WHZ Westsächsische Hochschule Zwickau
Hochschule für Mobilität

Westsächische Hochschule Zwickau

Fakultät Kraftfahrzeugtechnik

Diplomarbeit

Erweiterte Verkehrszählung durch Achsbasierte Fahrzeugklassifikation mithilfe von Instance Segmentation

---

vorgelegt von:	Leonard Kämpf, Matr.-Nr. 41561
eingereicht am:	29. Januar 2026
geboren am:	22. 12. 2001 in Weißenfels
Studiengang:	Kraftfahrzeugtechnik
Studienschwerpunkt:	Kraftfahrzeuge und Mechatronik
zur Erlangung des akademischen Grades:	Diplomingenieur (FH)
Anfertigung im Fachgebiet:	Automatisiertes Fahren und Fahrerassistenzsysteme
	Fakultät für Kraftfahrzeugtechnik
Ausgegeben von:	
Erstbetreuer:	Prof. Dr.-Ing. Rick Voßwinkel
Zweitbetreuer:	Prof. Dr.-Ing. Felix Rudolph

Kurzfassung

In dieser Arbeit wird ein bildbasiertes Verfahren zur erweiterten Verkehrszählung vorgestellt, das über die reine Erfassung von Fahrzeugen hinaus eine achsbasierte Fahrzeugklassifikation ermöglicht. Ziel ist es, Fahrzeuge nicht nur zu detektieren und zu verfolgen, sondern deren Achskonfiguration zuverlässig aus Bilddaten abzuleiten. Hierzu wird eine modulare Verarbeitungspipeline entwickelt, die lernbasierte Detektion, Multi-Object-Tracking und feinaufgelöste Segmentierung kombiniert. Fahrzeuge werden zunächst detektiert und über mehrere Frames verfolgt. Ergänzend werden Reifen mittels textbasierter Detektion und Segmentierung erkannt und in eine eigens entwickelte Logik zur Zuordnung von Rädern zu einzelnen Fahrzeugen integriert. Auf Basis der räumlichen Anordnung der detektierten Räder erfolgt eine achsbasierte Klassifikation. Hierfür wird eine normierte Distanzmetrik definiert, die eine robuste und auflösungsunabhängige Bewertung der Achsabstände ermöglicht. Die Ergebnisse zeigen, dass der entwickelte Ansatz grundsätzlich geeignet ist, eine feinere, strukturbasierte Fahrzeugklassifikation auf Grundlage bildbasierter Daten zu realisieren und klassische Verkehrszählsysteme um zusätzliche Fahrzeugmerkmale zu erweitern.

Abstract

This thesis presents a vision-based approach for extended traffic counting that goes beyond simple vehicle detection by enabling axle-based vehicle classification. The objective is not only to detect and track vehicles, but also to reliably determine their axle configuration from image data. To this end, a modular processing pipeline is developed that combines learning-based detection, multi-object tracking, and fine-grained segmentation. Vehicles are first detected and tracked across consecutive frames. In addition, wheels are identified using text-based detection and segmentation, and integrated into a custom association logic to assign detected wheels to individual vehicles. Based on the spatial arrangement of the detected wheels, an axle-based classification is performed. For this purpose, a normalized distance metric is defined to achieve a robust and resolution-independent evaluation of axle distances. The results demonstrate that the proposed approach is generally suitable for enabling a more fine-grained, structure-based vehicle classification from vision-based data and for extending classical traffic counting systems with additional vehicle-level structural information.

Inhaltsverzeichnis

1 Einleitung

Die Arbeit befasst sich mit der Entwicklung und Validierung eines kamerabasierten Verkehrszählsystems, das klassische Verfahren zur Fahrzeugerkennung mit einer erweiterten Klassifikation anhand sichtbarer Räder kombiniert. Zu Beginn werden die Relevanz der Thematik und die bestehenden technischen Ansätze zur automatisierten Verkehrserfassung dargestellt. Darauf aufbauend werden die methodischen Grundlagen erläutert und geeignete Modelle für Segmentierung und Tracking ausgewählt. Die entwickelten Verfahren werden in einem modularen System implementiert und anhand synthetischer sowie realer Bilddaten auf ihre Zählgenauigkeit und Klassifikationsleistung hin untersucht.

1.1 Motivation

Die effiziente und präzise Erfassung des Verkehrsaufkommens ist eine zentrale Voraussetzung für moderne Mobilitäts- und Infrastrukturplanung. Insbesondere in urbanen Gebieten, an Mautstationen oder auf stark befahrenen Transitstrecken sind verlässliche Verkehrsdaten essenziell für die Optimierung von Verkehrsflüssen, den Ausbau der Infrastruktur sowie für gesetzliche und wirtschaftliche Entscheidungen. Klassische Verkehrszählsysteme, wie Induktionsschleifen oder kamerabasierte Objekterkennungsverfahren, liefern dabei zwar grundlegende Informationen über Anzahl und Typen von Fahrzeugen, stoßen jedoch bei der feingranularen Klassifikation an ihre Grenzen.
Mit dem Fortschritt im Bereich des Deep Learning und der Bildverarbeitung haben sich insbesondere Methoden der Instance Segmentation für verkehrsbezogene Aufgaben etabliert. Diese ermöglichen nicht nur die pixelgenaue Segmentierung, sondern auch die individuelle Erkennung und Verfolgung einzelner Objekte wie Fahrzeuge oder Reifen. Dennoch bleibt die Unterscheidung zwischen äußerlich ähnlichen Fahrzeugen, beispielsweise zwischen einem Lkw mit Einzel-, Doppel- oder Dreifachachse, eine Herausforderung.

Eine vielversprechende Möglichkeit zur Verbesserung der Klassifikationsgenauigkeit liegt in der Erweiterung bestehender Instance-Segmentation-Modelle um die gezielte Erkennung einzelner Fahrzeugkomponenten insbesondere der Reifen. Durch die Analyse von Anzahl und Position der Reifen kann auf die Achskonfiguration eines Fahrzeugs geschlossen werden, was eine feingranularere und strukturorientierte Fahrzeugklassifikation erlaubt. Diese geht über die herkömmliche Einteilung in grobe Kategorien wie Pkw, Lkw oder Bus hinaus und erlaubt eine differenzierte Bewertung der tatsächlichen Fahrzeugstruktur.
Ziel dieser Arbeit ist es daher, ein Verfahren zu entwickeln, das bestehende Instance-Segmentation-Modelle erweitert, um neben der Fahrzeugerkennung auch eine robuste Reifendetektion zu ermöglichen. Basierend auf den detektierten Reifen soll anschließend die jeweilige Achskonfiguration bestimmt und der zugehörigen Fahrzeuginstanz zugeordnet werden.
Mit diesem Ansatz wird ein Beitrag zur Weiterentwicklung intelligenter Verkehrszählsysteme geleistet, die nicht nur zuverlässiger, sondern auch informativer sind als konventionelle Verfahren. Die Arbeit adressiert ein relevantes Problemfeld im Kontext der digitalen Verkehrsdatenerhebung und eröffnet neue Perspektiven für Anwendungen in Logistik, Infrastrukturplanung, automatisierter Verkehrssteuerung und Mautsystemen.

1.2 Problemstellung

Die präzise Erfassung und Klassifikation von Fahrzeugen ist eine zentrale Anforderung in vielen Bereichen der Verkehrstechnik, Logistik und öffentlichen Infrastruktur. Insbesondere bei der automatisierten Verkehrszählung gewinnen bildbasierte Verfahren zunehmend an Bedeutung. Moderne Deep-Learning-Methoden zur Instance Segmentation ermöglichen dabei die pixelgenaue Erkennung und Abgrenzung einzelner Objekte in einem Bild, was eine differenzierte Analyse von Verkehrsszenen erlaubt. Trotz dieser Fortschritte bestehen nach wie vor grundlegende Herausforderungen, wenn es darum geht, Fahrzeuge nicht nur zu detektieren, sondern auch hinsichtlich ihrer spezifischen baulichen Merkmale zu klassifizieren insbesondere in Bezug auf die Achskonfiguration. Die Anzahl der Achsen eines Fahrzeugs hat direkten Einfluss auf Mautsysteme, die Belastung von Infrastrukturelementen wie Brücken und Fahrbahndecken sowie auf verkehrsplanerische und ökologische Bewertungen. Während konventionelle Klassifikationen häufig nur zwischen Fahrzeugtypen wie Pkw, Lkw oder Bus unterscheiden, ist in vielen realen Anwendungen eine feingranularere Kategorisierung notwendig beispiels-

weise eine Trennung zwischen Lkw mit Einzel-, Doppel- oder Dreifachachsen. Diese Unterscheidung ist visuell häufig nicht allein anhand der äußeren Fahrzeugform möglich und erfordert die Analyse der zugrundeliegenden Achsstruktur.
Ein zentrales Problem dabei ist die zuverlässige Detektion der Fahrzeugreifen. Die Reifen sind visuelle Indikatoren für die Achsanzahl, jedoch häufig nur teilweise sichtbar, verdeckt oder perspektivisch verzerrt. Hinzu kommt, dass mehrere Fahrzeuge gleichzeitig im Bild sein können, was die korrekte Zuordnung erkannter Reifen zu den jeweiligen Fahrzeuginstanzen erschwert. Die reine Objektsegmentierung reicht daher nicht aus. Es wird zusätzlich ein robustes Zuordnungsverfahren benötigt, das Kontextinformation über Position, Bewegung und Zusammengehörigkeit berücksichtigt.
Diese Arbeit adressiert genau dieses Problemfeld: Ziel ist die Entwicklung eines Verfahrens, das in der Lage ist, aus Bilddaten nicht nur Fahrzeuge als Instanzen zu erkennen, sondern auch deren Reifen zu detektieren, die Achskonfiguration abzuleiten und diese der jeweiligen Fahrzeuginstanz korrekt zuzuordnen. Dazu sollen bestehende Methoden der Instance Segmentation gezielt erweitert und mit Tracking-Komponenten kombiniert werden, um auch unter realen Bedingungen, wie wechselnden Perspektiven, Verdeckungen und Verkehrsdichte, zuverlässige Ergebnisse zu liefern.
Die daraus resultierende Systemarchitektur soll es ermöglichen, über eine reine Fahrzeugzählung hinauszugehen und eine strukturorientierte Klassifikation auf Basis physikalischer Merkmale vorzunehmen. Dies schafft die Grundlage für intelligentere, datengetriebene Verkehrsanalyseverfahren mit hoher praktischer Relevanz.

1.3 Stand der Technik

Der Stand der Technik beleuchtet bestehende Ansätze zur automatisierten Verkehrszählung und -klassifikation mit Fokus auf deren technologischen Aufbau und Einsatzgrenzen. Neben einer Unterteilung in intrusive und nicht-intrusive Systeme werden etablierte Sensorprinzipien nach [1] beschrieben. Darüber hinaus wird ein Überblick über aktuelle Verfahren der bildbasierten Objektdetektion gegeben, insbesondere mit Blick auf deren Eignung für segmentierungsbasierte Zähllogiken im Verkehrsbereich. Diese Analyse dient als Grundlage für die Auswahl der im weiteren Verlauf eingesetzten Modelle.

1.3.1 Klassische Verfahren zur Verkehrszählung

Klassische Verkehrszählsysteme lassen sich grundsätzlich in intrusive und nicht-intrusive Technologien unterteilen. Erstere erfordern eine bauliche Integration von Sensoren in die Fahrbahnoberfläche, etwa durch Einschnitte, Bohrungen oder Tunnelungen. Diese Systeme bieten in der Regel eine hohe Messgenauigkeit, sind jedoch mit erheblichem Aufwand sowie hohen Kosten für Installation, Wartung und Reparatur verbunden. Nicht-intrusive Verfahren hingegen erfordern keine Eingriffe in die Fahrbahn und lassen sich mit deutlich geringerem Verkehrsaufwand installieren und warten. Die Sensorik ist bei diesen Systemen meist oberhalb der Fahrbahn, etwa an Masten oder Brücken, oder seitlich positioniert.

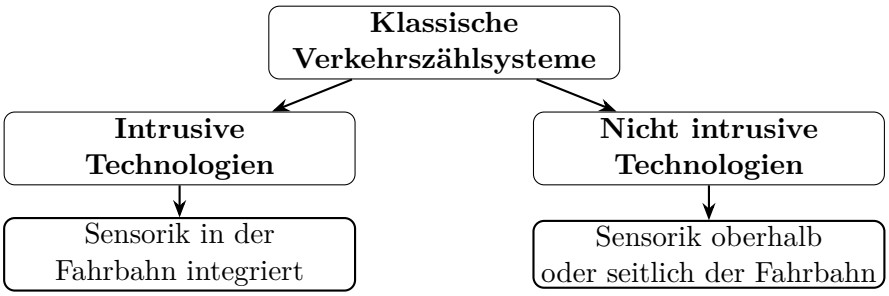


Abbildung 1.1: Grundlegende Unterteilung klassischer Verkehrszählsysteme in intrusive und nicht intrusive Technologien

Zu den am weitesten verbreiteten Systemen zählen Induktionsschleifen. Dabei werden Drahtschleifen in den Straßenbelag eingelassen, die ein Magnetfeld im Bereich von etwa 10–50 kHz erzeugen. Fährt ein Fahrzeug über die Schleife, verändert es die Induktivität und löst dadurch ein Detektionssignal aus. Die Technologie erlaubt nicht nur eine präzise Fahrzeugerkennung, sondern auch Geschwindigkeitsmessungen (z. B. mithilfe von Doppelschleifen) sowie eine rudimentäre Klassifikation auf Basis charakteristischer Signaturen. Induktionsschleifen gelten als Industriestandard für dauerhafte Detektion, weisen jedoch eine erhöhte Ausfallrate auf, insbesondere durch Straßenschäden, Temperaturbelastung und mechanische Beanspruchung.
Pneumatische Zählschläuche bestehen aus quer zur Fahrbahn ausgelegten Gummischläuchen. Wird der Schlauch durch ein Rad überfahren, entsteht ein Luftdruckimpuls, der über einen Schalter in ein elektrisches Signal umgewandelt wird. Diese Technik ist energiesparend und lässt sich schnell installieren, weshalb sie primär für temporäre Verkehrszählungen oder Achsanalysen eingesetzt wird. Ihre Genauigkeit nimmt bei hohem Verkehrsaufkommen ab, insbesondere bei schweren Fahrzeugen. Zudem reagieren diese

Systeme empfindlich auf Temperaturschwankungen und sind stark verschleißanfällig. Piezoelektrische Sensoren, beispielsweise aus Quarzmaterialien, erzeugen bei Krafteinwirkung eine elektrische Spannung. Sie werden zur Fahrzeugdetektion, Achszählung, Geschwindigkeitsmessung sowie zur Fahrzeugklassifikation eingesetzt. Im Vergleich zu pneumatischen Systemen erfordern sie einen deutlich höheren Installations- und Kalibrieraufwand, bieten jedoch auch eine höhere Präzision. Piezoelektrische Sensoren eignen sich zudem für Weigh-in-Motion-Systeme, bei denen Fahrzeuggewichte im fließenden Verkehr gemessen werden.
Weigh-in-Motion-Systeme ermöglichen eine kontinuierliche Gewichtserfassung von Fahrzeugen. Dabei werden sowohl Achslasten als auch das Gesamtgewicht und der Fahrzeugtyp bestimmt. Diese Systeme sind hochkomplex, benötigen eine präzise Kalibrierung und ihre Messgenauigkeit hängt stark von Umwelteinflüssen wie Straßentemperatur, Geschwindigkeit und Feuchtigkeit ab. Sie werden insbesondere zur Überwachung von Nutzlastgrenzen im Schwerlastverkehr eingesetzt.

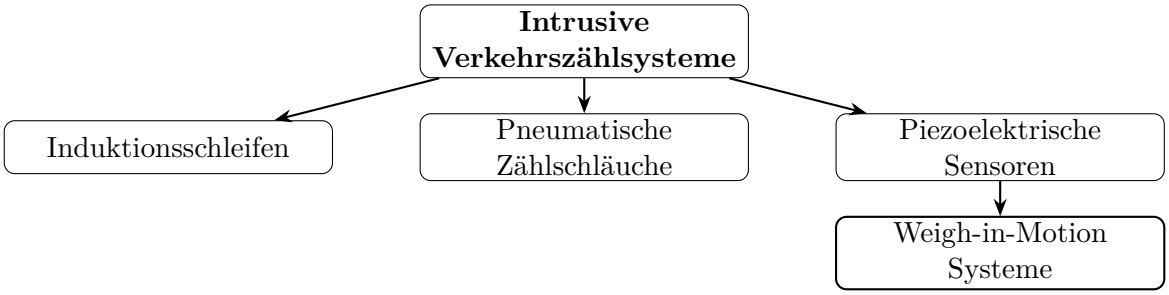


Abbildung 1.2: Typische intrusive Verkehrszählsysteme zur Fahrzeugdetektion, Achszählung und Gewichtserfassung

Mikrowellenradarsysteme arbeiten im Frequenzbereich von 1–30 GHz und erfassen Fahrzeuge durch Reflexion elektromagnetischer Wellen. Sie sind witterungsresistent und werden in unterschiedlichen Varianten eingesetzt. CW-Doppler-Radare senden eine konstante Frequenz aus, deren Dopplerverschiebung zur Geschwindigkeitsschätzung genutzt wird. Diese Methode erkennt jedoch keine stehenden Fahrzeuge. FMCW-Radare hingegen nutzen Laufzeitdifferenzen eines frequenzmodulierten Signals zur Abstandsmessung und können somit auch stationäre Fahrzeuge erfassen.
Infrarotbasierte Systeme nutzen elektromagnetische Strahlung im nahen oder fernen Infrarotbereich zur Detektion. Aktive IR-Systeme senden gezielte Infrarotstrahlung aus und detektieren deren Reflexion an Fahrzeugen. Passive Systeme hingegen basieren auf der Eigenstrahlung von Objekten und können sowohl punktuell als auch bildgebend

arbeiten. Trotz ihres wartungsarmen Aufbaus sind sie anfällig gegenüber Umwelteinflüssen wie Sonnenlicht, Regen oder Nebel.
Auch Ultraschallsysteme werden zur Fahrzeugdetektion eingesetzt. Sie senden Schallwellen im Bereich von 25–50 kHz aus und ermitteln den Abstand zu Objekten über die Laufzeitmessung. Wird eine kürzere Distanz als die zur Straßenoberfläche gemessen, erfolgt eine Fahrzeugdetektion. Die Systeme sind jedoch wind- und temperaturanfällig, was ihre Genauigkeit einschränkt.

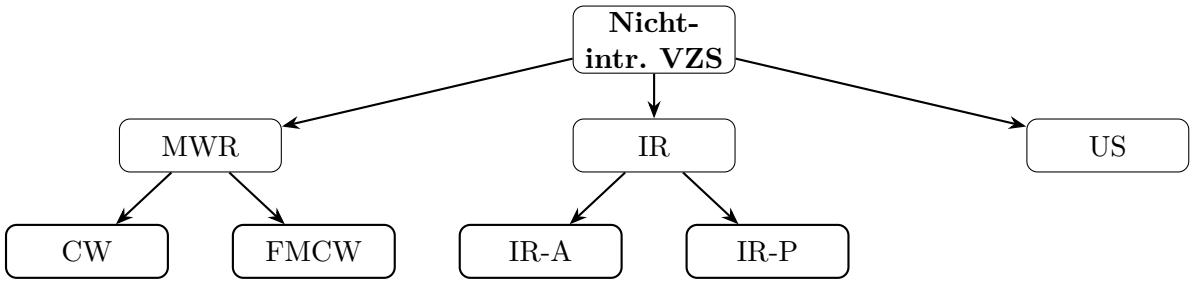


Abbildung 1.3: Nicht intrusive Verkehrszählsysteme (VZS). MWR: Mikrowellenradar, CW: Continuous-Wave-Doppler-Radar, FMCW: Frequency-Modulated Continuous-Wave-Radar, IR: Infrarotsysteme, IR-A: Aktive Infrarotsysteme, IR-P: Passive Infrarotsysteme, US: Ultraschallsysteme

1.3.2 Bildbasierte Zählmethoden in der Verkehrsüberwachung

Vision-basierte Systeme, die mit handelsüblichen RGB- oder Infrarotkameras arbeiten, bieten eine kostengünstige und flexible Alternative zu klassischen Verfahren. Sie ermöglichen die Detektion, Zählung und Verfolgung von Fahrzeugen direkt aus Videobildern und lassen sich in bestehende CCTV-Infrastrukturen (Closed-Circuit-Television) integrieren. Da sie keine Eingriffe in die Fahrbahn erfordern, sind sie einfacher zu installieren und zu warten.
Ein etablierter Ansatz, wie im Paper Vehicle detection, tracking and classification in urban traffic [2] vorgestellt, kombiniert mehrere klassische Bildverarbeitungsverfahren. Zunächst erfolgt die Hintergrundsubtraktion mittels Gaussian Mixture Model (GMM), bei dem jeder Bildpunkt als Mischung mehrerer Gaußverteilungen modelliert wird. Neue Pixelwerte werden mit diesem Modell verglichen, wobei signifikante Abweichungen auf Vordergrundobjekte hinweisen. Dank adaptiver Lernrate passt sich GMM auch bei wechselnden Lichtverhältnissen zuverlässig an.

