21. AALE-Konferenz
Dresden, 12.–14. März 2025

Erkennung und Zuordnung von Lichtsignalanlagen für autonome Fahrzeuge in urbanen Testsites

M. Kocsis[12], B. Hartmann[1], E. Heinrich[1], R. Zöllner[1], N. C. Stache[1]

Zusammenfassung

Die präzise und zuverlässige Erkennung von Lichtsignalanlagen ist eine zentrale Herausforderung für die Entwicklung autonomer Fahrzeuge, insbesondere in urbanen Räumen. Die vorliegende Arbeit befasst sich mit einer lernbasierten Erkennung von Lichtsignalanlagen und deren Zustand im urbanen Straßenverkehr. Weiterhin wird eine Zuordnung dieser Anlagen anhand einer topologischen Karte, der aktuellen Pose des Fahrzeugs und der geplanten Route durchgeführt. Die Evaluation wird auf realen Daten, welche mittels autonom fahrenden Shuttles in den Testsites Heilbronn und Bad Wimpfen aufgenommen wurden, durchgeführt. Die Integration des Verfahrens im Autonomous Driving (AD) Stack der Shuttles und die Evaluation in den beiden Zielgebieten ist nach der Erteilung der Level 4 Zulassung durch das Kraftfahrt Bundesamt (KBA) geplant.

Stichwörter

Erkennung von Lichtsignalanlagen, Ampelerkennung, Autonomes Fahren, Künstliche Intelligenz, Trainingsdatensätze für Verkehrssignalanlagen.

1 Einleitung

Die Erkennung und Zuordnung von Lichtsignalanlagen (Ampeln) ist ein wichtiges Forschungsthema im Bereich des autonomen und vernetzen Fahrens, das maßgeblich zur Sicherheit und Effizienz des Straßenverkehrs beiträgt. Dabei umfasst die Aufgabe nicht nur die visuelle Identifikation von Lichtsignalanlagen und die Ermittlung des Zustands, sondern auch die Zuordnung zu den relevanten Fahrspuren, um fundierte Fahrentscheidungen treffen zu können.

Künstliche Intelligenz (KI), insbesondere Methoden des maschinellen Lernens und Deep Learning, hat sich in den letzten Jahren als zentraler Ansatz für diese Aufgabe etabliert. Die am häufigsten eingesetzten KI-Methoden basieren auf künstlichen neuronalen Netzwerken, die durch große Mengen annotierter Daten trainiert werden. Convolutional Neural Networks (CNNs) wie YOLO (You Only Look Once) [1], SSD (Single Shot Multibox Detector) [2] und Faster R-CNN [3] sind Standardmodelle zur Erkennung von Objekten in Echtzeit, und werden unter anderem auch zur Erkennung von Lichtsignalanlagen eingesetzt.

[1] Affiliation M. Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache, Hochschule Heilbronn
[2] Affiliation M. Kocsis, FZI Forschungszentrum Informatik Karlsruhe

Die Forschung, die zu diesen Ergebnissen geführt hat, wird vom Bundesministerium für Wirtschaft und Klimaschutz im Rahmen des Projekts „Shuttle2X“ (Förderkennzeichen 19S22001B) gefördert.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Die folgende Arbeit betrachtet erstens mehrere existierende Datensätze und erweitert diese mit eigenen Daten welche mittels autonomer Shuttles aufgenommen wurden. Die Aufnahmen erfolgen auf zwei Strecken in Heilbronn und Bad Wimpfen wo autonome Shuttles auf täglicher Basis Personen in urbanen Verkehr transportieren. Aktuell sind diese Shuttles auf Level 2 zugelassen und das Bremsen bei den Anlagen wird von einem Sicherheitsfahrer veranlasst. Durch die Erweiterung der Datensätze mit eigenen gelabelten Daten wird sichergestellt, dass genau die Lichtsignalanlagen, welche im Realbetrieb in Betracht genommen werden, in den Trainingsdaten vorhanden sind. Zweitens werden unterschiedliche Ansätzen von Modellarchitekturen anhand existierender Referenzen analysiert. Es wird ein mehrstufiger Ansatz bestehend aus Detektion und Klassifikation der Lichtsignalanlagenzustände ausgewählt. Lichtsignalanlagen werden ausgemessen und in einer toplogischen Karte eingetragen, die konsistent mit der digitalen Karte des Shuttles ist. Somit ist zur Laufzeit eine Zuordnung der erkannten Lichtsignalanlagen möglich. Dabei wird in dem Zuordnungsprozess die aktuelle Position und Orientierung des Shuttles verwendet.

