Torchvision zu finden. Keine Version gefunden, damit Dcnv2 richtig funktioniert. Während alte Versionen Probleme mit dem Training hatten, während Dcnv2 ausgeführt wurde, konnten neue Versionen Dcnv2 nicht richtig ausführen. Wir haben ChatGPT aktiv genutzt, während wir versuchten, Probleme zu lösen. Während seine Vorschläge unsere Probleme Schritt für Schritt lösten, stießen wir in jeder Phase auf ein anderes Problem.

Danach wurden zwar die Gewichtsdateien, aber die Datensätze für das Training geladen, und alles Notwendige für das Training war fertig, Dcnv2 konnte nicht ausgeführt werden, sodass keine Ergebnisse erzielt werden konnten.

Während wir die Ergebnisse mit Mask-R-CNN verglichen und den optimalen Weg wählen möchten. Es wurde der Schluss gezogen, dass YOLACT nicht einmal Ergebnisse lieferte, zu viele Fehler verursachte und nicht zum Arbeiten geeignet war, da keine Lösung hergestellt werden konnte.

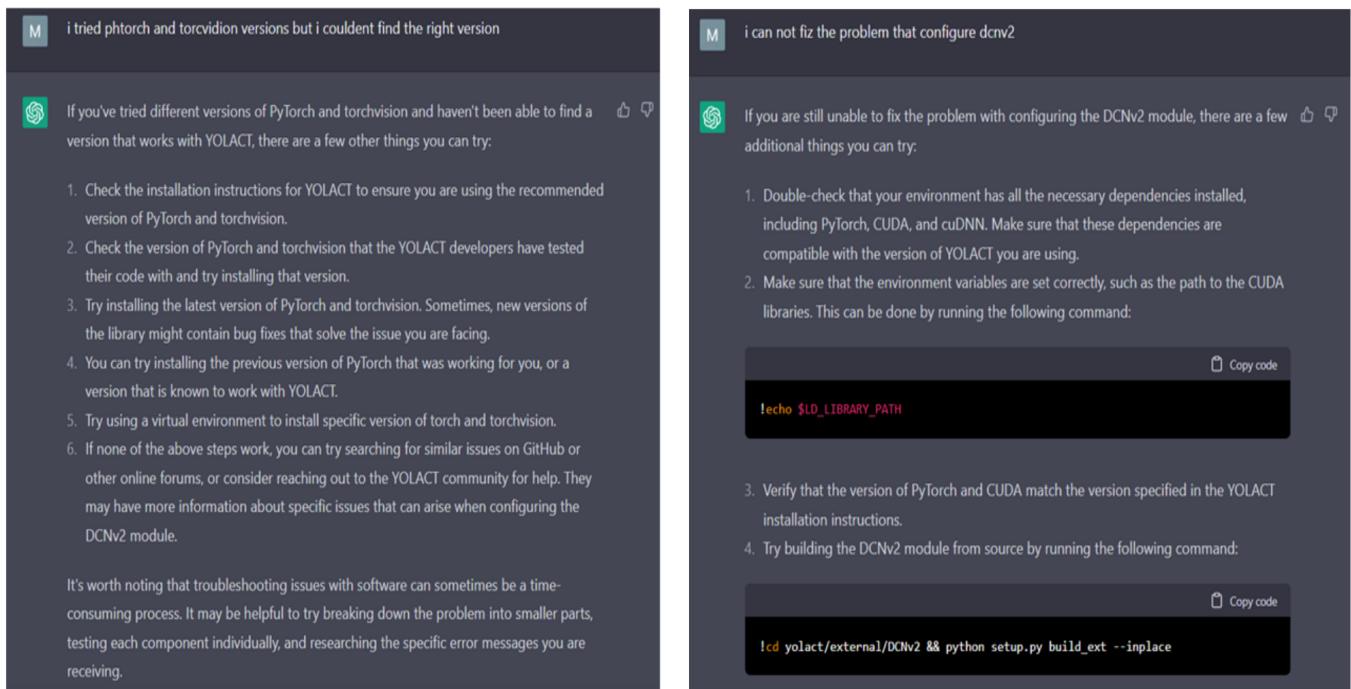


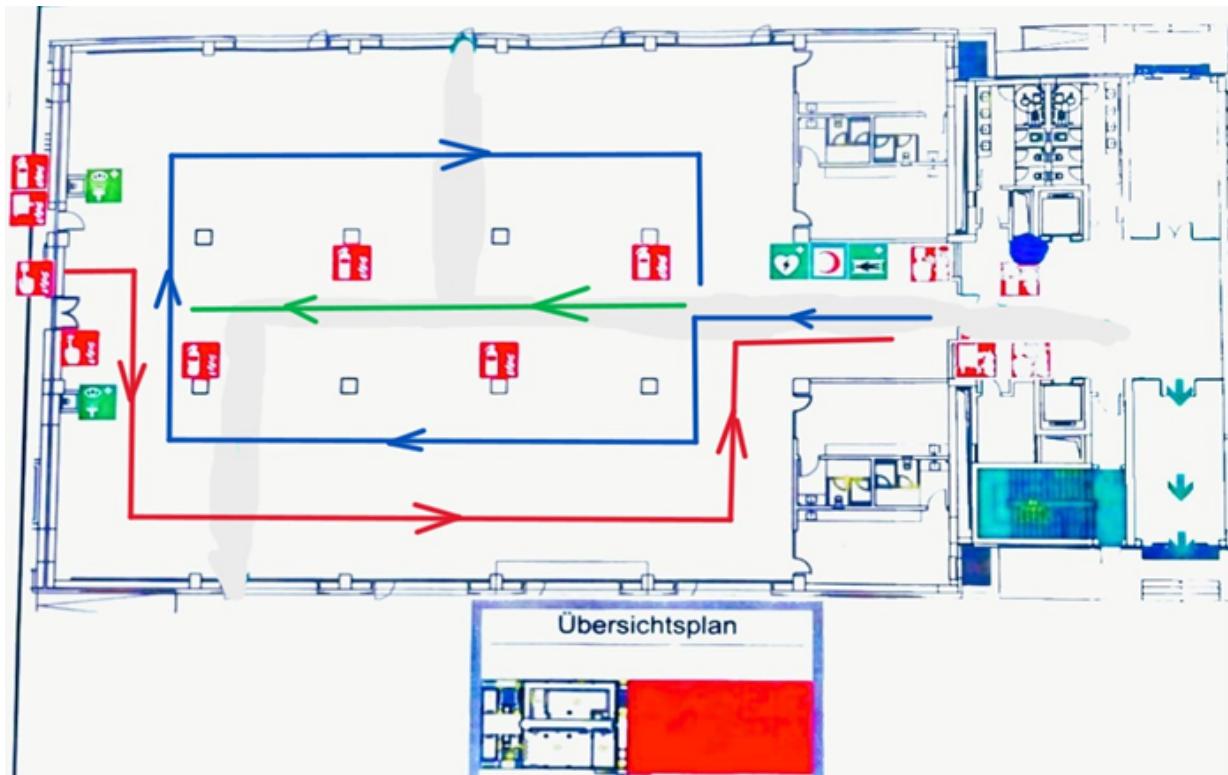
Abb.7 Chat-GPT Nutzungsbeispiele