Auf die resultierende binäre Maske werden morphologische Operationen angewandt, um Störungen zu entfernen und die Objektstruktur zu verbessern. Dazu zählen unter anderem Erosion, Dilation sowie Kombinationen wie Opening und Closing. Anschließend werden mittels BLOB-Analyse zusammenhängende Regionen identifiziert. Das Seitenverhältnis der Objekte erlaubt dabei eine grobe Unterscheidung zwischen Fußgängern und Fahrzeugen.
Zur Verfolgung detektierter Objekte kommt ein Kalman-Filter [3] zum Einsatz. Dieser prognostiziert die Bewegung eines Fahrzeugs auf Basis vorheriger Positionen und gleicht auch kurzfristige Unterbrechungen innerhalb der Region of Interest (ROI) aus. Die Datenassoziation zwischen Detektionen und bestehenden Spuren wird mithilfe des Kuhn-Munkres-Algorithmus [4] (auch Ungarischer Algorithmus genannt) vorgenommen. Hierbei wird eine Kostenmatrix erstellt, typischerweise basierend auf Distanz- oder Ähnlichkeitsmaßen, und eine optimale Zuordnung berechnet. Gezählt wird ein Fahrzeug nur dann, wenn es die definierte ROI betritt. Diese regelbasierte Ereigniszählung verringert die Rechenlast und minimiert Fehlzählungen. Trotz ihrer Robustheit stoßen solche klassischen Methoden bei komplexeren Aufgaben wie der achsbasierten Fahrzeugklassifikation an ihre Grenzen. Ohne semantische oder strukturelle Segmentierungsinformationen lassen sich Achsanzahl und -abstände nicht zuverlässig bestimmen. Klassische bildbasierte Verfahren bieten somit eine bewährte Grundlage für einfache Zähl- und Trackingaufgaben. Ihre Leistungsfähigkeit ist jedoch begrenzt, wenn es um feinere Objektklassifikationen geht. Im folgenden Kapitel wird untersucht, wie moderne Deep-Learning-Ansätze diese Lücken durch lernbasierte Detektion und Segmentierung schließen können.

1.3.3 Deep Learning für Detektion und Segmentierung im Verkehrsbereich

Die klassischen bildbasierten Verfahren zur Fahrzeugerkennung stoßen bei komplexen Szenen mit überlappenden Objekten, stark variierenden Lichtverhältnissen oder bei Teilokklusionen an ihre Grenzen. Deep-Learning-Ansätze bieten hier eine robuste Alternative, da sie in der Lage sind, visuelle Merkmale automatisch aus großen Datenmengen zu extrahieren und dabei komplexe Zusammenhänge zu lernen [5]. Besonders Convolutional Neural Networks (CNNs) haben sich im Bereich der Objektdetektion etabliert.

Moderne Modelle wie MMSFormer [6] oder OneFormer [7] basieren auf Vision Transformers und gelten als aktuelle Entwicklung im Bereich der bildbasierten Verkehrsüberwachung. Sie ermöglichen eine robuste und kontextbasierte Objekterkennung auch in komplexen Szenen mit hohem Verkehrsaufkommen. Aufgrund ihrer Architektur können sie globale Bildkontexte besser erfassen als konventionelle CNNs und liefern dadurch zuverlässige Ergebnisse bei schwierigen Bedingungen wie Teilverdeckungen, Schattenwurf oder dichter Verkehr. Ihre hohe Generalisierungsfähigkeit macht sie besonders attraktiv für den Einsatz in realen Verkehrsumgebungen. Zur Objektdetektion werden üblicherweise einstufige Detektoren wie YOLO [8] oder SSD [9] sowie zweistufige Architekturen wie Faster R-CNN [10] eingesetzt. Während zweistufige Verfahren eine höhere Genauigkeit bei geringer Bildrate liefern, sind einstufige Ansätze für Echtzeit-Anwendungen im Verkehrsbereich besser geeignet. Durch den Einsatz vortrainierter Modelle und Transfer Learning können auch kleinere Datensätze effizient genutzt werden.

Neben der reinen Detektion gewinnt die Segmentierung zunehmend an Bedeutung. Semantic Segmentation erlaubt die pixelgenaue Klassifizierung von Flächen in einem Bild, ohne jedoch zwischen verschiedenen Objektinstanzen zu unterscheiden [11]. Typische Modelle hierfür sind nnU-Net [12] [13] oder DeepLab [14]. Für Anwendungen, bei denen die Unterscheidung einzelner Fahrzeuge erforderlich ist, beispielsweise bei der Fahrzeugzählung oder bei instanzbasierter Klassifikation, wird Instance Segmentation eingesetzt. Mask R-CNN [5] ist ein etabliertes Modell, das Detektion und Instance Segmentation kombiniert. Neuere Modelle wie OneFormer verfolgen einen universellen Ansatz zur Segmentierung und vereinen Semantic, Instance und Panoptic Segmentation in einer einheitlichen Architektur. Noch weitergehend nutzt das Segment Anything Model 2 (SAM2) [15] ein Prompt-basiertes Paradigma zur flexiblen Segmentierung beliebiger Objekte und stellt damit ein starkes Werkzeug für bildbasierte Aufgaben in offenen Szenarien dar.

Trotz dieser Fortschritte bestehen Herausforderungen: Deep-Learning-Methoden sind datenhungrig, erfordern hohe Rechenleistung und können bei domänenspezifischen Abweichungen in den Eingabedaten ihre Generalisierungsfähigkeit verlieren. Zudem sind Fehlklassifikationen in sicherheitskritischen Anwendungen nicht tolerierbar, weshalb eine fundierte Validierung und Optimierung notwendig ist.

Im folgenden Abschnitt werden Trackingverfahren vorgestellt, die es ermöglichen, Detektionen über mehrere Frames hinweg zu einem konsistenten Objektverlauf zusammenzuführen.

1.3.4 Trackingverfahren in der Verkehrsbeobachtung

Das Ziel von Trackingverfahren in der Verkehrsüberwachung besteht darin, erkannte Objekte über mehrere aufeinanderfolgende Bilder hinweg eindeutig zu identifizieren und ihre Bewegungsverläufe zu rekonstruieren. Dies ist essenziell für eine zuverlässige Zählung, zur Vermeidung von Mehrfachzählungen und als Grundlage für weiterführende Analysen wie die Bestimmung von Geschwindigkeit, Fahrtrichtung oder Fahrzeugklassifikation [9].
Klassische Trackingverfahren basieren häufig auf Kalman-Filtern, die unter der Annahme eines linearen Bewegungsmodells die zukünftige Position eines Objekts prognostizieren. Diese Methode ist besonders effizient bei stabiler Bildrate und geringem Bewegungsrauschen. Für komplexere Bewegungsprofile oder bei stark variierender Bildqualität kann zusätzlich der optische Fluss (z. B. Lucas-Kanade) verwendet werden. Weitere einfache Methoden wie Mean-Shift oder CamShift eignen sich nur bedingt für Verkehrsszenarien mit mehreren sich überlagernden Objekten [16].
Für das Tracking mehrerer Objekte (Multi-Object Tracking, MOT) [17] ist eine zuverlässige Datenassoziation notwendig. Hierbei wird jede neue Detektion einem bestehenden Track zugeordnet. Ein verbreitetes Verfahren zur Lösung dieses Optimierungsproblems ist der Kuhn-Munkres-Algorithmus, der auf einer Kostenmatrix basiert. Die Kosten können etwa durch den Abstand der Bounding Boxen oder visuelle Ähnlichkeit definiert werden.
Mit dem Aufkommen tiefer neuronaler Netze haben sich auch Deep-Learning-gestützte Trackingverfahren etabliert. Deep SORT erweitert das klassische SORT-Verfahren um ein Appearance-Modell auf Basis von CNNs. ByteTrack verfolgt einen anderen Ansatz, indem auch Detektionen mit geringen Confidence Scores in die Datenassoziation einbezogen werden, wodurch die Robustheit des Trackings insbesondere bei zeitweiligem Okklusion oder schwacher Erkennung steigt.
Für videobasierte Segmentierungs- und Trackingaufgaben existieren moderne Architekturen wie TrackFormer [18] oder Track Anything [19], die gezielt auf konsistentes Masken-Tracking ausgerichtet sind. In verkehrstechnischen Anwendungen kommen sie derzeit vor allem in der Forschung zum Einsatz.
Trotz dieser Fortschritte sind Trackingverfahren anfällig für Fehler bei stark ähnlichen Objekten, schnellen Richtungswechseln oder kurzen Okklusionen. Eine sorgfältige Parametrierung und Kombination mit robusten Detektoren sind daher essenziell.
Die Kombination aus Detektion, Segmentierung und Tracking bildet die Grundlage für komplexere Verkehrsanalyseverfahren.

2 Theoretische Grundlagen

Dieses Kapitel legt das methodische Fundament für die im weiteren Verlauf dieser Arbeit entwickelte Systemarchitektur. Es stellt die wesentlichen Konzepte der Objekterkennung, Segmentierung und Objektverfolgung vor, die für die Realisierung einer bildbasierten, achsbasierten Verkehrszählung unerlässlich sind. Dabei werden die Unterschiede zwischen Semantic und Instance Segmentation herausgearbeitet sowie zentrale Deep-Learning-Ansätze zur Detektion und Segmentierung erläutert. Anschließend erfolgt ein Überblick über bestehende Systeme zur Verkehrserfassung, insbesondere im Hinblick auf deren Sensortechnologien und Anwendungsgrenzen. Abschließend werden etablierte Bewertungsmetriken eingeführt, anhand derer die Leistungsfähigkeit der eingesetzten Segmentierungs- und Trackingverfahren im späteren Evaluationskapitel beurteilt wird. Ziel dieses Kapitels ist es, die für die Systementwicklung relevanten Grundlagen systematisch aufzuarbeiten und in den anwendungsbezogenen Kontext der verkehrstechnischen Bildverarbeitung einzuordnen.

2.1 Grundlagen der Objekterkennung und Segmentierung

Die zuverlässige Erkennung und differenzierte Analyse von Objekten in digitalen Bildern bildet eine zentrale Grundlage der verkehrstechnischen Bildverarbeitung. Für die in dieser Arbeit angestrebte achsbasierte Klassifikation von Fahrzeugen sind dabei insbesondere die Verfahren der Instance Segmentation und Semantic Segmentation von Bedeutung. Dieses Unterkapitel stellt die theoretischen und technischen Grundlagen beider Segmentierungsansätze vor und erläutert deren Funktionsprinzipien sowie typische Architekturen. Ergänzend werden Verfahren zur Objektverfolgung eingeführt, die für zeitlich konsistente Klassifikationen und Zählvorgänge im Rahmen einer videobasierten Verkehrsdatenerfassung relevant sind. Ziel ist es, das methodische Fundament für die im späteren Verlauf eingesetzten Modelle und Systemkomponenten zu legen.

2.1.1 Semantic Segmentation vs. Instance Segmentation

Die Segmentierung gehört zu den grundlegenden Aufgaben in der computergestützten Bildverarbeitung. Ziel ist es, ein digitales Bild in sinnvolle Segmente zu unterteilen, indem zusammenhängende Pixelbereiche mit ähnlichen visuellen Eigenschaften gruppiert werden. Die daraus resultierende Struktur erleichtert die Analyse und Interpretation visueller Daten und bildet die Grundlage für viele weiterführende Verfahren der Bildauswertung [11].
Ein typisches Segmentierungskriterium ist die Ähnlichkeit benachbarter Pixel hinsichtlich Farbe, Helligkeit oder Textur. Bei der regionenbasierten Segmentierung werden angrenzende Bildbereiche identifiziert, die sich visuell ähneln. Alternativ kann auch die Kanteninformation genutzt werden, um scharfe Übergänge zwischen verschiedenen Objekten oder Bildbereichen zu erkennen.
Die Semantic Segmentation geht über diese Konzepte hinaus. Hier wird jedem einzelnen Pixel eines Bildes eine semantische Klasse zugewiesen, beispielsweise „Fahrzeug“, „Straße“ oder „Gebäude“. Ziel ist es, die gesamte Bildfläche in inhaltlich beschriebene Regionen zu unterteilen. Dabei unterscheidet Semantic Segmentation nicht zwischen verschiedenen Instanzen einer Klasse; alle erkannten Pixel der Klasse „Fahrzeug“ gehören zur selben Kategorie, unabhängig davon, ob sich mehrere Fahrzeuge im Bild befinden.
Modelle für Semantic Segmentation basieren in der Regel auf Deep Learning, insbesondere auf Convolutional Neural Networks (CNNs), die als Encoder-Decoder-Architekturen ausgelegt sind. Bekannte Ansätze sind Fully Convolutional Networks (FCNs), U-Net [13] oder DeepLab [14]. Sie bestehen aus einem kontrahierenden Pfad zur Merkmalsextraktion und einem expandierenden Pfad zur rekonstruktiven Pixelklassifikation. Trainiert werden diese Netzwerke auf umfangreichen, annotierten Bilddatensätzen mit pixelgenauer Labelinformation.
Im Unterschied dazu verfolgt die Instance Segmentation einen differenzierteren Ansatz. Sie weist nicht nur jedem Pixel eine Klasse zu, sondern unterscheidet zusätzlich zwischen einzelnen Objekten derselben Kategorie. So erkennt das System beispielsweise mehrere Fahrzeuge im Bild und weist jedem eine eigene Instanzkennung zu. Diese Methode ist insbesondere für Anwendungen relevant, bei denen es auf das Zählen, Verfolgen oder individuell differenzierte Analysieren von Objekten ankommt. Etwa in der Verkehrsüberwachung, bei medizinischen Zellanalysen oder in der automatisierten Inventarverwaltung.

Technisch verbindet Instance Segmentation zwei Aufgaben: die Objektdetektion (z. B. über Bounding Boxes) und die pixelweise Segmentierung. Ein prominenter Ansatz ist Mask R-CNN [5], der auf Faster R-CNN [20] basiert und zusätzlich pro erkannten Bereich (Region of Interest) eine binäre Maske zur präzisen Segmentierung erzeugt. Weitere Verfahren wie YOLO11 [21] zielen auf besonders schnelle und robuste Anwendung in realen Szenarien ab.
Im Vergleich zur Semantic Segmentation ist die Instance Segmentation deutlich komplexer: Sie erfordert mehr Trainingsdaten, höhere Rechenressourcen und aufwendigere Modellarchitekturen. Dafür bietet sie eine erheblich genauere Bildanalyse, insbesondere bei überlappenden Objekten oder bei der Notwendigkeit, einzelne Elemente exakt zu identifizieren.
Beide Verfahren haben ihre spezifischen Stärken. Semantic Segmentation eignet sich vor allem zur globalen Analyse von Szenen, bei denen der strukturelle Aufbau des Bildes im Vordergrund steht. Instance Segmentation hingegen liefert detailreichere Informationen auf Objektebene und wird überall dort eingesetzt, wo exakte Zählung und Individualisierung erforderlich sind. In modernen Anwendungen wie autonomen Fahrsystemen, Smart Cities oder der präzisen landwirtschaftlichen Analyse werden beide Methoden zunehmend kombiniert, um sowohl die semantische Struktur als auch die konkreten Objektinstanzen zuverlässig zu erfassen.

2.1.2 Objektverfolgung (Tracking)

Kalman-Filter Der Kalman-Filter ist ein rekursiver, linearer Zustandsschätzer zur optimalen Fusion von Messdaten und Systemmodellen unter Annahme von gaußverteiltem Prozess- und Messrauschen. Er wird eingesetzt, um den Zustand eines dynamischen Systems, beispielsweise die Position und Geschwindigkeit eines Fahrzeugs, aus verrauschten Beobachtungen zu schätzen und vorherzusagen. Der Filter besteht aus zwei Hauptschritten: der Prädiktion und der Korrektur. In der Prädiktionsphase wird der aktuelle Systemzustand mithilfe eines Bewegungsmodells in die Zukunft extrapoliert. In der anschließenden Korrekturphase wird diese Vorhersage mit einer neuen Messung kombiniert, wobei die Gewichtung durch die jeweilige Unsicherheit bestimmt wird. Dadurch ermöglicht der Kalman-Filter eine Glättung verrauschter Messungen sowie eine robuste Schätzung auch bei kurzzeitigen Detektionsausfällen. In Tracking-Anwendungen bildet er die Grundlage für die Bewegungsmodellierung und die Vorhersage von Objektpositionen zwischen aufeinanderfolgenden Frames.

Kuhn-Munkres-Algorithmus Der Kuhn-Munkres-Algorithmus ist ein effizienter Algorithmus zur Lösung des Zuordnungsproblems (Assignment Problem) in bipartiten Graphen. Ziel ist es, eine optimale Eins-zu-Eins-Zuordnung zwischen zwei Mengen zu bestimmen, sodass die Gesamtkosten minimiert werden. In der Objektverfolgung wird der Algorithmus typischerweise eingesetzt, um aktuelle Detektionen optimal bestehenden Objektspuren zuzuordnen. Hierzu wird eine Kostenmatrix definiert, deren Einträge die Ähnlichkeit oder Distanz zwischen vorhergesagten Objektpositionen und aktuellen Detektionen repräsentieren, beispielsweise basierend auf euklidischer Distanz, Intersection-over-Union (IoU) oder Mahalanobis-Distanz. Der Ungarische Algorithmus berechnet anschließend in polynomieller Zeit eine globale optimale Zuordnung, wodurch lokale Fehlentscheidungen vermieden werden. Aufgrund seiner Robustheit und Effizienz ist er ein Standardverfahren in Multi-Object-Tracking-Systemen.

2.2 Überblick über bestehende Verkehrszählungsansätze

Zur Erfassung des Verkehrsaufkommens wurden in den vergangenen Jahrzehnten verschiedene Technologien und Methoden entwickelt. Klassische Ansätze beruhen auf stationären Sensoren wie Induktionsschleifen, piezoelektrischen Sensoren oder Radarsystemen. Induktionsschleifen werden im Fahrbahnbelag installiert und registrieren die Veränderung des elektromagnetischen Feldes beim Überfahren durch ein Fahrzeug. Sie ermöglichen eine präzise Erfassung der Fahrzeuganzahl sowie eine grobe Klassifikation nach Fahrzeuglänge und Geschwindigkeit [22].
Auch Radarsensoren und Infrarotsysteme werden in stationären Anlagen eingesetzt, um Fahrzeuge anhand ihrer Reflektionseigenschaften zu erkennen. Diese Systeme sind besonders robust gegenüber Umwelteinflüssen, liefern jedoch meist nur aggregierte Daten und keine Information über Fahrzeugstruktur oder Achsanzahl [23].
In den letzten Jahren haben sich bildbasierte Systeme, insbesondere kamerabasierte Verkehrszählung, als vielversprechende Alternative etabliert. Hierbei werden Bilddaten in Echtzeit ausgewertet, um Fahrzeuge zu detektieren und zu klassifizieren. Erste Systeme basierten auf Bewegungserkennung oder optischem Fluss, moderne Lösungen verwenden jedoch überwiegend Deep-Learning-Methoden zur Objekterkennung und Segmentierung.
Verfahren wie YOLO11, Faster R-CNN oder Mask R-CNN ermöglichen eine präzise Erkennung und Unterscheidung einzelner Fahrzeuge. In Kombination mit Multi-

Object-Tracking-Algorithmen (z. B. DeepSORT oder ByteTrack [24], [25]) lassen sich Zählungen, Fahrzeugklassifikationen und Trajektorienanalysen durchführen.
Für großflächige Erhebungen kommen auch UAVs (Unmanned Aerial Vehicles) mit hochauflösenden Kameras zum Einsatz. Diese ermöglichen eine temporär flexible, großflächige Erhebung in städtischen oder ländlichen Bereichen und finden zunehmend Anwendung in Forschungsprojekten [26].