Die Arbeit ist wie folgt strukturiert: Kapitel 2 beschreibt den Stand der Technik bezüglich Datensätzen und Detektion und Klassifikation von Lichtsignalanlagen insbesondere Methoden welche auf KI-basierte Verfahren beruhen. In Kapitel 3 wird die experimentelle Vorgehensweise dargestellt welche die verwendeten Datensätze, die Modellarchitektur, das Trainingsverfahren und die Zuordnung Methodik beschreibt. In Kapitel 4 wird das Verfahren validiert und Kapitel 5 beinhaltet die Schlussfolgerung und beschreibt zukünftig geplante Arbeiten.

2 Stand der Technik

Für die Erkennung von Lichtsignalanlagen werden immer mehr Methoden der Künstlichen Intelligenz angewandt, da diese gegenüber der klassische Bildverarbeitungsmethoden [4] deutlich bessere Ergebnisse liefern. Dabei spielen repräsentative Datensätze sowie Modelarchitekturen eine wichtige Rolle. In [5] wird eine Analyse dieser zwei Aspekte durchgeführt.

2.1 Datensätze

Aus der durchgeführten Analyse hebt sich das „Drive UTraffic Light Dataset“ (DTLD) [6] durch die große Anzahlt der annotierten Bilder und die Klassenanzahl hervor. Außerdem wurden die Daten in Deutschland aufgenommen und sind somit auch für die in der Arbeit betrachtenden Testsites geeignet. Der Datensatz Bosch Small Traffic Lights (BSTLD) [7] hebt sich durch Bilder hervor, welche viele potenziell irrenführende Bildbereiche enthalten.

2.1.1 DriveU Traffic Light Dataset

Der DTLD-Datensatz umfasst mehr als 230.000 annotierte Lichtsignalanlagen in Kamerabildern mit einer Auflösung von 2048×1024 Pixeln, aufgenommen in 11 deutschen Städten bei einer Frequenz von 15 Hz. Die Annotationen umfassen eine Vielzahl von Attributen wie Orientierung, Relevanz, Zustand und Piktogramm, wodurch insgesamt 344 verschiedene Klassen definiert werden. Zusätzlich zu den Kamerabildern werden Disparitätsbilder bereitgestellt, die tiefenbasierte Analysen und Detektionen ermöglichen. Ein wesentlicher Vorteil des DTLD-Datensatzes liegt in der Möglichkeit, Objekte mit einer Breite von 5 Pixeln oder weniger zu annotieren, wodurch zahlreiche sehr kleine Objekte berücksichtigt werden können. Dies macht den Datensatz besonders wertvoll für die Untersuchung von Objekterkennungsverfahren bei niedrigen Auflösungen. Eine weitere Besonderheit ist die hohe Varianz im Seitenverhältnis der Lichtsignalanlagen, da diese mit einer, zwei, drei oder vier aktiven Lichtsignalen auftreten. Die räumliche Vielfalt der Annotationen wird durch die