Der entsprechende YOLACT-Code kann über den unten stehenden Github-Link aufgerufen werden: <https://github.com/MurselErkin/YOLACT>

6. Testen und Evaluierung des Modells

In diesem Abschnitt wird unser Modell getestet. In diesem Test werden verschiedene Eigenschaften von neuronalen Netzen und deren Ergebnisse bei verschiedenen Routen und Bedingungen getestet.

Tab.5 Übersichtsplan von TDU-Aluam



Route 1 (Rot)

- Hell, direkte Bewegung, dynamisches Bild
- Hell, nicht direkte Bewegung, dynamisches Bild

Route 2 (Blau)

- Hell, direkte Bewegung, dynamisches Bild
- Hell, nicht direkte Bewegung, dynamisches Bild

Route 3 (Grün)

- Hell, direkte Bewegung, dynamisches Bild
- Hell, direkte Bewegung, statisches Bild

- Route 1 : Hell,nicht direkte Bewegung, dynamisches Bild:



Abb.8 Ergebnisse des Models



Abb.9 Manuelle markiertes Bild

Im Vergleich zum manuell maskierten Bild macht es die Maskierung grundsätzlich korrekt, aber es gibt an einigen Stellen einige Fehler, zum Beispiel hat es die Kanten der Drehmaschine weicher bestimmt.