2.3 Bewertungskriterien

Die Leistungsbewertung von Segmentierungsverfahren erfolgt anhand verschiedener Metriken, die je nach Segmentierungsaufgabe unterschiedliche Aspekte der Vorhersagequalität erfassen. Während bei der Semantic Segmentation insbesondere die Übereinstimmung zwischen vorhergesagten und tatsächlichen Klassenmasken im Vordergrund steht, fließen bei der Instance und Panoptic Segmentation zusätzlich Faktoren wie Objektlokalisierung und -trennung in die Bewertung ein.
Zu den zentralen Kennzahlen zählen die Mean Intersection over Union (mIoU) zur Messung der Flächenüberdeckung, die Average Precision (AP) zur Beurteilung von Detektion und Instanzzuordnung sowie die Panoptic Quality (PQ), welche beide Aspekte in einer gemeinsamen Metrik vereint. Diese Metriken erlauben eine differenzierte Analyse der Modellleistung und sind in aktuellen Benchmarks wie Cityscapes [27], COCO [28] oder ADE20K [29] weit verbreitet.
Die Mean Intersection over Union [30], auch als Jaccard-Koeffizient bekannt, ist eine etablierte Metrik zur Bewertung der Vorhersagequalität in allen Segmentierungsarten. Sie beschreibt die Überlappung zwischen den vorhergesagten Segmenten und den zugehörigen Ground-Truth-Labels. Für jede Klasse i wird das Verhältnis der Schnittmenge zur Vereinigungsmenge der vorhergesagten (P_i) und tatsächlichen (G_i) Pixel berechnet:

$$\mathrm{IoU}_i = \frac{|P_i \cap G_i|}{|P_i \cup G_i|} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{2.1}$$

Dabei stehen True Positives (TP) für korrekt vorhergesagte Pixel einer Klasse, False Positives (FP) für fälschlich zugewiesene Pixel und False Negatives (FN) für nicht erkannte Pixel. Die mittlere IoU über alle C Klassen ergibt sich durch

$$\mathrm{mIoU} = \frac{1}{C} \sum_{i=1}^{C} \mathrm{IoU}_i. \tag{2.2}$$

Ein Wert von 1,0 signalisiert perfekte Übereinstimmung, während 0,0 keine Überlappung bedeutet. In der Praxis erreichen leistungsfähige Modelle mIoU-Werte zwischen 0,5 und 0,9, abhängig von der Komplexität der Daten und der Architektur des Modells. Im Vergleich zur Pixelgenauigkeit ist die mIoU robuster gegenüber Klassenungleichgewichten, da sie jede Klasse gleich gewichtet. Allerdings kann die Metrik bei seltenen Klassen instabil sein und bildet die visuelle Segmentierungsqualität nur begrenzt ab.
Die Average Precision ist eine gebräuchliche Metrik zur Bewertung der Objektdetektion und Instance Segmentation. Sie beschreibt die Fläche unter der Precision-Recall-Kurve und wird in der Praxis meist diskret berechnet. Eine gängige Definition lautet:

$$\mathrm{AP} = \sum_{k=1}^{n} P(k) \cdot \Delta r(k), \tag{2.3}$$

wobei $P(k)$ die Precision an Position k ist und $\Delta r(k)$ die Änderung des Recall zwischen den Positionen $k-1$ und k darstellt. Die Precision-Recall-Kurve wird damit schrittweise integriert, was der Fläche unter der Kurve entspricht. Für eine umfassende Bewertung wird häufig die mean Average Precision (mAP) über alle Klassen berechnet. Im COCO-Benchmark wird die AP über verschiedene Intersection-over-Union-Schwellenwerte gemittelt, typischerweise im Bereich von 0,5 bis 0,95 in 0,05-Schritten, um ein vollständigeres Bild der Modellleistung zu erhalten [31].
Die Panoptic Quality wurde eingeführt, um in der Panoptic Segmentation sowohl die semantische Korrektheit als auch die instanzspezifische Präzision in einem einzigen Wert zusammenzufassen. Sie ist definiert als

$$\mathrm{PQ} = \frac{\sum_{(p,g)\in TP} \mathrm{IoU}(p,g)}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}, \tag{2.4}$$

wobei TP die Menge korrekt erfasster Vorhersage-Ground-Truth-Paare mit IoU $> 0{,}5$, FP die Anzahl falsch positiver Vorhersagen und FN die Zahl der nicht erkannten Ground-Truth-Instanzen bezeichnen. Die Metrik PQ balanciert damit Präzision und Vollständigkeit und ist besonders geeignet für komplexe Szenen mit vielen Objektinstanzen [11].
Die Bewertung von Object-Tracking-Verfahren erfolgt üblicherweise anhand etablierter Metriken, welche die Leistungsfähigkeit eines Systems in Bezug auf die Detektion sowie die zeitlich konsistente Verfolgung von Objekten erfassen. Zu den am häufigsten verwendeten Kennzahlen zählen die Multiple Object Tracking Accuracy (MOTA), der Identification F1-Score (IDF1) sowie die Higher Order Tracking Accuracy (HOTA).
Die MOTA-Metrik berücksichtigt drei Fehlerquellen: False Positives (Detektionen ohne

tatsächliches Objekt), False Negatives (verpasste Objekte trotz Ground Truth) sowie Identity Switches (Verwechslung der Objektzuordnung über aufeinanderfolgende Frames hinweg). Ein Identity Switch liegt vor, wenn ein Ground-Truth-Objekt im aktuellen Frame einem Track zugeordnet wird und im darauffolgenden Frame einem anderen Track, obwohl es sich weiterhin um dasselbe Objekt handelt.
Die Berechnung von MOTA erfolgt, nach [32], über alle Frames t hinweg nach folgender Formel:

$$\mathrm{MOTA} = 1 - \frac{\sum_t (\mathrm{FN}_t + \mathrm{FP}_t + \mathrm{IDSW}_t)}{\sum_t \mathrm{GT}_t}. \tag{2.5}$$

Dabei bezeichnet GT_t die Anzahl der Ground-Truth-Objekte im Frame t.
Eine Schwäche von MOTA besteht darin, dass ein einmaliger Identity Switch nur einmal gezählt wird, selbst wenn die falsche Zuordnung über viele Frames bestehen bleibt. Zudem dominieren in Szenen mit hoher Objektdichte die False Positives und False Negatives das Gesamtergebnis, wodurch Identity Switches im Vergleich wenig Gewicht erhalten. Da die Erkennung als korrekt oder falsch ausschließlich auf Basis eines festen IoU-Schwellenwerts erfolgt, bleibt eine verbesserte oder verschlechterte Lokalisierung unberücksichtigt.
Der IDF1-Score basiert auf den Konzepten Identification Precision (Verhältnis korrekt zugeordneter Tracks zu allen zugewiesenen Tracks) und Identification Recall (Verhältnis korrekt zugeordneter Tracks zu allen existierenden Ground-Truth-Spuren). Im Gegensatz zu MOTA betrachtet IDF1 die gesamte Sequenz und nicht nur den Übergang zwischen zwei Frames. Die Bewertung erfolgt damit konsistenter über die Zeit hinweg. Die Berechnung wird nach [33] definiert, als

$$\mathrm{IDF1} = \frac{2 \cdot \mathrm{IDTP}}{2 \cdot \mathrm{IDTP} + \mathrm{IDFP} + \mathrm{IDFN}}. \tag{2.6}$$

Hierbei bezeichnet IDTP die Anzahl korrekt zugeordneter Objektinstanzen über die gesamte Sequenz, IDFP die Zahl an Track-Zuordnungen ohne zugehöriges Ground-Truth-Objekt und IDFN die Anzahl der Objekte, für die keine korrekte Zuordnung gefunden wurde.
IDF1 erlaubt eine deutlich detailliertere Bewertung der Assoziierungsgenauigkeit, kann jedoch komplexe Szenarien mit temporären Verdeckungen, Objektfusionen oder stark variierenden Detektionen vereinfachen. Besonders in Szenen mit schwankender Detektionsqualität beeinflussen FP das IDF1-Ergebnis, auch wenn diese nicht durch den Tracker selbst verursacht wurden.
Mit HOTA wurde eine Metrik eingeführt, die sowohl die räumliche Genauigkeit der Detektion als auch die zeitliche Konsistenz der Assoziierung berücksichtigt. Die Higher

Order Tracking Accuracy ist als geometrisches Mittel zweier Teilmetriken definiert: Detection Accuracy (DetA) und Association Accuracy (AssA). DetA misst die Qualität der Objektlokalisierung über alle Frames hinweg. Dabei wird eine Detektion nur dann als korrekt (True Positive) gewertet, wenn der Intersection-over-Union-Wert (IoU) mit dem Ground Truth ein definiertes Schwellwertniveau α übersteigt. AssA hingegen beschreibt die Fähigkeit des Trackers, einem Objekt über die Zeit hinweg eine konsistente ID zuzuweisen.
Die HOTA-Formel, nach [34], lautet:

$$\text{HOTA} = \sqrt{\text{DetA} \cdot \text{AssA}}. \tag{2.7}$$

Zur Stabilisierung der Bewertung wird HOTA nicht nur für ein einzelnes α berechnet, sondern über 19 diskrete Schwellenwerte $\alpha \in \{0.05, 0.10, \ldots, 0.95\}$ gemittelt. Dadurch ergibt sich eine differenzierte, gleichzeitig robuste Einschätzung der Gesamtleistung.
HOTA bietet somit eine ausgewogene Alternative zu MOTA und IDF1, indem sie sowohl präzise Lokalisierung als auch zuverlässige Objektverfolgung in die Bewertung einbezieht. Ein Ansatz, der besonders bei komplexen Szenen mit hoher Objektanzahl und wechselnden Sichtbedingungen von Vorteil ist.

3 Hauptteil

Der Hauptteil dieser Arbeit behandelt die Entwicklung eines kamerabasierten Systems zur automatisierten Verkehrszählung und Fahrzeugklassifikation. Ziel war es, eine modulare Lösung zu konzipieren, die auf bestehenden Segmentierungs- und Tracking-Modellen basiert und um eine eigens entwickelte Logik zur Zählung und Klassifikation erweitert wird. Dabei wurden zwei Ansätze untersucht: Eine klassische Zählung auf Basis kompletter Fahrzeuginstanzen und eine erweiterte Zählung mithilfe der Radpositionen zur Ableitung der Achskonfiguration. Die Auswahl geeigneter Modelle, die Definition relevanter Fahrzeugklassen sowie die Umsetzung der beiden Ansätze werden im Detail dargestellt und anschließend anhand simulierter und realer Bilddaten evaluiert.

3.1 Konzeption und Methodik

Im folgenden Abschnitt werden die konzeptionellen Grundlagen und methodischen Entscheidungen zur Entwicklung des Zielsystems dargelegt. Zunächst werden die übergeordneten Anforderungen und Rahmenbedingungen beschrieben, die sich aus dem geplanten Anwendungskontext ergeben. Darauf aufbauend erfolgt die Auswahl geeigneter Modelle für die einzelnen Teilschritte der Verkehrszählung. Abschließend wird die verwendete Klassenstruktur definiert, die als Grundlage für Detektion, Klassifikation und spätere Auswertung dient.

3.1.1 Anforderungen und Randbedingungen an das Zielsystem

Ziel des zu entwickelnden Verfahrens ist die präzise Klassifikation von Fahrzeugen anhand ihrer Achskonfiguration. Dabei soll das System in der Lage sein, zwischen Einzel-, Doppel- und Dreifachachsen zu unterscheiden und die erkannten Achsstrukturen zuverlässig der jeweiligen Fahrzeuginstanz zuzuordnen. Die Segmentierung der Fahrzeuge und der Räder erfolgt mittels Deep-Learning-gestützter Instance Segmenta-

tion und wird durch ein nachgeschaltetes Tracking-Modul ergänzt, um eine konsistente Objektverfolgung zu gewährleisten.
Ein zentrales Kriterium ist die Robustheit der Detektion. Das System soll in der Lage sein, auch unter erschwerten Bedingungen, wie widrigen Wetterverhältnissen, teilweiser Verdeckung von Objekten oder variabler Belichtung, zuverlässige Ergebnisse zu liefern. Besonders bei geringer Bildqualität oder niedrigeren Auflösungen bis hinunter zu 720p muss die Detektion und Zuordnung von Räder und Fahrzeugen korrekt funktionieren, da derartige Einschränkungen in praxisnahen Szenarien (z. B. bei älteren Kamerasystemen) häufig auftreten.
Im Gegensatz zu Systemen für den Einsatz in der Echtzeitanalyse bestehen für diese Arbeit keine Anforderungen an eine laufzeiteffiziente Implementierung. Stattdessen liegt der Fokus auf der Erkennungsgenauigkeit und der strukturellen Integrität der Klassifikation. Die Erkennungsergebnisse sollen in sich konsistent und nachvollziehbar sein, auch wenn die Verarbeitung zeitlich verzögert erfolgt.
Ein weiteres Ziel ist die skalierbare Modulierbarkeit der Methode. Sie soll mit unterschiedlichen Kameraquellen, variablen Blickwinkeln und Bildfrequenzen ab 30 Hz kompatibel sein. Zudem muss sich das System durch eine saubere Trennung der Komponenten (Detektion, Tracking und Klassifikation) modular erweitern und anpassen lassen. Dies erlaubt eine spätere Integration in komplexere Verkehrsanalysesysteme oder die Erweiterung um zusätzliche Merkmale wie Fahrzeuglänge, Aufbautyp oder Beleuchtungseinrichtung.
Die zur Verfügung stehende Hardware umfasst eine NVIDIA RTX 4060 Ti mit 16 GB VRAM sowie einen AMD Ryzen 7 7700X Prozessor. Damit steht ein leistungsfähiges, aber dennoch auf Einzelplatzniveau begrenztes System zur Verfügung. Es erlaubt die Durchführung rechenintensiver Deep-Learning-Verfahren unter praxisnahen Bedingungen, jedoch ohne Zugriff auf High-End-Rechencluster oder Multi-GPU-Umgebungen. Die Auswahl und Konfiguration der Netzwerke muss daher so erfolgen, dass sie auch bei längeren Trainingsläufen und hoher Speicherbelastung stabil auf dieser Hardware betrieben werden können. Besonders speicherintensive Modelle oder Architekturen mit extrem tiefen Feature-Pyramiden sollten vermieden oder hinsichtlich ihrer Speicheranforderung optimiert werden. Ziel ist es, eine möglichst gute Segmentierungs- und Assoziierungsleistung bei vertretbarem Ressourcenverbrauch zu erzielen.

3.1.2 Auswahl geeigneter Modelle

Für die Auswahl eines geeigneten Modells zur Instance Segmentation standen verschiedene Aspekte im Vordergrund, die sich aus der Zielsetzung der Arbeit und den praktischen Rahmenbedingungen ergeben. Ziel ist es, Fahrzeuge und deren Räderinstanzen zuverlässig zu segmentieren, um im Anschluss eine robuste Bestimmung der Achskonfiguration durchführen zu können. Da insbesondere Räder vergleichsweise kleine, häufig verdeckte und teilweise nur ausschnittweise sichtbare Objekte darstellen, sind Modelle erforderlich, die auch unter solchen Bedingungen feinkörnige Segmentierungen erzeugen. Eine zentrale Anforderung an das verwendete Modell ist daher eine hohe Segmentierungsgenauigkeit insbesondere im Hinblick auf die Separierbarkeit von Einzelobjekten derselben Klasse. Die Fähigkeit, verschiedene Fahrzeuge und ihre zugehörigen Räder als unabhängige Instanzen im Bild korrekt zu erkennen, bildet die Grundlage für die nachgelagerte Klassifikation nach Achsstruktur. Ein Fokus liegt dementsprechend auf der präzisen Objekttrennung bei gleichzeitig stabiler Maskengenerierung für kleine Objekte wie Räder. Im Hinblick auf die verfügbaren Rechenressourcen wurde besonderer Wert auf die Speicher- und Laufzeiteffizienz gelegt. Während die Ausführung in Echtzeit nicht erforderlich ist, soll das System dennoch in der Lage sein, auch längere Bildsequenzen mit hoher Objektanzahl bei moderaten Auflösungen (z. B. 1280x720) stabil und ohne Speicherauslastung auszuführen. Dies schließt sowohl die Inferenz des Segmentierungsmodells als auch dessen Einbindung in eine größere Verarbeitungspipeline mit ein.
Gleichzeitig ist zu berücksichtigen, dass die Aufnahmebedingungen in realen Szenarien stark variieren können. Daher muss das Modell auch unter ungünstigen Umgebungsbedingungen wie schlechtem Wetter, Schattenwurf oder variabler Belichtung eine zuverlässige Vorhersagequalität liefern. Die Robustheit gegenüber solchen Störfaktoren ist entscheidend, um eine konsistente Klassifikation über verschiedene Bildabschnitte hinweg sicherzustellen. Die Auswahl geeigneter Segmentierungsmodelle für den Einsatz in verkehrsanalytischen Anwendungen basiert auf mehreren technischen und funktionalen Kriterien. Ein zentrales Auswahlkriterium ist die Fähigkeit des Modells, unterschiedliche Fahrzeugklassen sowie feinere Objektmerkmale, etwa Räder, zuverlässig zu segmentieren. Dabei ist insbesondere die Trennschärfe zwischen einzelnen Instanzen relevant, da eine saubere Abgrenzung der Objekte Voraussetzung für die spätere Zählung und Klassifikation ist.
Ein weiteres Kriterium ist die Flexibilität bei der Definition der zu segmentierenden Zielobjekte. Modelle, die es erlauben, Detektionsziele dynamisch und anwendungsspezi-

fisch festzulegen, bieten deutliche Vorteile gegenüber statisch trainierten Architekturen mit festen Klassen. Insbesondere text- oder promptbasierte Steuerungsmechanismen können hier die Konfigurierbarkeit und Adaptierbarkeit des Systems wesentlich verbessern.
Darüber hinaus spielt die Struktur der Ausgabedaten eine entscheidende Rolle. Für eine effiziente Weiterverarbeitung, beispielsweise im Kontext des Multi-Object-Trackings, müssen die Segmentierungsergebnisse klar strukturiert, instanzbasiert abgelegt und maschinenlesbar exportiert werden können. Eine konsistente und standardisierte Ergebnisrepräsentation ist dabei ebenso wichtig wie die Verfügbarkeit von Informationen zur Position und Form der Objekte.
Schließlich sind auch praktische Aspekte wie die Kompatibilität mit bestehenden Tracking-Modulen, die Rechenanforderungen des Modells sowie die Möglichkeit zur späteren Erweiterung auf weitere Domänen oder Sensormodalitäten relevante Auswahlkriterien. Diese betreffen weniger die Modellleistung an sich, beeinflussen jedoch maßgeblich die Integrierbarkeit in ein komplexes Gesamtsystem.
Andere Auswahlkriterien wie die Skalierbarkeit auf zusätzliche Modalitäten oder Domänen wurden im Rahmen dieser Arbeit nicht weiterverfolgt, da eine domänenspezifische Anwendung auf sichtbasiertes Videomaterial im Straßenverkehr im Vordergrund steht. Auch die Kompatibilität mit Tracking-Systemen stellte kein Ausschlusskriterium dar, da die Auswahl und Einbindung eines geeigneten Tracking-Models explizit Teil des zu entwickelnden Gesamtsystems ist.