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Berücksichtigung von Szenen erreicht, in denen sich mindestens eine Lichtsignalanlage im Zustand geändert hat, was zu einer breiten Streuung der Lichtsignalanlagepositionen im Bild führt. Die Annotationen enthalten detaillierte Informationen, darunter die Bounding-Box-Koordinaten sowie Attribute wie perspektivische Orientierung, Relevanz, Anzahl der Lichtsignale, Zustand und das Piktogramm. In der Version 1.0 des Datensatzes werden diese Attribute durch eine sechsstellige Klassenidentitätsnummer kodiert. Ab Version 2.0 wurde diese Nummer durch ein Attribut-Dictionary ersetzt, das eine flexiblere Darstellung der Eigenschaften ermöglicht. Ein zusätzliches Attribut, "Reflektion", wurde eingeführt, um anzugeben, ob es sich bei einer detektierten Lichtsignalanlage um eine Spiegelung handelt. Die Klassenidentitätsnummer setzt sich wie folgt zusammen: Die erste Stelle beschreibt die perspektivische Orientierung, die zweite die Relevanz, die dritte die Orientierung, die vierte die Anzahl der Lichtsignale, die fünfte den Zustand und die sechste das Piktogramm des aktiven Lichtes. Für jede Lichtsignalanlage wird zudem eine Tracking-Identitätsnummer vergeben, die über Sequenzen hinweg erhalten bleibt, um Tracking-Analysen zu erleichtern. Zusätzlich zu den Bilddaten stellt der Datensatz Zeitstempel sowie Fahrzeugdaten wie GPS-Position, Geschwindigkeit und Gierrate bereit. Diese Zusatzinformationen ermöglichen eine detaillierte zeitliche und räumliche Analyse der Szenen und verbessern die Möglichkeiten zur Validierung von Algorithmen für die Detektion und Verfolgung von Lichtsignalanlagen.

2.1.2 Bosch Small Traffic Lights Dataset

Der Bosch Small Traffic Lights Dataset ist ein spezialisierter Datensatz zur Erkennung von Lichtsignalanlagen, der primär für die Entwicklung von Computer-Vision-Modellen für automatisiertes Fahren in urbanen Umgebungen konzipiert wurde. Er dient insbesondere der Validierung von Objekterkennungsverfahren mit Fokus auf kleine Objekte in hochauflösenden Bildern. Die im Datensatz enthaltenen Szenen repräsentieren eine Vielzahl von Verkehrssituationen und typischen Herausforderungen im urbanen Kontext. Dazu zählen stark frequentierte Straßen in Stadtzentren, mehrspurige Straßen in Vororten mit variierenden Verkehrsdichten, dichter Verkehr, Baustellen sowie erhebliche Änderungen der Lichtverhältnisse. Zusätzlich enthält der Datensatz potenziell irreführende Bildbereiche, beispielsweise große runde Rücklichter, die mit Lichtsignalanlagen verwechselt werden könnten. Der Datensatz umfasst insgesamt 13.427 Kameraaufnahmen mit einer Auflösung von 1280 × 720 Pixeln und rund 24.000 annotierten Lichtsignalanlagen. Diese Annotationen beinhalten sowohl die Umrisse der Lichtsignalanlagen als auch deren aktuellen Lichtzustand. Die Aufnahmen liegen sowohl als 12-Bit-HDR-Bilder als auch als rekonstruierte 8-Bit-RGB-Farbbilder vor. Die RGB-Bilder wurden ursprünglich für Debugging-Zwecke erstellt, können jedoch auch für Trainingszwecke eingesetzt werden, obwohl sie gelegentlich Artefakte aufweisen und Farbabweichungen enthalten können. Der Datensatz ist in einen Trainings- und einen Testdatensatz unterteilt. Der Trainingssatz besteht aus 5093 Bildern, die in etwa alle zwei Sekunden annotiert wurden, und umfasst 10.756 Lichtsignalanlagen mit einer durchschnittlichen Breite von etwa 8,6 Pixeln. Der Testsatz umfasst 8334 aufeinanderfolgende Bilder, die mit einer Frequenz von etwa 15 Bildern pro Sekunde annotiert wurden, und enthält 13.486 Lichtsignalanlagen mit einer mittleren Breite von etwa 8,5 Pixeln. Im Testsatz wurden alle Bilder annotiert, wobei zeitliche Informationen genutzt wurden, um die Genauigkeit der Annotationen zu verbessern. Beide Datensätze wurden unabhängig voneinander, jedoch innerhalb derselben Region, aufgenommen.