- Route 2 : Hell, nicht direkte Bewegung, dynamisches Bild:

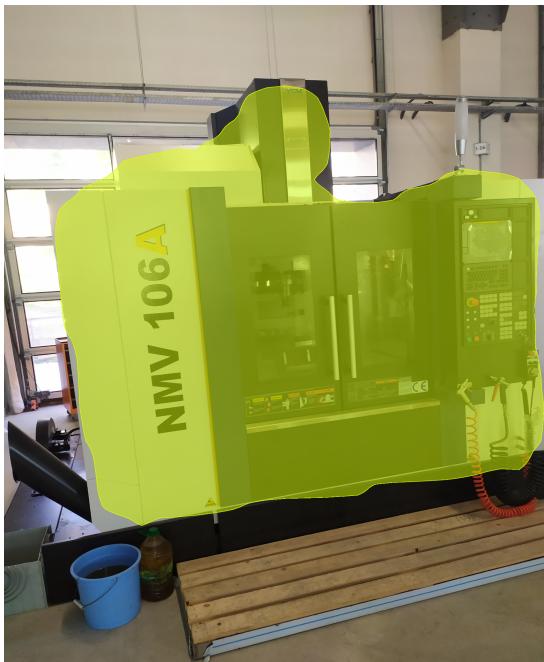
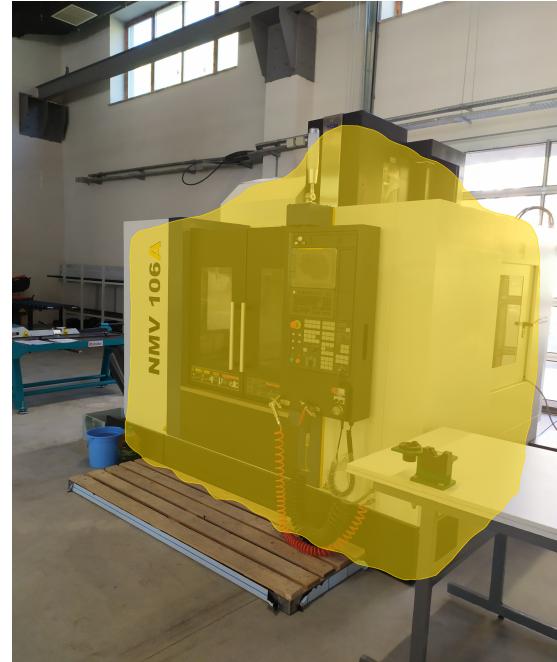


Abb.10 Ergebnisse des Models



Für die CNC-Maschine wurde die Segmentierung ordnungsgemäß durchgeführt, aber sie konnte den Tisch davor nicht vollständig erkennen.

- Route 3 : Hell, direkte Bewegung, dynamisches Bild:

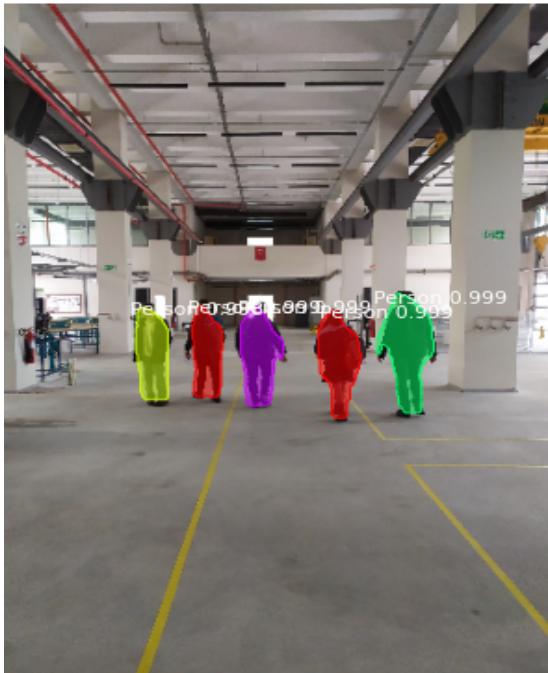


Abb.11 Ergebnisse des Models

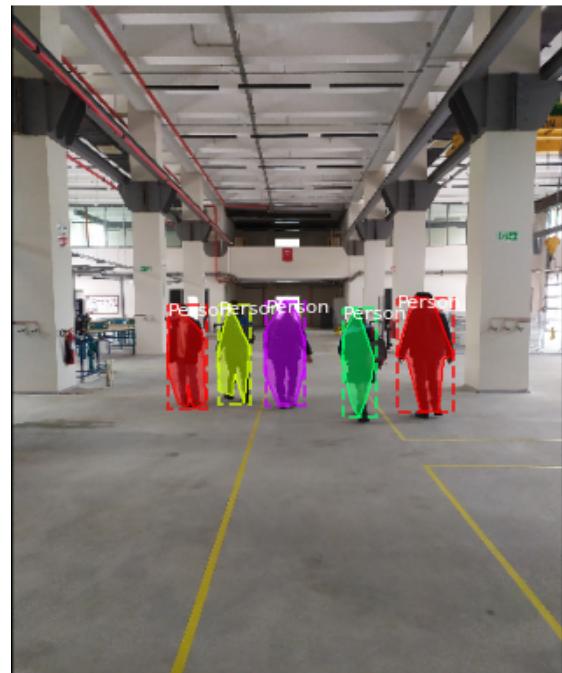


Abb.12 Manuelle markiertes Bild

Bei der Erkennung und Segmentierung des Personenobjekts wurde ein genaues Ergebnis erhalten, und alle fünf Personenobjekte im Bild wurden ohne Verlust erkannt.

Zusammenfassung

TDU-Aluam-Datensatz, dessen Mangel bemerkt wurde, wurde erstellt. Für seine Vorbereitung wurden vorgefertigte Objektklassen von Coco-Mengen genommen. Für unbekannte Objektklassen (Drehen, Fräsen, CNC...) wurde nach der Aufnahme der Fotos und der Beschriftung der notwendige Datensatz erstellt.

Datensätze wurden mit Mask-R-CNN trainiert. Bewertet wurden Themen wie Arbeitsschwierigkeit, Schnelligkeit, Genauigkeit. Er plante auch, Schulungen mit YOLACT durchzuführen.

Wenn Sie die Ergebnisse mit Mask-R-CNN vergleichen und den am besten geeigneten Pfad für die Verwendung im Projekt auswählen möchten. Es wurde der Schluss gezogen, dass YOLACT nicht einmal Ergebnisse lieferte, zu viele Fehler verursachte und für die Studie nicht geeignet war, da keine Lösung gefunden werden konnte.

7.Literaturverzeichnis

[1] *Instance Segmentation Python* Demo*, OpenVino, unter:

https://docs.openvino.ai/latest/omz_demos_instance_segmentation_demo_python.html (abgerufen am 6.11.2022)

[2] Waleed Abdulla(2018), *Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow*

<https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46> (abgerufen am 4.12.2022)

[3] Merve Elif Sarac(2020), *CNN , R-CNN , Fast R-CNN , Mask R-CNN* , unter :

<https://merveelifsarac.medium.com/cnn-r-cnn-fast-r-cnn-mask-r-cnn-c90a1a4d76fb> (abgerufen am 6.11.2022)

[4] *the comparion between FCN and ParseNet output* ,unter:

<https://www.v7labs.com/blog/semantic-segmentation-guide> (abgerufen am 28.10.2022)

[5] Kernzählung und Segmentierung ,unter :

<https://github.com/SUYEGit/Surgery-Robot-Detection-Segmentation> (abgerufen am 25.11.2022)

[6] Maske R-CNN für Chirurgieroboter, unter :

<https://github.com/SUYEGit/Surgery-Robot-Detection-Segmentation> (abgerufen am 22.9.2022)

[7]Ren, H.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 2016, 39, 1137–1149. [Google Scholar] [CrossRef] [PubMed][Green Version] (abgerufen am 16.10.2022)

[8] Hsia, C.-H.; Chang, T.-H.W.; Chiang, C.-Y.; Chan, H.-T. Real-time retail product detection with new augmentation features. In Proceedings of the IEEE International Conference on Electronic Communications, Internet of Things and Big Data, Yilan County, Taiwan, 10–12 December 2021. [Google Scholar] (abgerufen am 4.12.2022)

[9] Daniel Bolya, Chong Zhou, Fanyi Xiao, Yong Jae Lee, YOLACT: Real-time Instance Segmentation, 4 Apr 2019 <https://arxiv.org/abs/1904.02689> (abgerufen am 15.11.2022)

[10] Prakash Jay, Understanding and Implementing Architectures of ResNet and ResNeXt for state-of-the-art Image Classification: From Microsoft to Facebook, 2018. <https://medium.com/@14prakash/understanding-and-implementing-architectures-of-resnet-and-resnext-for-state-of-the-art-image-cf51669e1624> (abgerufen am 22.10.2022)

[11] Jonathan Hui, Understanding Feature Pyramid Networks for object detection (FPN), 27 Mar 2018. unter:<https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c> (abgerufen am 19.10.2022)