3.1.3 Definition der Klassen

Für die Klassifikation der Verkehrsteilnehmer im Rahmen dieser Arbeit wird zwischen zwei Ebenen unterschieden: einer grundständigen Einteilung gemäß der klassischen Verkehrszählung sowie einer erweiterten Klassifikation nach Achskonfiguration auf Basis des TLS-8+1-Schemas. Diese beiden Ebenen unterscheiden sich sowohl im Detailgrad als auch in ihren Anforderungen an die Bildauswertung.
Die klassische Verkehrszählung verfolgt das Ziel, einen allgemeinen Überblick über die Zusammensetzung des Verkehrsflusses zu gewinnen. Hierfür ist in der Regel eine Unterteilung in drei grundlegende Klassen ausreichend: *Personenkraftwagen (Pkw)*, *Lastkraftwagen (Lkw)* und *Busse*. Diese Kategorisierung erlaubt bereits grundlegende Analysen hinsichtlich Verkehrsaufkommen und zeitlicher Verteilung, ist jedoch nicht in der Lage, unterschiedliche Belastungswirkungen einzelner Fahrzeugtypen zu erfassen. Ein leichter Kleintransporter und ein schwerer Sattelschlepper würden innerhalb der

Kategorie „Lkw“ gleich behandelt, obwohl sie sich hinsichtlich Achslast, Emissionen und Straßenabnutzung stark unterscheiden.
Um diesen Mangel zu adressieren, wurde in Deutschland das TLS-Klassifikationsschema entwickelt, das im Rahmen der Technischen Lieferbedingungen für Streckenstationen (TLS) standardisiert ist. Die TLS wurden durch die Bundesanstalt für Straßenwesen (BASt) eingeführt und erstmals im Jahr 1996 veröffentlicht. Sie definieren technische Anforderungen an automatische Dauerzählstellen und deren Auswertealgorithmen. Die Version TLS 8+1, nach [35], beschreibt ein Klassenschema, das acht reguläre Fahrzeugklassen und eine Sonderklasse umfasst. Dieses Schema basiert auf der Kombination aus Fahrzeuglänge und Achskonfiguration und ermöglicht damit eine deutlich differenziertere Einordnung des motorisierten Verkehrs.
Die acht Hauptklassen unterteilen den Verkehr unter anderem in Motorräder, Pkw ohne und mit Anhänger, leichte und schwere Lkw, Sattelkraftfahrzeuge, Busse sowie sonstige mehrachsige Fahrzeuge. Die zusätzliche Sonderklasse „9“ dient der Erfassung nicht eindeutig zuordenbarer oder fehlerhaft detektierter Fahrzeuge. Für eine automatisierte Umsetzung dieser Klassifikation ist es notwendig, die Anzahl und Verteilung der Achsen zuverlässig aus Bildmaterial abzuleiten, was wiederum eine präzise Detektion der sichtbaren Räder voraussetzt.
Im Kontext dieser Arbeit stellt die Anwendung des TLS 8+1-Schemas eine Erweiterung zur klassischen Zählung dar. Ziel ist es, über eine instanzbasierte Segmentierung und eine darauf aufbauende Analyse der Räderpositionen und -anzahl eine automatische Einordnung von Fahrzeugen nach diesem Schema zu ermöglichen. Die dadurch erzielte Differenzierung erlaubt eine realitätsnähere Bewertung des Verkehrs in Bezug auf Infrastrukturbelastung, Emissionsverhalten und Kategorisierung für verkehrsrechtliche Maßnahmen.
Die Umsetzung dieser erweiterten Klassifikation stellt höhere Anforderungen an die Bildauswertung, insbesondere in Bezug auf die Segmentierungsqualität bei verdeckten oder teilverdeckten Räder sowie die korrekte Assoziierung der Räder zu einzelnen Fahrzeuginstanzen. Damit bildet sie eine zentrale Herausforderung und zugleich das Hauptziel dieser Arbeit.

3.2 Implementierung

Dieser Abschnitt beschreibt die technische Umsetzung des entwickelten Systems zur Verkehrszählung. Zunächst wird der Aufbau der klassischen Verkehrszählung erläutert, inklusive der gewählten Architektur, der Fahrzeugdetektion und des Trackings. Anschließend folgt die Darstellung der erweiterten Verkehrszählung, bei der zusätzlich sichtbare Reifen segmentiert und auf Basis ihrer Positionen Klassifikationen vorgenommen werden. Dabei wird auf die eingesetzten Modelle sowie deren Integration in die Systemarchitektur eingegangen.

3.2.1 Aufbau der klassischen Verkehrszählung

Die Grundlage der im Rahmen dieser Arbeit entwickelten Methode bildet ein klassischer Ansatz zur automatisierten Verkehrszählung auf Basis bildbasierter Instance Segmentation. Ziel ist es, Fahrzeuge zuverlässig zu erkennen, über mehrere Bilder hinweg zu verfolgen und basierend auf ihrer erkannten Kategorie zu zählen. Dieser Grundmechanismus bildet das Rückgrat der später erweiterten Analysepipeline, in der zusätzlich eine Differenzierung nach Achskonfiguration erfolgt.
Die klassische Zählung orientiert sich an etablierten Verkehrsklassen und erfasst ausschließlich die Objektkategorien *Pkw*, *Lkw* und *Bus*. Eine Differenzierung innerhalb dieser Gruppen wird an dieser Stelle nicht vorgenommen. Im Vordergrund steht vielmehr die zuverlässige Identifikation und Zählung jedes Fahrzeugs, auch unter erschwerten Bedingungen wie Bildrauschen, verdeckten Strukturen, wechselnden Lichtverhältnissen oder der Verwendung von Graustufenvideos anstelle von RGB-Videos.
Zur Umsetzung wird eine segmentierungsbasierte Detektion mit einem nachgeschalteten Tracking-System kombiniert. Das verwendete Instance Segmentation Modell erkennt Fahrzeuge als einzelne, voneinander getrennte Objekte innerhalb eines Bildausschnitts. In der Folge werden diese über einen eigens angebundenen Tracker zu Bewegungsbahnen über mehrere Frames hinweg verknüpft. Dadurch kann ein Objekt auch dann korrekt gezählt werden, wenn es nicht vollständig im Bild erscheint oder kurzzeitig verdeckt ist.
Die Kommunikation zwischen den Modulen sowie die Speicherung der Ausgabedaten erfolgt lokal. Im nachfolgenden Kapitel wird die Modellarchitektur zur Fahrzeugdetektion im Detail vorgestellt, bevor anschließend das Trackingverfahren des verteilten Systems erläutert wird.

Detektion von Fahrzeugen Die präzise Detektion von Fahrzeugen bildet die Grundlage für alle nachgelagerten Verarbeitungsschritte innerhalb der entwickelten Systemarchitektur. Ohne eine zuverlässige Erkennung einzelner Fahrzeuginstanzen ist weder eine korrekte Zählung noch eine nachfolgende Klassifikation der Fahrzeugtypen oder Achskonfigurationen möglich. Die Detektion legt somit den Grundstein für das gesamte Verfahren zur erweiterten Verkehrszählung.
Im Gegensatz zu herkömmlichen Verfahren, die lediglich auf die grobe Erkennung von Verkehrsobjekten abzielen, werden in dieser Arbeit Instanzen auf Pixelebene segmentiert. Dies ermöglicht es, einzelne Fahrzeuge eindeutig zu identifizieren, auch in komplexen Szenen mit Überlappungen, Teilverdeckungen oder variierenden Perspektiven. Die hohe räumliche Trennschärfe der Instanzmasken erhöht die Genauigkeit der Zählung und schafft die notwendige Voraussetzung für die anschließende Verknüpfung mit weiteren Detektionen, etwa von Räder, im Rahmen der Achsbestimmung.
Besonders relevant ist die Auswahl eines geeigneten Modells, das sowohl in Hinblick auf die Detektionsleistung als auch auf die Integration in das technische System überzeugt. Dabei ist nicht nur die Qualität der Segmentierung entscheidend, sondern auch die Robustheit gegenüber wechselnden Umweltbedingungen, etwa bei ungünstigem Wetter oder niedriger Bildauflösung.
Für die Detektion der Fahrzeuge wurde das Modell OneFormer verwendet, das in einem separaten Docker-Container betrieben wird. Die Umgebung basiert auf dem offiziellen NVIDIA-Image nvidia/cuda:11.3.1-devel-ubuntu20.04, in dem ein Python-3.8-Interpreter innerhalb einer conda-Umgebung ausgeführt wird.
OneFormer zeichnet sich durch eine Architektur aus, die es ermöglicht, mit nur einem einzigen Trainingslauf ein Modell bereitzustellen, das gleichzeitig für Semantic Segmentation, Instance Segmentation und Panoptic Segmentation eingesetzt werden kann. Dabei erreicht OneFormer in allen drei Aufgabenstellungen Ergebnisse auf damaligem Stand der Technik, gemessen an etablierten Benchmarks wie Cityscapes†, COCO val2017† [28] und ADE20K val† [29], ohne dass für jede dieser Aufgaben ein separates Training erforderlich ist.
Für die vorliegende Anwendung der Instance Segmentation von Fahrzeugen zur Zählung wurde ein auf dem COCO val2017†-Datensatz vortrainiertes OneFormer-Modell mit DiNAT-L† [36] als Backbone verwendet. Der COCO-Datensatz enthält die für die Verkehrszählung relevanten Objektkategorien, darunter *car (Pkw)*, *(Bus)* und *truck (Lkw)*. Obwohl OneFormer mit dem alternativen Backbone Swin-L† [37] eine höhere Panoptic Quality (PQ) für die Kategorie Things erzielt, wurde in der Konfiguration mit DiNAT-L† eine höhere Average Precision (AP) im Bereich der Instance Segmentation

gemessen. Da für die angestrebte Anwendung die Genauigkeit der Maskenkonturen eine untergeordnete Rolle spielt und stattdessen eine möglichst hohe Precision-Recall-Rate bei der Erkennung von Fahrzeuginstanzen im Vordergrund steht, wurde die Konfiguration mit DiNAT-L† als Backbone ausgewählt.
Die verwendeten Testdaten weisen im Vergleich zu etablierten Benchmark-Datensätzen eine deutlich geringere Bildauflösung auf. In Kombination mit Kompressionsartefakten und ungünstigen Aufnahmebedingungen führt dies zu einer erhöhten Unsicherheit in der Objekterkennung. Unter diesen Randbedingungen kommt es vermehrt zu falsch positiven Detektionen, bei denen statische Objekte wie beispielsweise Abfallbehälter, Werbeträger oder bauliche Elemente fälschlicherweise als Fahrzeuginstanzen klassifiziert werden. Die dabei erzeugten Segmentierungsmasken sind in sich konsistent, repräsentieren jedoch kein tatsächlich vorhandenes Fahrzeugobjekt. Auch eine Anpassung des Confidence Schwellwertes des Detektionsmodells reduziert diese Effekte nur eingeschränkt.

Zur Reduktion solcher Fehlklassifikationen wird eine Region of Interest eingeführt, die ausschließlich den Bildbereich umfasst, in dem das Auftreten von Fahrzeugen plausibel ist. Bildbereiche wie Gehwege, Randzonen oder angrenzende Bebauung werden dadurch systematisch von der Detektion ausgeschlossen. Diese räumliche Einschränkung reduziert die Anzahl potenzieller Störobjekte erheblich und erhöht die Robustheit der Fahrzeugdetektion, insbesondere bei niedriger Eingangsauflösung.
Durch die Einführung der Region of Interest kann die Häufigkeit falsch positiver Detektionen deutlich reduziert werden. Dennoch ist zu berücksichtigen, dass die Region of Interest insbesondere in urbanen Szenarien vergleichsweise groß ausfallen muss, um unterschiedliche Fahrspuren, Haltezonen oder komplexe Verkehrssituationen vollständig abzudecken. In Verbindung mit der eingeschränkten Bildqualität kann es daher weiterhin zu vereinzelten Fehlklassifikationen kommen, etwa wenn Objekte in unmittelbarer Nähe von Bushaltestellen, Verkehrsinseln oder parkenden Fahrzeugen aufgrund ihrer Form oder Textur fälschlicherweise als Fahrzeuge erkannt werden.
Diese verbleibenden Unsicherheiten werden im Gesamtsystem bewusst akzeptiert und in den nachgelagerten Verarbeitungsschritten berücksichtigt. Insbesondere die anschließende achsbasierte Klassifikation nutzt zeitliche Konsistenz, geometrische Plausibilitätskriterien sowie die Verknüpfung mit Räderdetektionen, um inkonsistente oder nicht plausible Fahrzeughypothesen zu identifizieren und auszuschließen. Auf diese Weise wird sichergestellt, dass falsch positive Detektionen auf Ebene der Fahrzeugerkennung keinen maßgeblichen Einfluss auf das Ergebnis der erweiterten Verkehrszählung haben.

[0]†Backbone wurde auf ImageNet-22K vortrainiert

Tracking von Fahrzeugen Das Tracking von Fahrzeugen stellt eine zentrale Komponente innerhalb des Gesamtsystems dar, da es die zeitliche Verknüpfung einzelner Detektionen über aufeinanderfolgende Frames hinweg ermöglicht. Nur durch die eindeutige Zuordnung von Instanzen zu konsistenten Objekt-IDs kann zwischen wiederholten Erkennungen desselben Fahrzeugs und tatsächlich neuen Objekten unterschieden werden. Diese Fähigkeit ist essenziell, um eine verlässliche Zählung der Fahrzeuge zu gewährleisten, ohne Mehrfachzählungen oder Verluste durch temporäre Verdeckungen. Insbesondere für die in dieser Arbeit verfolgte Erweiterung der klassischen Verkehrszählung um eine achsbasierte Klassifikation ist ein stabiles und robustes Tracking erforderlich. Die genaue Bestimmung der Achskonfiguration setzt voraus, dass zusätzlich erkannte Räderkonfigurationen zuverlässig den zugehörigen Fahrzeuginstanzen zugeordnet werden können. Dies gelingt nur, wenn sowohl die Fahrzeuge als auch die Räder über die Zeit hinweg konsistent verfolgt werden. Ohne ein leistungsfähiges Tracking-Modul wären die im Anschluss durchgeführten Zuordnungs- und Klassifikationsschritte nicht hinreichend robust oder könnten sogar zu systematischen Fehlern führen.
Darüber hinaus ermöglicht das Tracking-Modul die Implementierung weiterer Validierungsmechanismen innerhalb der Pipeline. Beispielsweise können inkonsistente Bewegungsverläufe oder sprunghafte Positionsänderungen zur Zählung fehlerhafter Zuordnungen oder fehlgeschlagener Segmentierungen herangezogen werden. Tracking bildet somit nicht nur die Grundlage für die eigentliche Zählung, sondern stellt auch ein wichtiges Werkzeug zur Qualitätskontrolle innerhalb der gesamten Segmentierungs- und Klassifikationskette dar.
MOT hat das Ziel, Objekte in einer Videosequenz über mehrere Frames hinweg konsistent zu verfolgen, indem deren Positionen (in Form von Bounding Boxes) erkannt und ihnen eindeutige Identifikatoren (IDs) zugewiesen werden. Auf diese Weise soll jedes Objekt über die Zeit hinweg unter derselben Identität geführt werden. Das Tracking bildet damit die Brücke zwischen der frameweisen Detektion und einer objektbasierten Analyse im Zeitverlauf.
Eine besondere Herausforderung bei der Fahrzeugzählung im Straßenverkehr besteht darin, dass Fahrzeuge zeitweise durch andere Objekte verdeckt oder aufgrund schlechter Sichtverhältnisse nur schwach detektiert werden. In solchen Fällen besteht die Gefahr, dass Objekte verloren gehen und bei späterer Wiedererkennung fälschlicherweise als neue Objekte gezählt werden, was zu einer Doppelzählung führt.
Für die Tracking-Komponente wurde in dieser Arbeit der ByteTrack-Algorithmus eingesetzt, welcher 2022 von Zhang et al. vorgestellt wurde [25]. ByteTrack erweitert klassische Tracking-by-Detection-Ansätze, indem er auch Detektionen mit niedrigem Con-

fidence Score in den Matching-Prozess integriert. Auf diese Weise kann der Algorithmus bestehende Tracks auch in schwierigen Situationen zuverlässig weiterverfolgen, ohne jedoch anfällig für das Aufnehmen falscher Objekte zu werden. Gleichzeitig ignoriert ByteTrack Detektionen mit zu niedriger Wahrscheinlichkeit, die mit hoher Wahrscheinlichkeit Falschpositiven entsprechen.
ByteTrack erreichte auf dem MOT17-Datensatz bei 30 FPS eine MOTA von 80,3, einen IDF1-Score von 77,3 sowie eine HOTA von 63,1 und liegt damit auf dem damaligen Stand der Technik (SOTA). Die hohe Tracking-Genauigkeit bei gleichzeitig effizienter Laufzeit macht das Verfahren besonders geeignet für den Einsatz in ressourcenbeschränkten Anwendungen wie dieser Arbeit.
Die Matching-Strategie von ByteTrack ist in zwei Assoziationsphasen unterteilt. Zunächst werden alle Detektionen mit einem Confidence Score oberhalb eines definierten Schwellenwertes τ als *high score detection boxes* D_{high} bezeichnet, während alle unterhalb dieser Schwelle als *low score detection boxes* D_{low} klassifiziert werden. Anschließend wird für jeden bestehenden Track mit Hilfe eines Kalman-Filters eine Vorhersage der nächsten Objektposition berechnet.
In der ersten Assoziationsphase erfolgt die Zuordnung der Detektionen aus D_{high} zu den aktiven Tracks $\boldsymbol{\mathcal{T}}$ sowie zu den kurzfristig verlorenen Tracks $\boldsymbol{\mathcal{T}}_{\text{lost}}$. Hierfür wird die *Intersection over Union* (IoU) zwischen der vorhergesagten Bounding Box des Kalman-Filters und der Detektionsbox berechnet und als Kostenfunktion für den Kuhn-Munkres-Algorithmus (Hungarian Algorithmus) verwendet, um eine optimale Zuordnung zu bestimmen.
Detektionen, für die keine Übereinstimmung gefunden wurde, werden in D_{remain} gespeichert. Analog dazu werden die verbleibenden, nicht zugeordneten Tracks als $\boldsymbol{\mathcal{T}}_{\text{remain}}$ bezeichnet. In einer zweiten Matching-Phase werden die übrig gebliebenen low-score-Detektionen D_{low} mit den nicht zugeordneten Tracks $\boldsymbol{\mathcal{T}}_{\text{remain}}$ verglichen. Nach dieser zweiten Assoziation werden alle verbleibenden Tracks als $\boldsymbol{\mathcal{T}}_{\text{re-remain}}$ markiert, und die verbleibenden Detektionen aus D_{low} werden verworfen, da sie als Rauschen klassifiziert werden. Die verbliebenen, nicht zugeordneten Tracks werden anschließend in $\boldsymbol{\mathcal{T}}_{\text{lost}}$ überführt. Dort verbleiben sie, solange die maximale erlaubte Dauer im Frame-Buffer nicht überschritten wurde. Andernfalls werden sie gelöscht. Für alle unzugeordneten high-score-Detektionen aus D_{high} werden abschließend neue Tracks initialisiert.
Durch diese zweistufige Matching-Strategie gelingt es ByteTrack, die Vorteile robuster Low-Score-Detektionen nutzbar zu machen, ohne die Genauigkeit durch Falschzuweisungen zu beeinträchtigen.