2.2 Modellarchitekturen zur Detektion von Lichtsignalanlagen

Beim Vergleich von Modellarchitekturen zur Detektion von Lichtsignalanlagen lassen sich drei Hauptansätze unterscheiden: der Einsatz modifizierter generischer Objektdetektoren, mehrstufige Ansätze (generischer Objetkdetektor oder Non-Deep Learning Detektor + (CNN)-Klassifikator) und aufgabenspezifische einstufige Ansätze.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Die erste Gruppe umfasst Ansätze, die ein vorhandenes CNN-basiertes Modell für die generische Objekterkennung verwenden, wobei geringfügige Anpassungen vorgenommen werden, um kleinere Objektgrößen zu berücksichtigen (z.B. angepasster Yolo oder Faster R-CNN). Generische Objektdetektoren sind besonders vorteilhaft aufgrund ihrer hohen Inferenzgeschwindigkeit.

Der mehrstufige Ansatz kombiniert mehrere Modelle, die aufeinander aufbauend arbeiten. Zunächst wird ein Modell zur Erkennung von Regions of Interest (ROI) eingesetzt. Dieses Modell markiert potenzielle Bereiche der Lichtsignalanlage im Bild mithilfe von Bounding Boxes. Die extrahierten Bildausschnitte werden anschließend von einem oder mehreren nachgelagerten Modellen verarbeitet, die die Klassifikation übernehmen. Diese Modelle bestimmen entweder den aktuellen Zustand der Lichtsignalanlage (z. B. rot, gelb, grün) oder analysieren das auf der Leuchte angezeigte Piktogramm.

Die dritte Gruppe umfasst Ansätze, bei denen die Lichtsignalanlageerkennung innerhalb eines einzigen Netzwerks erfolgt, das speziell für diese Aufgabe entwickelt wurde. Ein Beispiel ist DeepTLR [8] das mit einem reinen CNN Probability Maps für die Zustände (rot, gelb, grün) erstellt und basierend darauf Bounding Box-Vorhersagen trifft. HDTLR [9] baut darauf auf und verbessert die Erkennung, wobei verschiedene CNN-Architekturen wie AlexNet, GoogLeNet und VGG genutzt werden können. Die Aufgabe wird hier end-to-end gelernt ohne dass eine explizite Trennung zwischen Detektions- und Klassifikationsschritte existiert.

2.3 Schlussfolgerung

Die Datensätze DriveU Traffic Light Dataset und Bosch Small Traffic Lights Dataset bieten eine große Menge und Variation an Bilddaten welche zugeschnitten auf die in der Arbeit beschriebenen Lichtsignalanlagenerkennung. Dennoch enthalten diese Daten die Testsites Heilbronn und Bad Wimpfen nicht. Daher ist eine Erweiterung dieser Datensätze durch Annotation von selbst durchgeführten Aufnahmen des vorhandenen Shuttles empfehlenswert.

Als Architektur wird ein mehrstufiges Verfahren ausgewählt, da diese bei der Erkennung kleiner oder schwer identifizierbarer Objekte Vorteile bieten und eine gezielte Anpassung an spezifische Anwendungsfälle erleichtern.

3 Experimentelle Vorgehensweise

Zum Trainieren der Modelle wurden die in Kapitel 2.1 beschriebenen Datensätze herangezogen und ein mehrstufiger Ansatz, welcher auf zwei YOLO Modellen für die Detektion und Klassifikation der Zustände der Lichtsignalanlegen basiert, ausgewählt.