Architektur Die Architektur der klassischen Verkehrszählung basiert auf den beiden zuvor beschriebenen Komponenten zur Fahrzeugdetektion und zum Multi-Object-Tracking sowie einer eigenentwickelten, einfachen Zählungslogik. Die Zählungslogik ist dem Tracking-Modul nachgelagert und wird im selben Docker-Container wie ByteTrack ausgeführt. Sie dient ausschließlich der Erfassung und Aggregation der detektierten und getrackten Fahrzeuginstanzen für die klassische Verkehrszählung.
Für die Anbindung an ByteTrack werden nicht die von OneFormer erzeugten Instanzmasken verwendet, sondern ausschließlich die zugehörigen Bounding Boxen. Dies ist erforderlich, da ByteTrack, wie zuvor beschrieben, Bounding Boxen als Eingabe erwartet und keine Masken verarbeitet. Die Masken werden in diesem Verarbeitungspfad nicht weiter berücksichtigt.
ByteTrack wurde in der vorliegenden Implementierung nur in begrenztem Umfang erweitert. Insbesondere wurde die Möglichkeit ergänzt, eine Track-ID mit dem zugehörigen Objektlabel zu verknüpfen und diese beiden Eigenschaften gemeinsam aus der Tracking-Ausgabe als *target* zu extrahieren. Diese Zuordnung ist erforderlich, da die Objektlabels in den nachgelagerten Verarbeitungsschritten für die fahrzeugklassenbasierte Zählung genutzt werden.
Über die einzelnen Frames hinweg werden die extrahierten *targets* entsprechend ihres Labels in separaten Listen geführt. Eine Fahrzeuginstanz wird nur dann in die jeweilige Liste aufgenommen, wenn die zugehörige eindeutige Track-ID dort noch nicht existiert. Auf diese Weise wird sichergestellt, dass jedes getrackte Fahrzeug innerhalb der klassischen Verkehrszählung nur einmal gezählt wird.
Die zugehörigen Bounding Boxen werden ausschließlich innerhalb von ByteTrack verarbeitet und temporär gespeichert. Nach Überschreiten des definierten Frame-Buffers werden diese Informationen verworfen. Eine darüber hinausgehende Speicherung oder ein weiteres Matching der Bounding Boxen ist für die klassische Verkehrszählung nicht erforderlich, da ausschließlich die Fahrzeuginstanzen der Klassen Pkw, Lkw und Bus berücksichtigt werden. Komplexere Zuordnungen, wie etwa die Verknüpfung von Anhängern mit Zugfahrzeugen, sind in diesem klassischen Zählpfad nicht vorgesehen und werden ausschließlich im erweiterten Zählsystem behandelt.

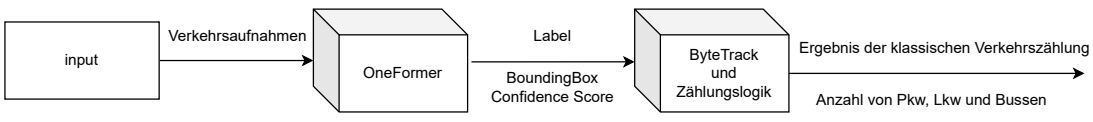


Abbildung 3.1: Systemarchitektur der klassischen Verkehrszählung

3.2.2 Aufbau der erweiterten Verkehrszählung

Die erweiterte Verkehrszählung ergänzt das klassische Verfahren um eine differenzierte Analyse anhand sichtbarer Räder, um Rückschlüsse auf Achskonfigurationen und Fahrzeugtypen zu ermöglichen. Dazu wird das Modell LangSAM zur textbasierten Segmentierung einzelner Radinstanzen eingesetzt, das auf den Komponenten Grounding DINO [38] und Segment Anything 2 (SAM2) [15] basiert. Ergänzend wurde eine eigene Zähl- und Gruppierungslogik implementiert, um die erkannten Räder zu Achsgruppen zusammenzufassen und daraus Klassifikationen abzuleiten. Der Aufbau dieser Komponenten sowie die zugrunde liegende Systemarchitektur werden im Folgenden beschrieben.

Segment Anything Model 2 SAM2 bildet die Grundlage für die präzise Segmentierung beliebiger Objektmasken innerhalb eines Bildes. Im Kontext dieser Arbeit wird es zur Maskierung potenzieller Räderbereiche verwendet, wobei es besonders durch seine Generalisierungsfähigkeit auf neue Objekte und Szenen überzeugt. Im Folgenden

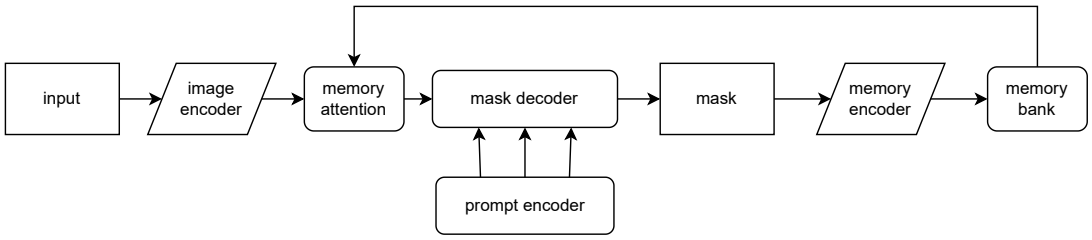


Abbildung 3.2: SAM2 Architektur

werden der architektonische Aufbau sowie die wesentlichen Komponenten von SAM2 erläutert, um die Rolle dieses Modells innerhalb des Gesamtsystems LangSAM nachvollziehbar zu machen.
SAM2 stellt eine Weiterentwicklung des ursprünglichen Segment Anything Model dar und adressiert gezielt Aufgaben der Video Object Segmentation (VOS). Während SAM auf Einzelbildsegmentierung beschränkt ist, erweitert SAM2 diese Funktionalität auf die zeitliche Dimension. Dadurch ist es möglich, Objekte nicht nur in einzelnen Bildern zu segmentieren, sondern deren maskierte Repräsentationen über eine Sequenz von Frames hinweg zu verfolgen.
Das Modell benötigt sogenannte promptable Visual Segmentation Tasks als Eingabe. Dazu zählen positive oder negative Punkte (die angeben, ob sie auf dem Objekt von Interesse liegen oder nicht), Boxen, Texteingaben oder auch bereits vorhandene

Segmentierungsmasken. Diese Prompts dienen als Initialisierung der Segmentierung, welche anschließend über nachfolgende Frames hinweg iterativ verfeinert wird.
Die Architektur von SAM2 besteht aus sechs wesentlichen Komponenten: Image Encoder, Prompt Encoder, Memory Attention, Mask Decoder, Memory Encoder und Memory Bank.
Als Image Encoder kommt ein Hiera-Modell zum Einsatz, das zuvor mithilfe eines Masked Autoencoders (MAE) vortrainiert wurde. Dieser Encoder erzeugt Feature Embeddings, die die visuelle Information des Frames in tokenisierter Form abbilden. Diese Embeddings sind zu diesem Zeitpunkt noch nicht auf spezifische Segmentierungen zugeschnitten.
Die Memory Attention bildet das Herzstück der zeitlichen Kontextverarbeitung. Sie dient dazu, die Feature Embeddings des aktuellen Frames im Zusammenhang mit vorherigen Frames und den zugehörigen Segmentierungsergebnissen sowie neuen Prompts zu interpretieren. Hierfür werden Transformer-Blöcke verwendet, in denen Self Attention den Bezug zu aktuellen Prompts und Cross Attention den Bezug zu vorherigen Frames und den dort gespeicherten Object Pointers herstellt.
Der Prompt Encoder verarbeitet zwei unterschiedliche Arten von Prompts: spärliche Eingaben (sparse) wie Punkte und Boxen sowie dichte Eingaben (dense) wie vorgegebene Masken. Punkte und Boxen werden mithilfe von Positionsembeddings codiert und anschließend mit jeweils eigens trainierten Embeddings des Prompt-Typs addiert. Texteingaben werden über einen CLIP-basierten Textencoder verarbeitet. Masken hingegen werden über Faltungsschichten extrahiert und anschließend elementweise mit den Bildembeddings multipliziert.
Im Mask Decoder erfolgt die Fusion der Bild- und Promptinformationen mit einem speziellen Output Token. Dieser Token aggregiert kontextuell relevante Informationen, aus denen die finale Segmentierungsmaske erzeugt wird. Eine Neuerung gegenüber SAM besteht darin, dass dieser Mask Token zusätzlich als Object Pointer für den jeweiligen Frame in der Memory Bank gespeichert wird. Damit wird eine kohärente zeitliche Verfolgung des Objekts über mehrere Frames hinweg ermöglicht.
Für Anwendungen wie die hier behandelte Verkehrszählung besonders relevant ist der Occlusion Prediction Head. Hierbei wird ein weiterer Token generiert und über ein Multi-Layer Perceptron (MLP) verarbeitet, um eine Wahrscheinlichkeit für die Sichtbarkeit des Objekts zu berechnen. Wird das Objekt als verdeckt erkannt, wird ein gelerntes Occlusion Embedding zur Memory Bank hinzugefügt, um die Informationslücke zu kompensieren.
Der Memory Encoder nimmt die Feature Embeddings aus dem Image Encoder entge-

gen und kombiniert sie mit der vorhergesagten Maske. Daraus werden die sogenannten Memory Embeddings generiert, ohne dass hierfür ein zusätzlicher Encoder erforderlich ist. Diese Embeddings stellen die Grundlage für konsistente Segmentierungsergebnisse in den Folgeframes dar.

Grounding DINO Ein zentrales Problem bei der Fahrzeugklassifikation auf Achsebene besteht darin, dass es derzeit keine öffentlich verfügbaren Datensätze gibt, die über geeignete Masken für die Instance Segmentation von Rädern verfügen. Zwar wäre eine manuelle Erweiterung bestehender Datensätze durch zusätzliche Annotationen grundsätzlich möglich, jedoch erweist sich dies aufgrund des hohen Zeitaufwands und der begrenzten lokalen Rechenressourcen als wenig praktikabel.
Eine vielversprechende Lösung für diese Herausforderung bietet das Modell Grounding DINO. Im folgenden Abschnitt werden dessen Architektur und zentrale Funktionsweisen vorgestellt, um das Zusammenspiel mit SAM2 im Kontext des übergeordneten LangSAM-Systems nachvollziehbar darzulegen. Es ermöglicht die textbasierte Detektion von Objekten, ohne dass diese explizit im Trainingsdatensatz vorhanden sein müssen. Nutzerinnen und Nutzer können hierzu Text Prompts definieren, die Kategorien oder sogar detaillierte Beschreibungen von Objekten enthalten. Grounding DINO basiert auf dem transformerbasierten Detektor DINO und erweitert diesen um die Fähigkeit, Objekte durch umfangreiches grounded pre-training in Verbindung mit Sprache zu erkennen. Dabei werden sprachliche Konzepte mit ihren visuellen Repräsentationen verknüpft [38].

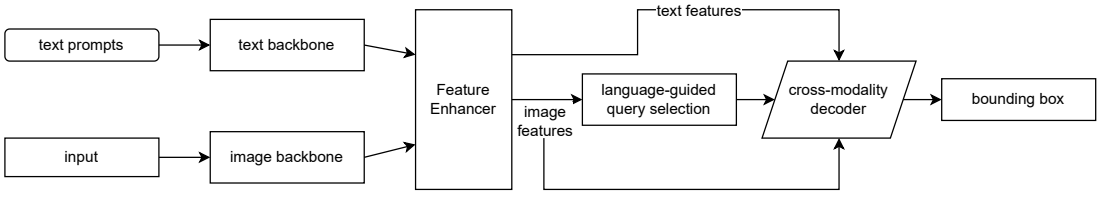


Abbildung 3.3: Vereinfachte Architektur von GroundingDINO

Im Unterschied zu früheren Ansätzen erfolgt die Fusion der Modalitäten in Grounding DINO nicht an einer einzelnen Stelle, sondern an drei verschiedenen Komponenten des Netzwerks. Das Modell folgt einer Dual-Encoder-Single-Decoder-Architektur, wobei als Image Encoder z. B. ein SWIN-Transformer verwendet wird und als Text Encoder ein BERT-Modell [37], [39]. In beiden Encodern werden Multi-Scale Features extrahiert, um sowohl globale als auch lokale Kontextinformationen zu erfassen.
Die erste Fusionskomponente ist der sogenannte Feature Enhancer. Dort durchlaufen Bild- und Textfeatures zunächst getrennte Self-Attention-Schichten. Anschließend erfolgt eine Modulation durch Text-to-Image Cross-Attention und Image-to-Text Cross-Attention, bevor ein Feedforward Network zur Verfeinerung der Repräsentation eingesetzt wird [38].
Die zweite Stelle zur Modalfusion ist das Language-Guided Query Selection-Modul im Detection Head. Hierbei werden diejenigen Feature-Repräsentationen identifiziert, die im semantischen Kontext der Nutzereingabe eine besonders hohe Relevanz besitzen.
Der dritte Aspekt der Modalfusion findet im Cross-Modality Decoder statt. Die dort verarbeiteten Queries bestehen aus einem positional part, der mit den Ausgaben des Encoders initialisiert wird (z. B. als anchor box), sowie einem content part, der im Training lernbar ist [40].
Jede dieser Queries durchläuft im Cross-Modality Decoder mehrere Layer, die jeweils aus Self-Attention, Image Cross-Attention, Text Cross-Attention sowie einem Feedforward Network bestehen. Ziel ist es, die Ausdrucksstärke der Query-Repräsentationen im Kontext der visuell-textuellen Übereinstimmung zu optimieren und somit eine möglichst präzise Detektion der durch Sprache spezifizierten Objekte zu ermöglichen.

Einsatz von LangSAM zur Raddetektion LangSAM ist ein Open-Source-Projekt, das zwei vortrainierte Modelle zu einer modularen Segmentierungspipeline kombiniert: Grounding DINO für die textbasierte Objektlokalisierung und SAM2 für die anschließende maskenbasierte Segmentierung. Eine eigene wissenschaftliche Veröffentlichung zu LangSAM liegt bislang nicht vor, jedoch bietet das Projekt eine funktionale Grundlage für anwendungsorientierte Aufgaben im Bereich der offenen Bildsegmentierung mithilfe von Visual Language Modellen.
Im Zentrum steht die Idee, mittels einer benutzerdefinierten Texteingabe (z. B. „tire“ oder „wheel“) mit Grounding DINO relevante Objekte in einem Bild zu lokalisieren. Grounding DINO generiert daraufhin präzise Bounding Boxen für alle Objekte, die der Bedeutung des Textes im gegebenen visuellen Kontext entsprechen.

Diese Boxen werden anschließend als Box Prompts an das Segment Anything Model 2 (SAM2) übergeben, das daraus hochauflösende Segmentierungsmasken erzeugt.
Diese Kombination erlaubt eine flexible, promptbasierte Detektion und Segmentierung von Objekten, ohne dass eine domänenspezifische Anpassung des zugrunde liegenden Modells erforderlich ist. Im Rahmen dieser Arbeit wurde LangSAM eingesetzt, um die Position und Form von FahrzeugRäder zuverlässig zu extrahieren. Die daraus gewonnenen Informationen bilden die Grundlage für die nachgelagerte achsbasierte Fahrzeugklassifikation, welche in enger Verbindung mit den Tracking-Ergebnissen steht und in einem eigenen Verarbeitungsschritt auf den segmentierten und verfolgten Objekten aufbaut.
Zur Initialisierung der Raddetektion wurde LangSAM mit dem Texteingabeprompt „wheel“ konfiguriert, da sich dieser Begriff in den vorangegangenen Versuchsreihen als die robusteste Alternative zur Identifikation von Fahrzeugrädern erwiesen hat. Die textbasierte Steuerung ermöglichte eine präzise Selektion relevanter Objekte im Bild, ohne zusätzliche Modellanpassung. Der Confidence Schwellwert $t_{CLangSAM}$ wurde auf einen festen Wert von 0,35 gesetzt. Diese Auswahl basiert auf empirischen Beobachtungen und stellt einen geeigneten Kompromiss zwischen Detektionszuverlässigkeit und Fehlervermeidung dar. Ein höherer Schwellenwert führte zu vermehrten Fehlklassifikationen durch unterlassene Detektionen relevanter Objekte, während ein niedrigerer Wert die Anzahl falsch-positiver Treffer signifikant erhöhte. Der gewählte Schwellenwert erlaubt somit eine stabile Raddetektion bei gleichzeitig hoher Präzision.
Im direkten Vergleich mit der Fahrzeuginstanzsegmentierung durch OneFormer zeigt sich, dass die von LangSAM bereitgestellten Confidence Scores eine geringere Trennschärfe zwischen relevanten und irrelevanten Objekten aufweisen. Aus diesem Grund ist LangSAM nicht als alleinige Grundlage für die Fahrzeugklassifikation vorgesehen. Die eigentliche Klassifizierung der Fahrzeuginstanzen erfolgt weiterhin ausschließlich auf Basis der durch OneFormer detektierten Objekte. LangSAM wird ausschließlich zur ergänzenden Extraktion von Radinformationen eingesetzt.
Auch durch die Wahl des Schwellenwerts $t_{CLangSAM}$ können falsch-positive Detektionen nicht vollständig ausgeschlossen werden. Eine weitere Erhöhung des Schwellenwerts würde zwar die Anzahl falsch-positiver Treffer reduzieren, führt jedoch gleichzeitig dazu, dass relevante Räder, insbesondere bei ungünstigen Aufnahmebedingungen, nicht mehr zuverlässig detektiert werden. Eine rein schwellenwertbasierte Filterung erweist sich daher als nicht ausreichend, um eine robuste Raddetektion sicherzustellen.
Aus diesem Grund wird im Rahmen der Raddetektion eine Region of Interest eingeführt, die ausschließlich den Bildbereich umfasst, in dem das Auftreten von Fahr-

zeugrädern plausibel ist. Diese Region of Interest wird auf Basis der zuvor detektierten Fahrzeuginstanzen definiert und begrenzt die Auswertung auf den unteren Bereich der jeweiligen Fahrzeugboundingbox. Dadurch werden irrelevante Bildbereiche systematisch ausgeschlossen und die Wahrscheinlichkeit falsch-positiver Detektionen weiter reduziert.
Durch die Kombination aus schwellenwertbasierter Filterung und räumlicher Einschränkung mittels Region of Interest kann die Stabilität der Raddetektion signifikant erhöht werden. Verbleibende Fehlklassifikationen werden in den nachgelagerten Verarbeitungsschritten der achsbasierten Klassifikation durch zeitliche Konsistenzkriterien berücksichtigt.

Entwicklung der Systemarchitektur Die Entwicklung der Systemarchitektur erfolgte in zwei aufeinander aufbauenden Phasen. Ziel war es, die bestehende klassische Verkehrszählung schrittweise zu erweitern, um zusätzliche Informationen aus der Raddetektion zu integrieren, ohne die Robustheit der etablierten Detektions- und Trackingkomponenten zu beeinträchtigen. Diese phasenweise Vorgehensweise ermöglichte eine kontrollierte Integration neuer Funktionalitäten und eine systematische Bewertung der jeweiligen Auswirkungen auf die Gesamtpipeline.
In Phase 1 wurde die klassische Verkehrszählung um die Berücksichtigung von Raddetektionen erweitert. Der Fokus lag dabei zunächst auf der Klassifikation von Fahrzeugen ausschließlich anhand der erkannten Achsanzahl, ohne eine weitergehende Unterscheidung nach konkreten Achskonfigurationen vorzunehmen. Für die Entwicklung und Validierung dieser Erweiterung wurden weiterhin ausschließlich die Fahrzeugklassen Pkw, Lkw und Bus berücksichtigt, um eine direkte Vergleichbarkeit mit der klassischen Zählung sicherzustellen und die Komplexität in der frühen Entwicklungsphase zu begrenzen.
Die Erweiterung der klassischen Verkehrszählung erfolgte durch die Integration der zusätzlichen Komponente LangSAM zur Raddetektion. Aufgrund der Arbeitsweise von ByteTrack war es jedoch nicht möglich, die Ausgaben der Fahrzeugdetektion und der Raddetektion innerhalb einer einzelnen ByteTrack-Instanz gemeinsam zu verarbeiten. Der Hauptgrund hierfür liegt in den signifikanten Unterschieden zwischen den Confidence Scores der Fahrzeugdetektionen und denen der Raddetektionen, welche zu instabilen Assoziationen und einer erhöhten Anfälligkeit gegenüber Fehlverknüpfungen geführt hätten.

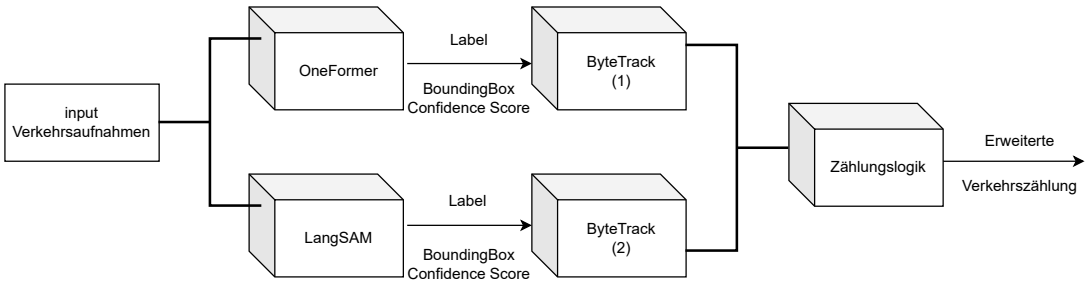


Abbildung 3.4: Systemarchitektur der erweiterten Verkehrszählung

Eine mögliche Lösung hätte in einer Anpassung von ByteTrack zur gemeinsamen Verarbeitung beider Detektionsquellen bestanden. Eine solche Modifikation hätte jedoch den positiven Effekt abgeschwächt, dass ByteTrack Detektionen unterhalb des jeweiligen Confidence Schwellwertes grundsätzlich verwirft, selbst wenn diese über einen längeren Zeitraum konsistent auftreten. Diese Filtereigenschaft ist insbesondere für die Reduktion von Falschdetektionen von zentraler Bedeutung und sollte daher erhalten bleiben.
Aus diesem Grund werden OneFormer und LangSAM jeweils mit einer eigenen, nachgelagerten ByteTrack-Instanz betrieben, sodass Fahrzeugdetektionen und Raddetektionen unabhängig voneinander getrackt werden. Als dritte Systemkomponente wurde eine eigene Zuordnungslogik entwickelt, die es ermöglicht, den Track-IDs der Fahrzeuge diejenigen Track-IDs der Räder zuzuordnen, die tatsächlich zu dem jeweiligen Fahrzeug gehören.
Als Zuordnungskriterium wird die prozentuale Schnittmenge zwischen der Bounding Box eines Rads und der Bounding Box eines Fahrzeugs verwendet, normiert über die Fläche der Rad-Bounding-Box. Auf Basis empirischer Untersuchungen hat sich hierfür ein Schwellenwert von $t_{FR} = 0{,}5$ als geeignet erwiesen. Dieses Kriterium erlaubt eine robuste Zuordnung von Raddetektionen zu den entsprechenden Fahrzeuginstanzen auch bei teilweiser Überlappung oder variierenden Perspektiven.
Durch diese Architektur wird die Klassifikation der Fahrzeuge nach der Achsanzahl ermöglicht, ohne die bestehende Trackingstruktur wesentlich zu verändern. Für die Realisierung wurden die bestehenden Fahrzeuglisten entsprechend erweitert, um die zusätzlichen Radzuordnungen zu verwalten und in die nachgelagerten Klassifikationsschritte einzubinden. Auf diese Weise kann die erweiterte Zählungslogik umgesetzt werden, während die bewährte Filtereigenschaft von ByteTrack zur Unterdrückung von Falschdetektionen vollständig erhalten bleibt.

In Phase 2 erfolgte die Implementierung der Funktion zur Identifikation der Achskonfiguration sowie der anschließenden Klassifizierung der Kombination aus Fahrzeuginstanz und erkannter Achskonfiguration. Damit wird die in Phase 1 eingeführte Klassifikation nach reiner Achsanzahl um eine differenziertere Betrachtung der relativen Achsanordnung erweitert. Bei der Umsetzung traten mehrere grundlegende methodische Herausforderungen auf. Ohne eine geometrische Referenz ist aus der zweidimensionalen Bildebene keine zuverlässige Aussage über reale dreidimensionale Weltkoordinaten möglich, selbst wenn die intrinsischen und extrinsischen Kameraparameter bekannt wären. Für den verwendeten Testdatensatz könnte die intrinsische Kameramatrix zwar näherungsweise bestimmt werden, sie würde jedoch unvermeidlich Abweichungen enthalten und wäre damit nur eingeschränkt zuverlässig. Eine Bestimmung der extrinsischen Kameramatrix ist hingegen für die vorliegende Datenbasis nicht möglich.
Darüber hinaus soll das entwickelte System nicht auf eine feste Bildauflösung beschränkt sein. Der direkte euklidische nach [41]

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3.1}$$

zwischen Objekten in der zweidimensionalen Bildebene stellt daher keine geeignete Metrik dar, da dieser unmittelbar von der gewählten Auflösung abhängt. Eine solche Metrik würde erfordern, die zugehörigen Schwellwerte für jede Auflösung erneut empirisch zu bestimmen. Diese müssten entweder fest im Quellcode hinterlegt oder beim Systemstart über frei wählbare Parameter übergeben werden. Beide Ansätze beeinträchtigen die Nutzerfreundlichkeit und würden zudem die Wartbarkeit des Quellcodes reduzieren, da die Anzahl konfigurierbarer Parameter steigt und die Systemkonfiguration zunehmend unübersichtlich wird.
Zur Lösung dieser Problematik wurde ein auf Relationen basierender Ansatz gewählt. Der Abstand zwischen zwei Achsen wird bestimmt, indem der euklidische Abstand zwischen den Mittelpunkten der zugehörigen Rad-Bounding-Boxen $\mathbf{c}_i$ und $\mathbf{c}_j$ in der zweidimensionalen Bildebene über die summierte Fläche der jeweiligen Bounding Boxen A_i und A_j normiert wird. Durch diese Normierung wird eine auflösungsunabhängige Metrik eingeführt, die eine präzise Aussage über relative Achsabstände ermöglicht, unabhängig von der absoluten Distanz zwischen Fahrzeug und Kamera.

$$d_{ij}^{\text{norm}} = \frac{\|\mathbf{c}_i - \mathbf{c}_j\|_2}{A_i + A_j} = \frac{\sqrt{(x_i^{(c)} - x_j^{(c)})^2 + (y_i^{(c)} - y_j^{(c)})^2}}{A_i + A_j} \tag{3.2}$$

Die gewählte Metrik ist zudem auf andere Bildauflösungen übertragbar, da lediglich das Verhältnis α zwischen der verwendeten Referenzauflösung und der jeweiligen Nutzerauflösung berücksichtigt werden muss. Auf diese Weise kann das Verfahren auf unterschiedliche Kamerasysteme und Auflösungen angewendet werden, ohne dass die Schwellwerte neu bestimmt werden müssen.

$$\alpha = \frac{Nutzerauflösung}{1280 * 720} \tag{3.3}$$

Damit folgt die resultierende Gleichung zur Bestimmung der Achsabstände in der 2d-Bildebene:

$$Achsabstand = \alpha \cdot d_{ij}^{\text{norm}} \tag{3.4}$$

Die Bestimmung der Achsabstände erfolgt zunächst unabhängig von den Fahrzeuginstanzen. Es werden ausschließlich die Abstände zwischen aufeinanderfolgenden, getrackten Rädern berechnet. Die resultierenden ID-Paarungen der Räder werden anschließend der jeweiligen Fahrzeug-ID zugeordnet, sofern das Fahrzeug beide Rad-Track-IDs in seinen zugeordneten Raddetektionen enthält.
Aus der Kombination der zugeordneten Radpaare und der daraus abgeleiteten Achsabstände wird mithilfe eines definierten Mappings die jeweilige Achskonfiguration bestimmt. In Verbindung mit der zuvor ermittelten Fahrzeuginstanz ermöglicht dies die Klassifizierung der Kombination aus Fahrzeugtyp und Achskonfiguration. Durch dieses modulare Vorgehen konnte die in Phase 1 entwickelte Architektur gezielt erweitert werden, ohne deren grundlegende Struktur zu verändern.

3.3 Evaluation und Ergebnisse

Die Evaluation des entwickelten Systems stellt besondere methodische Anforderungen, da die verfügbare reale Testdatenbasis nur eingeschränkt zur umfassenden Bewertung der Systemleistung geeignet ist. Die vorliegenden Testdaten weisen eine geringe Bildqualität auf und enthalten zudem nicht alle für die erweiterte Verkehrszählung relevanten Fahrzeug- und Achskonfigurationen in ausreichender Anzahl. Unter diesen Randbedingungen ist eine belastbare quantitative Bewertung der Gesamtperformance des Systems auf Basis realer Daten nur eingeschränkt möglich.
Um dennoch eine systematische und vergleichbare Bewertung der entwickelten Methodik zu ermöglichen, wurde ergänzend eine simulationsbasierte Evaluationsumgebung eingesetzt. Diese ermöglicht die kontrollierte Erzeugung synthetischer Bilddaten sowie die Bereitstellung konsistenter Ground-Truth-Informationen für die Instance Segmentation. Auf diese Weise können die entwickelten Verfahren unter reproduzierbaren Randbedingungen untersucht werden, ohne durch die Limitierungen der realen Testdaten dominiert zu werden.
Die simulationsbasierte Evaluation wurde in IsaacLab [42] realisiert und dient der gezielten Analyse der Raddetektion, der Achsabstandsbestimmung sowie der darauf aufbauenden Klassifikation. Somit konnte ein verlässlicher Schwellwert $t_{eA} = 0.07$ aus den Versuchen in IsaacLab bestimmt werden. Die konkrete Ausgestaltung der Evaluationsstrategie und die verwendeten Datenquellen werden im folgenden Abschnitt zur Validierungsmethodik beschrieben. Die anschließenden Ergebnisabschnitte präsentieren die Resultate getrennt für die einzelnen Systemkomponenten sowie für das Gesamtsystem.

3.3.1 Validierungsmethodik

Die Validierung des entwickelten Systems erfolgt entlang der einzelnen Systemkomponenten sowie auf Ebene der Gesamtsystemfunktionalität. Ziel ist es, sowohl die Leistungsfähigkeit der Fahrzeugdetektion als auch der Raddetektion sowie die Korrektheit der darauf aufbauenden achsbasierten Klassifikation getrennt voneinander zu bewerten. Durch diese modulare Evaluationsstrategie kann sichergestellt werden, dass Einschränkungen einzelner Komponenten nicht fälschlicherweise als systemische Schwächen interpretiert werden.
Aufgrund der begrenzten Qualität und Abdeckung der verfügbaren realen Testdaten wird eine zweigleisige Evaluationsstrategie verfolgt. Reale Bilddaten werden primär zur

Plausibilisierung der Fahrzeuginstanzdetektion mit OneFormer sowie zur qualitativen Bewertung der Raddetektion mit LangSAM herangezogen. Eine belastbare quantitative Validierung der achsbasierten Klassifikation ist auf Basis dieser Daten jedoch nur eingeschränkt möglich, da nicht alle relevanten Fahrzeug- und Achskonfigurationen enthalten sind und die Bildqualität die Segmentierung signifikant beeinflusst.
Zur systematischen und reproduzierbaren Validierung der entwickelten Methodik wird daher ergänzend eine simulationsbasierte Evaluationsumgebung in IsaacLab eingesetzt. Diese ermöglicht die kontrollierte Erzeugung synthetischer RGB-Bilddaten sowie die gleichzeitige Bereitstellung perfekter Ground-Truth-Informationen für die Instance Segmentation der Räder. Die Simulation erlaubt es, alle relevanten Klassen und Achskonfigurationen unter identischen Randbedingungen abzubilden und die entwickelte Architektur unabhängig von den Limitierungen realer Testdaten zu untersuchen.
Für die simulationsbasierte Evaluation wird ein abstrahiertes Pseudofahrzeug verwendet, das ausschließlich aus modellierten Radobjekten besteht und mit konstanter Geschwindigkeit an der virtuellen Kamera vorbeibewegt wird. Die Fahrzeuginstanz selbst wird dabei als ideal erkannt angenommen, sodass die Validierung gezielt auf die Raddetektion, die Achsabstandsbestimmung sowie die Ableitung der Achskonfiguration fokussiert werden kann. Diese Annahme erlaubt eine isolierte Bewertung der entwickelten Achslogik, ohne dass die Ergebnisse durch Fehler in der Fahrzeuginstanzdetektion dominiert werden.
Die Bewertung der einzelnen Systemkomponenten erfolgt anhand geeigneter, aufgabenspezifischer Metriken. Für die Fahrzeuginstanzdetektion mit OneFormer werden etablierte Detektionsmetriken zur qualitativen und quantitativen Einordnung herangezogen. Für die Raddetektion mit LangSAM werden Precision- und Recall-basierte Kennzahlen sowie qualitative Bildanalysen verwendet. Die Validierung der achsbasierten Klassifikation erfolgt durch den Vergleich der abgeleiteten Achskonfigurationen mit den in der Simulation verfügbaren Ground-Truth-Informationen.
Durch diese Trennung zwischen realdatenbasierter Plausibilisierung und simulationsbasierter, kontrollierter Validierung kann die Leistungsfähigkeit der entwickelten Systemarchitektur strukturiert bewertet werden. Die nachfolgenden Ergebnisabschnitte präsentieren die Resultate für die einzelnen Komponenten sowie für das Gesamtsystem getrennt und im jeweiligen Anwendungskontext.

3.3.2 Ergebnisse der Fahrzeugerkennung und klassischen Verkehrszählung

Ein Vergleich der realen Fahrzeugzählung mit den durch das entwickelte System erfassten Ergebnissen zeigt eine Übereinstimmung in der Gesamtanzahl der detektierten Fahrzeuge. Sowohl manuell als auch systemseitig wurden insgesamt 127 Fahrzeuge erfasst. Jedoch weicht die Verteilung auf die einzelnen Klassen leicht voneinander ab: Während die reale Zählung 116 Autos, 5 Lkw und 6 Busse ergibt, weist das System 116 Autos, 6 Lkw und 5 Busse aus (vgl. Abbildung 3.5).

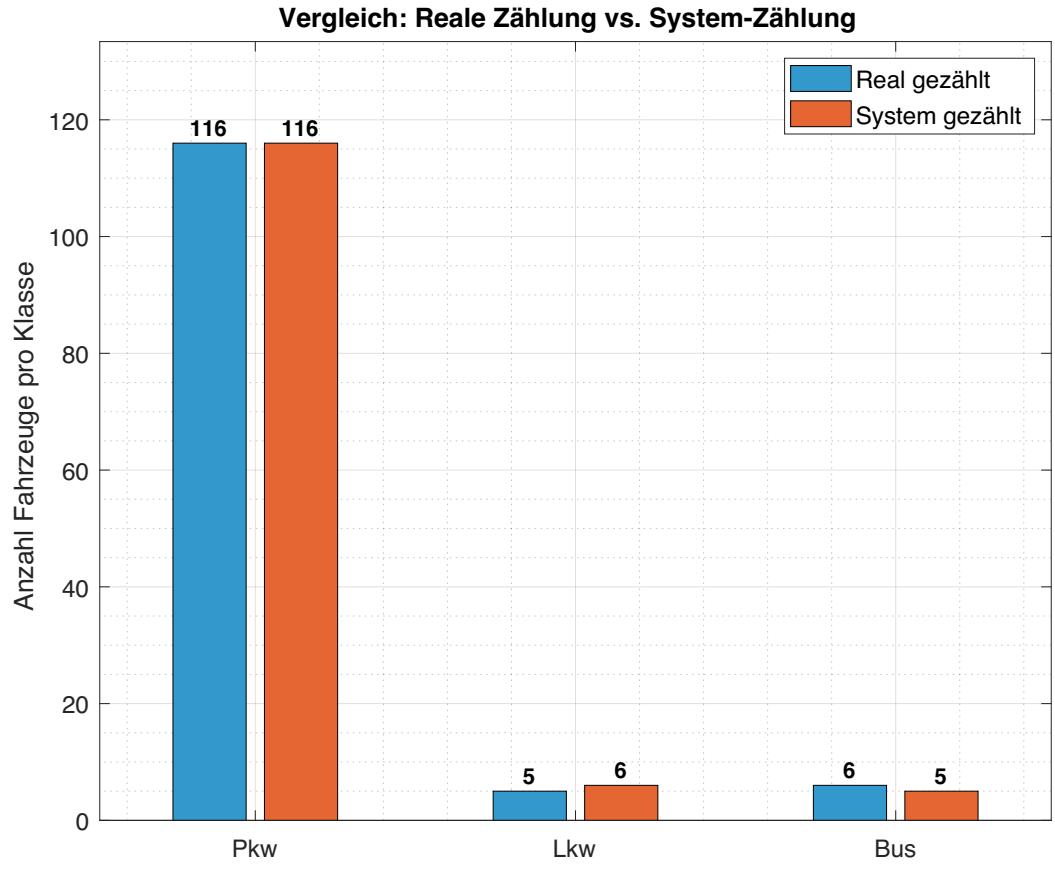


Abbildung 3.5: Beispielhafte Fehlklassifikationen infolge niedriger Bildauflösung und unklarer Fahrzeugmerkmale

Diese Abweichung lässt sich auf vier spezifische Fehlklassifikationen zurückführen, die im Rahmen der Auswertung identifiziert wurden: Zwei Pkw wurden vom System als Lkw klassifiziert, ein Lkw als Auto und ein Bus als Auto. Daraus resultieren eine Überschätzung der Klasse *Pkw* um zwei Instanzen, eine Überschätzung der *Lkw* um zwei Instanzen sowie eine Unterschätzung der *Busse* um eine Instanz.
Die Ursache für diese Fehlklassifikationen liegt vermutlich in der eingeschränkten Bildqualität des Testmaterials. Aufgrund geringer Auflösung und ungünstiger Kameraperspektiven verlieren segmentierte Objekte charakteristische Merkmale, was die Modellentscheidung erschwert. Besonders Fahrzeuge mit ähnlicher Silhouette, wie Busse und größere Vans, oder Lkw mit niedriger Bauhöhe, können unter diesen Bedingungen fehlerhaft zugeordnet werden.
Abbildung 3.6 zeigt exemplarisch zwei dieser Fehlklassifikationen. Die Bilder wurden aus dem Testlauf extrahiert und verdeutlichen die Schwierigkeit der korrekten Klassenzuordnung unter realen Bedingungen.

(a) Auto als Lkw erkannt (b) Bus als Auto erkannt

Abbildung 3.6: Beispielhafte Fehlklassifikationen infolge niedriger Bildauflösung und unklarer Fahrzeugmerkmale

Zur quantitativen Bewertung wird die Klassifikationsgenauigkeit pro Klasse und eine Konfusionsmatrix für die vollständige Zählung in 3.1 dargestellt.

Tabelle 3.1: Konfusionsmatrix der Systemklassifikation im Vergleich zur manuellen Zählung

Klassifiziert als	**Ground Truth**			**Klassifikationsgenauigkeit**
	Pkw	Lkw	Bus	
Pkw	116	1	1	98,3 %
Lkw	2	4	1	80,0 %
Bus	0	0	4	83,3 %

Die Ergebnisse der klassischen Verkehrszählung im Rahmen von Phase 1 lassen sich in Bezug auf vergleichbare Systeme aus der Literatur einordnen. Crouzil et al. [43] beschreiben ein bildbasiertes Verkehrserkennungssystem, das auf Kamerabildern mit guter Auflösung basiert und eine Einteilung in die Klassen *Light Vehicles* (LV), *Heavy Vehicles* (HV) sowie *Two-Wheelers* (TW) vornimmt. Da im vorliegenden System nur Fahrzeuge der Kategorien *Pkw*, *Lkw* und *Bus* berücksichtigt wurden, beschränkt sich die Einordnung auf die beiden Klassen LV und HV, wobei LV den Pkw und HV der Zusammenfassung von Lkw und Bussen entspricht. Im Unterschied zum genannten Verfahren wurde im hier untersuchten System auf Videomaterial mit deutlich geringerer Bildqualität zurückgegriffen, das jedoch aus einer niedrigeren Perspektive aufgenommen wurde.

(a) Pkw

(b) Lkw mit kleinem Fahrerhaus

(c) Transporter unter 3,5t

(d) Rettungswagen über 3,5t

(e) Gelenkbus auf dem Seitenstreifen

(f) Gelenkbus auf der Straße

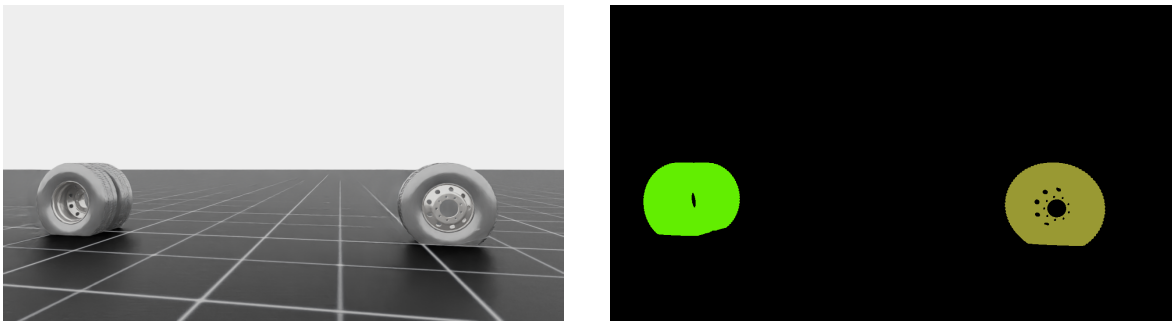
Abbildung 3.7: Beispiele korrekt klassifizierter Fahrzeuge durch das System. Die Identifikation von LKWs mit typischem Fahrerhaus gelingt zuverlässig. Auch ein aufgelasteter Rettungswagen wurde als *Lkw* eingeordnet, ob dies auf einer verallgemeinernden semantischen Einordnung oder lediglich äußerlich ähnlichen Merkmalen basiert, bleibt offen

Vor diesem Hintergrund sind die erzielten Ergebnisse als grundsätzlich vertretbar zu bewerten. Die hohe Klassifikationsgenauigkeit im Bereich der Kategorie *Pkw* und eine konsistente Gesamterkennung trotz der erschwerten Bedingungen deuten darauf hin, dass das System in der Lage ist, auch unter suboptimalen Bedingungen eine valide Zählung und Klassifikation durchzuführen. Gleichzeitig zeigt sich anhand der Fehleranalyse in den Klassen *Lkw* und *Bus*, dass insbesondere bei seltenen Klassen die Qualität der Segmentierung eine entscheidende Rolle spielt und das System in zukünftigen Iterationen gezielt in dieser Hinsicht optimiert werden kann.

3.3.3 Gesamtklassifikation und Verkehrszählung

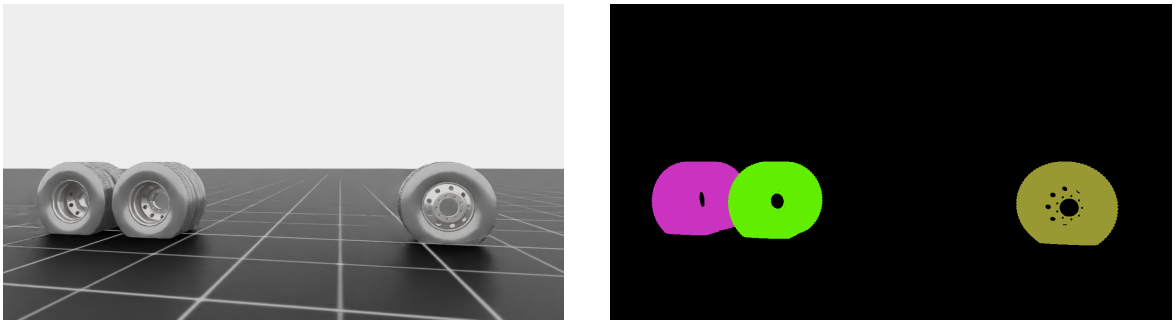
Zur Validierung der vollständigen Zähl- und Klassifikationslogik wurde eine simulationsbasierte Datengrundlage in IsaacLab aufgebaut. Ziel dieser Evaluation war die Analyse der Systemfunktionalität unter kontrollierten Bedingungen, insbesondere im Hinblick auf die korrekte Bestimmung der Achskonfigurationen auf Basis detektierter Radpositionen. Die Generierung synthetischer RGB-Bilddaten erfolgte innerhalb einer parametrisch definierten Szenenumgebung unter Verwendung eines internen Pinhole-Kameramodells. Die Bildauflösung betrug 1280×720 Pixel. Die Kamera war mit einer Simulationsschrittweite von 0,01 s im Weltkoordinatensystem der USD-Szene (Universal Scene Description) an der Position $[5,\ 4{,}2,\ 1]$ verortet und mathematisch positiv um $10\,^\circ$ um die z-Achse gedreht. Der Maßstab sämtlicher Objekte in der simulierten Umgebung entsprach einem Verhältnis von 1:1 zur realen Welt.
Die simulierten Fahrzeuginstanzen wurden dabei als ideal erkannt angenommen, mit einem festen Confidence Score von $1,0$. Eine Detektionsunsicherheit wurde nicht berücksichtigt, da das Hauptaugenmerk dieser Evaluation ausschließlich auf der Robustheit der Achsklassifikation lag. Entsprechend wurde auf eine explizite Fahrzeuggestalt verzichtet; das Pseudofahrzeug bestand ausschließlich aus modellierten Radobjekten, die in definierten Konfigurationen an der Kamera vorbeibewegt wurden.

Zur Bewertung der Klassifikationslogik wurden exemplarisch fünf verschiedene Achskonfigurationen simuliert: `1x1`, `1x2`, `1x3`, `2x1` und `2x2`, wobei die Notation der Anzahl an Achsgruppen und der Anzahl der Achsen pro Gruppe folgt. Basierend auf den erzeugten Bilddaten konnten normierte Achsabstände zwischen aufeinanderfolgenden Radpaaren berechnet und in Abhängigkeit ihrer relativen Distanz gruppiert werden. Durch empirische Analyse dieser Datenbasis ließ sich ein stabiler Schwellenwert zur Zuordnung von Achsgruppen identifizieren. Dies ermöglichte die zuverlässige Klassifikation der erkannten Fahrzeuginstanzen nach ihrer Achskonfiguration, einschließlich der Unterscheidung einzelner und mehrfacher Achsgruppen. Insgesamt konnte die grundlegende Funktionsfähigkeit der entwickelten Zähl- und Klassifikationskomponente auf Basis der simulationsbasierten Daten erfolgreich validiert werden.



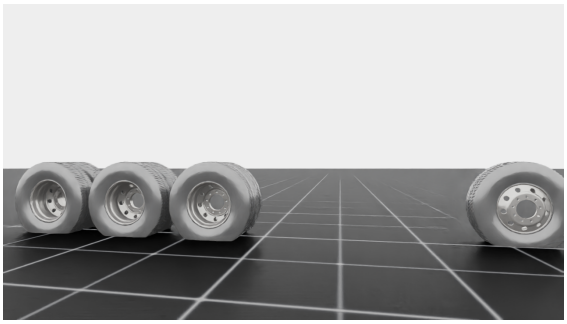
(a) Originalbild (b) Ground Truth

Abbildung 3.8: Synthetisches Bild einer Einfachachser Konfiguration (`1x1`) mit zugehöriger Ground-Truth-Maske

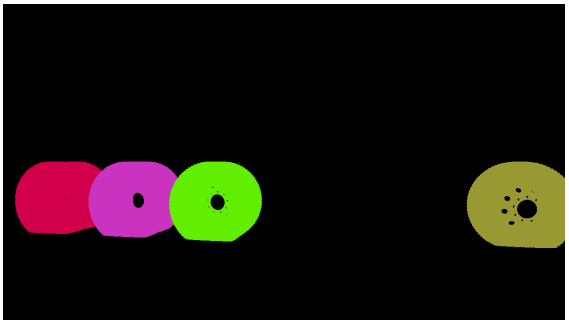


(a) Originalbild (b) Ground Truth

Abbildung 3.9: Synthetisches Bild einer Doppelachser Konfiguration (`1x2`) mit zugehöriger Ground-Truth-Maske

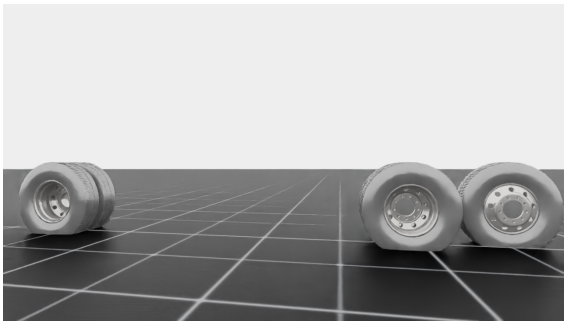


(a) Originalbild

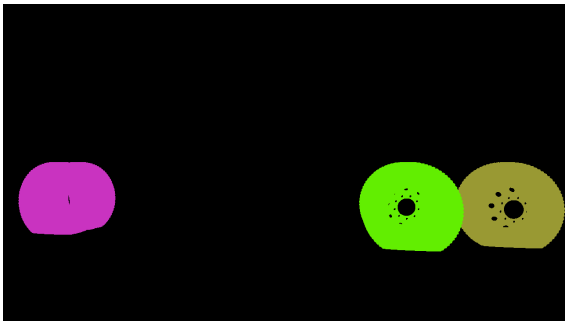


(b) Ground Truth

Abbildung 3.10: Synthetisches Bild einer Dreifachachser Konfiguration (`1x3`) mit zugehöriger Ground-Truth-Maske

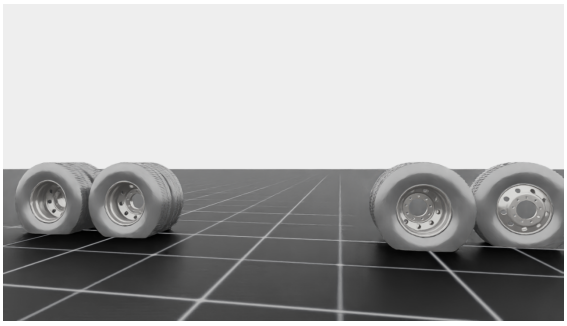


(a) Originalbild

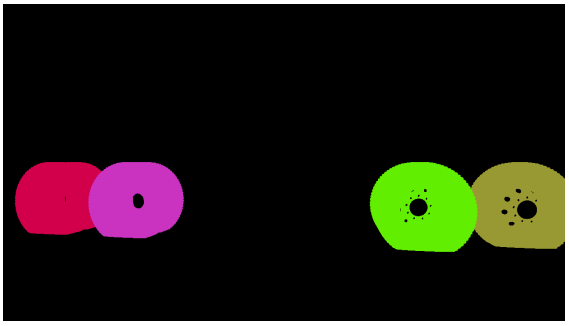


(b) Ground Truth

Abbildung 3.11: Synthetisches Bild einer Doppelachser Konfiguration (`2x1`) mit zugehöriger Ground-Truth-Maske



(a) Originalbild



(b) Ground Truth

Abbildung 3.12: Synthetisches Bild einer Doppelachser Konfiguration (`2x2`) mit zugehöriger Ground-Truth-Maske

Zur weiteren Plausibilisierung der entwickelten Zähl- und Klassifikationslogik wurde eine ergänzende Erprobung auf Basis realer Kamerabilder durchgeführt. Im Gegensatz zur simulationsbasierten Umgebung in IsaacLab unterlagen diese Daten deutlich stärkeren Einschränkungen hinsichtlich Auflösung, Bildqualität sowie der Kamerapositionierung. Insbesondere die wechselhafte Detektion einzelner Räder, kombiniert mit teils deutlich verschobenen Bounding Boxes und fehlerhaften Fahrzeugklassifikationen, führte dazu, dass keine einzige Achskonfiguration korrekt identifiziert werden konnte. Ein zentrales Problem stellte hierbei die Sequenzierung der detektierten Räder dar. Die Zähllogik geht davon aus, dass eine lineare Reihenfolge der Räder entlang der Fahrtrichtung rekonstruierbar ist. Wird jedoch ein einzelnes Rad später erkannt als jene Räder, die geometrisch hinter ihm liegen, so führt dies zu einer fehlerhaften Gruppierung innerhalb der Achsklassifikation. Diese Problematik zeigte sich wiederholt bei den realen Aufnahmen und führte in allen Fällen zu einer inkorrekten Zuordnung der Achskonfiguration.
Die eingesetzte Logik selbst zeigte im simulierten Umfeld eine konsistent korrekte Klassifikation aller getesteten Konfigurationen, sofern die Detektionsqualität und die räumliche Staffelung der Radpositionen gegeben waren, bis auf die Dreifachachser-Klasse mit der Achskonfiguration `1x3`. Dies unterstreicht, dass die aktuelle Methodik grundsätzlich funktional ist, jedoch eine gewisse Sensitivität gegenüber Eingabestörungen aufweist. Unter realen Bedingungen könnte eine verbesserte Kamerapositionierung sowie eine robustere Vorverarbeitung der Detektionen (z.,B. durch Filterung von verschobenen Bounding Boxen oder das abfangen verspäteter Detektion und ihre korrekte Einordnung) die Grundlage für eine zuverlässigere Anwendung schaffen.
Es bleibt festzuhalten, dass die untere Grenze für eine erfolgreiche Klassifikation bei einer Auflösung von 1280×720 Pixeln liegt. In dieser Konfiguration gelang in der Simulation bis auf eine Ausnahme die vollständige und korrekte Zuordnung aller Achsstrukturen. Somit konnte die prinzipielle Eignung der entwickelten Zähllogik für realitätsnahe Anwendungsfälle bestätigt werden, wenngleich deren Robustheit unter realen Bedingungen gegenwärtig noch limitiert ist.

3.3.4 Diskussion der Ergebnisse

Die Ergebnisse der ersten Evaluationsphase zur klassischen Verkehrszählung zeigen ein grundsätzlich funktionsfähiges System mit ausbaufähiger Genauigkeit. Die Abweichungen zur manuellen Referenzzählung lassen sich im Wesentlichen auf drei Faktoren zurückführen: die begrenzte Bildauflösung, die gewählte Kameraperspektive sowie die geometrische Krümmung im Bildfeld der Kamera. Diese Kombination erschwert die exakte Segmentierung der Fahrzeuge, insbesondere bei perspektivisch verzerrten Objektansichten. Zudem konnte bei der Validierung der erweiterten Verkehrszählung beobachtet werden, dass das Segmentierungsmodell LangSAM teilweise keine oder falsch platzierte Radsegmente detektierte (Abbildung 3.14), auch bei ansonsten klar erkennbaren Fahrzeugen, wie in Abbildung 3.13. Eine mögliche Ursache könnten implizite Detektionslücken in der Modellstruktur sein, etwa durch Einschränkungen in der Spatial-Attention-Mechanik oder suboptimale Feature Maps. Eine weitergehende Analyse der Modellarchitektur lag jedoch außerhalb des Umfangs dieser Arbeit.

Abbildung 3.13: Lkw wurde als *Pkw* klassifiziert und weist eine stark verschobene Bounding Box des Vorderrads auf, während das Hinterrad nicht erkannt wird

Abbildung 3.14: Falschdetektion eines Rads bei einem Bus am Bildrand

Auch im Bereich der Fahrzeugklassifikation kam es zu systematischen Fehlern. Zwei Aufnahmen derselben Gelenkbus-Klasse zeigen, dass trotz identischer Position im Bildfeld die Formgebung eine erhebliche Rolle spielt: Während das kantigere Modell (Abbildung 3.16) korrekt als Bus klassifiziert wurde, wurde das Rundlichere (Abbildung 3.17) als Pkw erkannt. Neben solchen Formeinflüssen könnten auch proportionale oder markenspezifische Merkmale eine Rolle spielen. So wurde etwa eine Mercedes V-Klasse als *Lkw* klassifiziert (Abbildung 3.6a).

(a) Transporter Kastenausführung korrekt als *Pkw* klassifiziert

(b) Transporter mit Kofferaufbau korrekt als *Pkw* klassifiziert

Abbildung 3.15: Beide Ausführungsvarianten wurden korrekt der Klasse *Pkw* zugeordnet

Wie bereits erwähnt wurde ein Rettungswagen mit Kofferaufbau (Abbildung 3.7d) korrekt als *Lkw* klassifiziert, was möglicherweise auf Überhänge, Volumenverhältnisse oder fehlende Unterscheidungsmerkmale im Seitenbereich zurückzuführen ist, während andere Transporter (auch mit Kofferaufbau) ebenfalls richtig der Klasse *Pkw* zugeordnet wurden (Abbildung 3.15). Gleichzeitig wurden Lkw mit kleinem Fahrerhaus teilweise fälschlich als *Pkw* eingeordnet (Abbildung 3.13). Aufgrund der geringen Anzahl an untersuchten Spezialfällen lassen sich diese Beobachtungen derzeit jedoch nicht verallgemeinern und müssen in zukünftigen Studien vertieft werden.

Abbildung 3.16: Gelenkbus mit kantiger Silhouette bei großer Entfernung richtig Klassifiziert. Räder nicht in richtiger Reihenfolge, aber vollständig und präzise erkannt

Abbildung 3.17: Gelenkbus mit runder Silhouette falsch als *Pkw* klassifiziert. Räder nicht in richtiger Reihenfolge, aber vollständig und präzise erkannt

Die eingeschränkte Bildqualität wirkte sich besonders deutlich auf die Erkennung kleinerer Objekte wie Fahrzeugräder aus. Dabei wurden einzelne Räder in bestimmten Fällen nicht nur fehlerhaft, sondern vollständig übersehen. Auffällig ist, dass in Aufnahmen von weiter entfernten Fahrzeugen (Abbildungen 3.16, 3.17) die Radsegmente korrekt, wenngleich in falscher Reihenfolge, erkannt wurden, während bei einem näheren Fahrzeug desselben Typs (Abbildung 3.18) keine Raderkennung erfolgte. Diese Diskrepanz legt nahe, dass nicht allein die absolute Pixelauflösung entscheidend ist. Vielmehr scheint ein Zusammenspiel aus Objektgröße, Position im Bild und interner Aufmerksamkeitsverteilung innerhalb des Modells die Segmentierung zu beeinflussen. Diese Zusammenhänge sollten in künftigen Arbeiten gezielt untersucht werden.

(a) Vorderrad wird nicht detektiert

(b) mittleres Rad wird nicht detektiert

(c) Hinterrad wird nicht detektiert

Abbildung 3.18: Räder des Gelenkbusses werden nicht detektiert, obwohl die Auflösung ausreichend ist

Die derzeit implementierte Zähllogik zeigt unter realen Bedingungen erhebliche Schwächen. Insbesondere bei nicht idealer Radanordnung oder fehlender Segmentierung scheitert die Gruppierung der Achsen, was in einer fehlerhaften Klassifikation resultiert. Unter den idealisierten Bedingungen der Simulation in IsaacLab konnten sämtliche getesteten Achskonfigurationen zuverlässig unterschieden werden (Ausnahme `1x3` Achskonfiguration). Für eine robuste Anwendung unter realen Bedingungen müssten jedoch zusätzliche logische Prüfmechanismen entwickelt werden. Durch den modularen Aufbau des Systems besteht die Möglichkeit, sowohl die verwendeten Segmentierungsmodelle OneFormer und LangSAM als auch die Zähllogik selbst durch leistungsfähigere Komponenten zu ersetzen oder zu erweitern. Neben einer besseren Segmentierung wäre insbesondere eine fehlertolerantere Klassifikationsstrategie erforderlich, um das System an reale Herausforderungen anzupassen.

4 Zusammenfassung und Ausblick

In diesem Kapitel werden die wesentlichen Inhalte und Ergebnisse der vorliegenden Arbeit zusammengefasst. Darüber hinaus werden die eigenen wissenschaftlichen Beiträge dieser Arbeit herausgestellt. Abschließend wird ein Ausblick auf mögliche Weiterentwicklungen und zukünftige Arbeiten gegeben, die sich aus den vorgestellten Ergebnissen ergeben.

4.1 Zusammenfassung

In dieser Arbeit wurde ein bildbasiertes Verfahren zur erweiterten Verkehrszählung entwickelt und untersucht, das über die reine Fahrzeugzählung hinaus eine achsbasierte Fahrzeugklassifikation ermöglicht. Ziel war es, auf Grundlage moderner Detektions-, Segmentierungs- und Trackingverfahren Fahrzeuge nicht nur zu erfassen, sondern deren Achskonfiguration zuverlässig zu bestimmen. Hierzu wurde eine modulare Verarbeitungspipeline konzipiert, die klassische und lernbasierte Methoden kombiniert. Zunächst werden Fahrzeuge mithilfe lernbasierter Detektionsverfahren identifiziert und über ein Multi-Object-Tracking verfolgt. Zur präziseren Analyse der Fahrzeugstruktur werden zusätzlich Reifen mittels textbasierter Detektion und Segmentierung erkannt, deren Ergebnisse in eine eigens entwickelte Logik zur Zuordnung von Rädern zu Fahrzeugen integriert werden. Auf Basis der räumlichen Anordnung der detektierten Räder wird eine achsbasierte Klassifikation durchgeführt, bei der sowohl Einzel- als auch Mehrfachachsen berücksichtigt werden. Hierzu wurden geeignete Distanzmetriken sowie Normalisierungsstrategien definiert, um eine robuste und auflösungsunabhängige Bewertung der Achsabstände zu ermöglichen. Die Kombination aus Tracking, Raddetektion und strukturierter Auswertung erlaubt eine erweiterte Verkehrsanalyse, die über klassische Zählsysteme hinausgeht. Die Ergebnisse zeigen, dass der entwickelte Ansatz grundsätzlich geeignet ist, eine feinere Fahrzeugklassifikation auf Basis bildbasierter Daten zu realisieren und eine Grundlage schafft, um bildbasierte Verkehrszählsysteme um strukturelle Fahrzeugmerkmale zu erweitern.

4.2 Eigene wissenschaftliche Beiträge

Im Rahmen dieser Arbeit wurden mehrere eigenständige methodische und konzeptionelle Beiträge zur bildbasierten Verkehrsüberwachung erarbeitet. Ein zentraler Beitrag ist die Konzeption und Implementierung einer modularen Verarbeitungspipeline, die lernbasierte Detektion, Multi-Object-Tracking und feinaufgelöste Segmentierung in einem einheitlichen System integriert. Die Kombination dieser Komponenten ermöglicht eine durchgängige Verarbeitung von der Fahrzeugdetektion bis zur strukturellen Analyse der Fahrzeuggeometrie. Ein weiterer wesentlicher Beitrag ist die Entwicklung eines Verfahrens zur achsbasierten Fahrzeugklassifikation auf Grundlage detektierter und segmentierter Räder. Hierzu wurde eine eigene Logik zur Zuordnung von Rädern zu einzelnen Fahrzeugen implementiert, die Tracking-Informationen, geometrische Kriterien und zeitliche Konsistenz berücksichtigt. Darüber hinaus wurde eine normierte Distanzmetrik zur Bewertung von Achsabständen eingeführt, die den euklidischen Abstand zwischen Rad-Bounding-Box-Zentren mit der Fläche der jeweiligen Bounding Boxen normalisiert. Diese Metrik ermöglicht eine auflösungsunabhängige und robuste Bewertung räumlicher Abstände in der Bildebene. Ein weiterer Beitrag besteht in der praktischen Integration und Anpassung moderner Open-Source-Modelle für die Anwendung im Kontext der Verkehrszählung. Insbesondere wurde eine individuelle Pipeline auf Basis von Oneformer, LangSAM und ByteTrack realisiert und für die spezifische Aufgabe der achsbasierten Fahrzeugklassifikation angepasst und erweitert. Zusammenfassend leistet diese Arbeit einen Beitrag zur Erweiterung klassischer bildbasierter Verkehrszählsysteme um strukturierte, achsbasierte Fahrzeugmerkmale und zeigt die praktische Umsetzbarkeit moderner Detektions- und Segmentierungsverfahren für weiterführende verkehrstechnische Analyseaufgaben.

4.3 Ausblick

Im Rahmen dieser Arbeit zeigte sich, dass das eingesetzte textbasierten Segmentierungsverfahren von LangSAM in realen Verkehrsszenarien nicht in allen Fällen die gewünschte Robustheit aufwiesen. Insbesondere bei schnell bewegten Objekten traten Fehldetektionen auf, was auf Limitierungen im Zusammenspiel von Bildauflösung, Bildrate und Modellanpassung schließen lässt. Eine systematische Untersuchung mit erhöhter Bildrate bei gleichbleibender Auflösung könnte hierzu weitere Erkenntnisse liefern. Im Gegensatz dazu erzielten die verwendeten Verfahren in simulierten Szenarien sehr zuverlässige Ergebnisse. Die Simulation erfolgte mit moderater Auflösung und ohne fotorealistische Darstellung, was darauf hindeutet, dass die kontrollierten visuellen Bedingungen einen positiven Einfluss auf die Detektions- und Segmentierungsleistung haben. Dies unterstreicht die Bedeutung von Simulationen als Entwicklungs- und Testumgebung, zeigt jedoch zugleich die Herausforderungen bei der Übertragung auf reale Verkehrsdaten. Die rasche Weiterentwicklung von Visual Language Models deutet darauf hin, dass zukünftige Modellgenerationen eine deutlich verbesserte Generalisierungsfähigkeit aufweisen werden. Erste qualitative Tests mit einer neueren Modellgeneration Segment Anything Model 3 (SAM 3) [44] zeigten, dass identische Text-Prompts in mehreren zuvor fehlgeschlagenen Beispielen erfolgreich zur Raddetektion führten (Abbildung 4.1). Aufgrund des späten Erscheinens von SAM 3 konnte eine vollständige Integration und systematische Evaluation jedoch nicht mehr im Rahmen dieser Arbeit erfolgen.

(a) Vorderrad wird detektiert

(b) mittleres Rad wird detektiert

(c) Hinterrad wird detektiert

Abbildung 4.1: Räder des Gelenkbusses werden nun durch SAM 3 zufriedenstellend detektiert (rosa)

Insgesamt verdeutlichen diese Beobachtungen das hohe Potenzial moderner Visual Language Modelle (VLM) auch außerhalb strukturierter Verkehrsanalyseaufgaben. Gleichzeitig wird deutlich, dass die praktische Einsetzbarkeit solcher Modelle in realen Verkehrsszenarien maßgeblich von Faktoren wie Bildrate, Bildqualität und Modellgeneration abhängt. Zukünftige Arbeiten können hier ansetzen, um die Leistungsfähigkeit von VLMs für den Einsatz in realen Verkehrszählsystemen weiter zu untersuchen und nutzbar zu machen.

Anhang

Abbildung 4.2: Alle Fahrzeugklassen

Literatur

[1] G. Di Leo, A. Pietrosanto und P. Sommella, „Metrological Performance of Traffic Detection Systems,“ *IEEE Transactions on Instrumentation and Measurement*, Jg. 58, Nr. 9, S. 3199–3206, 2009. DOI: 10.1109/TIM.2009.2017158.

[2] Z. Chen, T. Ellis und S. A. Velastin, „Vehicle detection, tracking and classification in urban traffic,“ in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, 2012, S. 951–956. DOI: 10.1109/ITSC.2012.6338852.

[3] Q. Li, R. Li, K. Ji und W. Dai, „Kalman Filter and Its Application,“ in *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, 2015, S. 74–77. DOI: 10.1109/ICINIS.2015.35.

[4] V. K. Shopov und V. D. Markova, „Application of Hungarian Algorithm for Assignment Problem,“ in *2021 International Conference on Information Technologies (InfoTech)*, 2021, S. 1–4. DOI: 10.1109/InfoTech52438.2021.9548600.

[5] P. Bharati und A. Pramanik, „Deep Learning Techniques—R-CNN to Mask R-CNN: A Survey,“ in *Computational Intelligence in Pattern Recognition*, A. K. Das, J. Nayak, B. Naik, S. K. Pati und D. Pelusi, Hrsg., Singapore: Springer, 2020, S. 657–668, ISBN: 978-981-13-9042-5. DOI: 10.1007/978-981-13-9042-5_56.

[6] M. K. Reza, A. Prater-Bennette und M. S. Asif, „MMSFormer: Multimodal Transformer for Material and Semantic Segmentation,“ *IEEE Open Journal of Signal Processing*, Jg. 5, S. 599–610, 2024, ISSN: 2644-1322. DOI: 10.1109/OJSP.2024.3389812.

[7] J. Jain, J. Li, M. Chiu, A. Hassani, N. Orlov und H. Shi, „OneFormer: One Transformer to Rule Universal Image Segmentation,“ in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, S. 2989–2998. DOI: 10.1109/CVPR52729.2023.00292.

[8] J. Redmon, S. Divvala, R. Girshick und A. Farhadi, „You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, S. 779–788. DOI: 10.1109/CVPR.2016.91.

[9] K.-J. Kim, P.-K. Kim, Y.-S. Chung und D.-H. Choi, „Multi-Scale Detector for Accurate Vehicle Detection in Traffic Surveillance Data," *IEEE Access*, Jg. 7, S. 78 311–78 319, 2019, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2922479.

[10] W. Li, „Analysis of Object Detection Performance Based on Faster R-CNN," *Journal of Physics: Conference Series*, Jg. 1827, Nr. 1, S. 012 085, März 2021. DOI: 10.1088/1742-6596/1827/1/012085.

[11] A. Kirillov, K. He, R. Girshick, C. Rother und P. Dollar, „Panoptic Segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Juni 2019, S. 9396–9405, ISBN: 978-1-7281-3293-8. DOI: 10.1109/CVPR.2019.00963.

[12] X. Xiong, Z. Wu, S. Tan u. a., „SAM2-UNet: Segment anything 2 makes strong encoder for natural and medical image segmentation," *Visual Intelligence*, Jg. 4, Nr. 1, S. 2, 13. Jan. 2026, ISSN: 2731-9008. DOI: 10.1007/s44267-025-00106-w.

[13] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen und K. H. Maier-Hein, „nnU-Net: A self-configuring method for deep learning-based biomedical image segmentation," *Nature Methods*, Jg. 18, Nr. 2, S. 203–211, Feb. 2021, ISSN: 1548-7105. DOI: 10.1038/s41592-020-01008-z.

[14] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy und A. L. Yuille, „DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Jg. 40, Nr. 4, S. 834–848, Apr. 2018, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2017.2699184.

[15] J. Hwang, I. Na, J. Kim und H. Park, „SAM2 for abdomen: One-shot and no finetuning," in *2024 9th International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, Bd. 9, 2024, S. 551–555. DOI: 10.1109/ICIIBMS62405.2024.10792815.

[16] N. Xu, W. Lin, X. Lu und Y. Wei, „Tracking," in *Video Object Tracking: Tasks, Datasets, and Methods*, N. Xu, W. Lin, X. Lu und Y. Wei, Hrsg., Cham: Springer Nature Switzerland, 2024, S. 3–115, ISBN: 978-3-031-44660-3. DOI: 10.1007/978-3-031-44660-3_2.

[17] N. Xu, W. Lin, X. Lu und Y. Wei, *Video Object Tracking: Tasks, Datasets, and Methods* (Synthesis Lectures on Computer Vision). Cham: Springer Nature Switzerland, 2024. DOI: 10.1007/978-3-031-44660-3.

[18] T. Meinhardt, A. Kirillov, L. Leal-Taixé und C. Feichtenhofer, „TrackFormer: Multi-Object Tracking with Transformers," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, S. 8834–8844. DOI: 10.1109/CVPR52688.2022.00864.

[19] H. K. Cheng, S. Wug Oh, B. Price, A. Schwing und J.-Y. Lee, „Tracking Anything with Decoupled Video Segmentation," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, S. 1316–1326. DOI: 10.1109/ICCV51070.2023.00127.

[20] S. Ren, K. He, R. Girshick und J. Sun. „Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." arXiv: 1506.01497 [cs]. (6. Jan. 2016), Adresse: http://arxiv.org/abs/1506.01497 (besucht am 20. 08. 2025), Vorveröffentlichung.

[21] Ultralytics. „YOLO11 NEU." (), Adresse: https://docs.ultralytics.com/de/models/yolo11 (besucht am 19. 08. 2025).

[22] B. Coifman, „Vehicle Re-Identification and Travel Time Measurement in Real-Time on Freeways Using Existing Loop Detector Infrastructure," *Transportation Research Record*, Jg. 1643, Nr. 1, S. 181–191, 1. Jan. 1998, ISSN: 0361-1981. DOI: 10.3141/1643-22. Adresse: https://doi.org/10.3141/1643-22 (besucht am 18. 08. 2025).

[23] L.-E. Y. Mimbela, L. A. Klein und United States. Joint Program Office for Intelligent Transportation Systems, „Summary of Vehicle Detection and Surveillance Technologies used in Intelligent Transportation Systems," 31. Aug. 2007. Adresse: /view/dot/50558 (besucht am 18. 08. 2025).

[24] A. Bewley, Z. Ge, L. Ott, F. Ramos und B. Upcroft, „Simple Online and Realtime Tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, S. 3464–3468. DOI: 10.1109/ICIP.2016.7533003.

[25] Y. Zhang, P. Sun, Y. Jiang u. a., „ByteTrack: Multi-object Tracking by Associating Every Detection Box," in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella und T. Hassner, Hrsg., Cham: Springer Nature Switzerland, 2022, S. 1–21, ISBN: 978-3-031-20047-2. DOI: 10.1007/978-3-031-20047-2_1.

[26] H. Gupta und O. P. Verma, „Monitoring and surveillance of urban road traffic using low altitude drone images: A deep learning approach," *Multimedia Tools and Applications*, Jg. 81, Nr. 14, S. 19 683–19 703, 1. Juni 2022, ISSN: 1573-7721. DOI: 10.1007/s11042-021-11146-x.

[27] M. Cordts, M. Omran, S. Ramos u. a., „The Cityscapes Dataset for Semantic Urban Scene Understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, S. 3213–3223. DOI: 10.1109/CVPR.2016.350.

[28] T.-Y. Lin, M. Maire, S. Belongie u. a., „Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele und T. Tuytelaars, Hrsg., Cham: Springer International Publishing, 2014, S. 740–755, ISBN: 978-3-319-10602-1. DOI: 10.1007/978-3-319-10602-1_48.

[29] A. Ess, B. Leibe und L. Van Gool, „Depth and Appearance for Mobile Scene Analysis," in *2007 IEEE 11th International Conference on Computer Vision*, 2007, S. 1–8. DOI: 10.1109/ICCV.2007.440909.

[30] Z. Lu, T. Wu, Y. Dai, W. Li und Z. Su, „Fine-Grained Metrics for Point Cloud Semantic Segmentation," in *Pattern Recognition and Computer Vision*, Z. Lin, M.-M. Cheng, R. He u. a., Hrsg., Singapore: Springer Nature, 2025, S. 232–245, ISBN: 978-981-97-8795-1. DOI: 10.1007/978-981-97-8795-1_16.

[31] S. Shajahan und T. Poovizhi, „A Novel Approach to Estimation Precision and Recall for Star Rating Online Customers Based on Negative Hotel Reviews using Multinomial Naive Bayes over Multischeme Classifier," in *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*, 2022, S. 1–6. DOI: 10.1109/ICBATS54253.2022.9759081.

[32] A. Milan, L. Leal-Taixe, I. Reid, S. Roth und K. Schindler. „MOT16: A Benchmark for Multi-Object Tracking." arXiv: 1603.00831 [cs]. (3. Mai 2016), Vorveröffentlichung.

[33] E. Ristani, F. Solera, R. Zou, R. Cucchiara und C. Tomasi, „Performance Measures and a Data Set for Multi-target, Multi-camera Tracking," in *Computer Vision – ECCV 2016 Workshops*, G. Hua und H. Jégou, Hrsg., Cham: Springer International Publishing, 2016, S. 17–35, ISBN: 978-3-319-48881-3. DOI: 10.1007/978-3-319-48881-3_2.

[34] J. Luiten, A. Osep, P. Dendorfer u. a., „HOTA: A Higher Order Metric for Evaluating Multi-object Tracking," *International Journal of Computer Vision*, Jg. 129, Nr. 2, S. 548–578, 1. Feb. 2021, ISSN: 1573-1405. DOI: 10.1007/s11263-020-01375-2.

[35] C. Mühleiß, T. Faul, D. Hu und R. Wörner, „Fahrzeugklassifizierung mittels Stereokamera – Untersuchung und Weiterentwicklung eines Prototyps unter Einsatz von künstlichen neuronalen Netzen, Deep Learning und Bildverarbeitung," in *Making Connected Mobility Work: Technische und betriebswirtschaftliche Aspekte*, H. Proff, Hrsg., Wiesbaden: Springer Fachmedien, 2021, S. 473–485, ISBN: 978-3-658-32266-3. DOI: 10.1007/978-3-658-32266-3_28.

[36] C. Jiang, Q. Zhou, Z. Mo u. a. „DNAT: Multi-scale Transformer with Dilated Neighborhood Attention for Image Classification." (2023).

[37] Z. Liu, Y. Lin, Y. Cao u. a., „Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, S. 9992–10 002. DOI: 10.1109/ICCV48922.2021.00986.

[38] S. Liu, Z. Zeng, T. Ren u. a., „Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection," in *Computer Vision – ECCV 2024*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler und G. Varol, Hrsg., Cham: Springer Nature Switzerland, 2025, S. 38–55, ISBN: 978-3-031-72970-6. DOI: 10.1007/978-3-031-72970-6_3.

[39] J. Devlin, M.-W. Chang, K. Lee und K. Toutanova, „BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran und T. Solorio, Hrsg., Minneapolis, Minnesota: Association for Computational Linguistics, Juni 2019, S. 4171–4186. DOI: 10.18653/v1/N19-1423.

[40] Y. Chen, B. Liu und L. Yuan, „PR-Deformable DETR: DETR for Remote Sensing Object Detection," *IEEE Geoscience and Remote Sensing Letters*, Jg. 21, S. 1–5, 2024. DOI: 10.1109/LGRS.2024.3483217.

[41] I. Dokmanic, R. Parhizkar, J. Ranieri und M. Vetterli, „Euclidean Distance Matrices: Essential theory, algorithms, and applications," *IEEE Signal Processing Magazine*, Jg. 32, Nr. 6, S. 12–30, 2015. DOI: 10.1109/MSP.2015.2398954.

[42] M. Mittal, C. Yu, Q. Yu u. a., „Orbit: A Unified Simulation Framework for Interactive Robot Learning Environments,“ *IEEE Robotics and Automation Letters*, Jg. 8, Nr. 6, S. 3740–3747, Juni 2023, ISSN: 2377-3766. DOI: 10.1109/LRA.2023.3270034.

[43] A. Crouzil, L. Khoudour, P. Valiere und D. N. Truong Cong, „Automatic Vehicle Counting System for Traffic Monitoring,“ in *ournal of Electronic Imaging, vol. 2*, 2016, S. 1–14. DOI: 10.1117/1.JEI.25.5.051207.

[44] N. Carion, L. Gustafson, Y.-T. Hu u. a. „SAM 3: Segment Anything with Concepts.“ arXiv: 2511.16719 [cs]. (20. Nov. 2025), Vorveröffentlichung.

Abbildungsverzeichnis

Selbstständigkeitserklärung

Die vorliegende Arbeit habe ich selbstständig ohne Benutzung anderer als der angegebenen Quellen angefertigt. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Quellen entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer oder anderer Prüfungen noch nicht vorgelegt worden.

Zwickau, den 29. Januar 2026 Leonard Kämpf