3.1 Datensätze

Damit die Datensätze in der YOLO Umgebung genutzt werden können, müssen sie in das spezifische YOLO-Format konvertiert werden. Dies betrifft insbesondere die Datensätze BSTLD und DTLD, die vor der Verarbeitung entsprechend angepasst werden müssen. Der DTLD Datensatz enthält darüber hinaus Annotationen für Lichtsignalanlagen für Fußgänger, Buse und horizontal ausgerichtete Anlagen. Da diese Klassen für die vorliegende Arbeit nicht relevant sind, wurden sie durch eine Filterung anhand der Klassenidentitätsnummer entfernt. Der Datensatz ist in zwei Unterordner, *train* und *val*, unterteilt, die jeweils die Unterordner *images* und *labels* enthalten. Der Ordner *train* umfasst etwa 80 % des Datensatzes und dient dem Modelltraining, während die verbleibenden 20 % im Ordner *val* zur Validierung des Trainings verwendet werden. Die Position der Bounding Boxes im Bild wird ebenfalls in einem spezifischen Format gespeichert. Das von YOLO verwendete Format beschreibt

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

jede erkannte Lichtsignalanlage mit einer Zeile, die fünf normierte Angaben enthält: die Klassen-ID, die Koordinaten des Mittelpunkts der Bounding Box in x- und y-Richtung sowie die normierte Breite und Höhe der Bounding Box. Diese Angaben sind relativ zur Breite und Höhe des jeweiligen Bildes normiert.

Besonderes Augenmerk liegt auf der Auflösung der Bilder. Während die Originalbilder des DriveU-Datensatzes eine Auflösung von 2048x1024 Pixeln haben, wird für die Erkennung von Lichtsignalanlagen im autonomen Shuttle eine reduzierte Auflösung von 640x480 Pixeln verwendet. Daher wurden die Bilder der Datensätze vor dem Training auf diese Zielauflösung skaliert. Unter Berücksichtigung, dass die Breite markierter Anlagen in einigen Fällen bis zu fünf Pixel oder weniger beträgt, war es erforderlich, die Bounding-Box-Koordinaten entsprechend an die neue Auflösung anzupassen, um eine konsistente Darstellung der Annotationen zu gewährleisten.

Die für das Training der Klassifikationsmodelle verwendeten Datensätze weisen eine andere Struktur auf als jene, die für Detektionsmodelle genutzt werden. Die Bilder sind klassenabhängig in separaten Unterordnern organisiert. Die Grundlage für die Erstellung dieser Datensätze bilden die in den Detektionsdatensätzen annotierten Lichtsignalanlagen. Hierbei wurden die markierten Bildbereiche mithilfe der Koordinaten der Bounding Boxes ausgeschnitten und entsprechend ihrer Klasse sortiert abgespeichert. Für die Verarbeitung durch das Klassifikationsmodell ist eine Mindestbildgröße von 30x30 Pixeln erforderlich, weshalb kleinere Bildausschnitte entsprechend skaliert oder verworfen wurden. Zusätzlich wurden Bilder für Klassen, die keine Lichtsignalanlage repräsentieren, manuell extrahiert. Diese manuellen Annotationen dienen dazu, eine ausgewogene Verteilung der Klassen sicherzustellen und mögliche Fehlklassifikationen im Modell zu minimieren.

Die neu erzeugten Datensätze wurden mit Aufnahmen vom Shuttle heraus erweitert. Dabei wurde eine Logitech Kamera benutzt welche auf einer Höhe von 2.13 Meter mittig vom Shuttle angebracht wurde. Es wurden von zehn Fahrten in Heilbronn und Bad Wimpfen mit unterschiedlichen Wetterbedingungen wie leichter Regen, sonniger Tag oder Spätnachmittag zur bürgerlichen Dämmerung (ca. 45 Minuten vor Sonnenuntergang), Bilddaten aufgenommen. Die Aufnahme erfolgte in Robotic Operating System (ROS) [10] spezifischen .bag Dateien. Die Bilder wurden aus der .bag Datei extrahiert und dann mit Hilfe von Roboflow [11] annotiert und augmentiert. Im Bild 1 wird der Annotationsprozess, das resultierende Bild und die für die Augmentation verwendeten Optionen dargestellt.

Bild 1: Annotation der aufgenommenen Bilder mittels Roboflow, annotiertes Bild Ergebnis, Optionen für die Augmentation.

3.2 Modellarchitektur

Das ausgewählte Verfahren besteht aus zwei Modellen, basierend auf YOLO, die in einer zweistufigen Pipeline kombiniert werden (Bild 2).

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

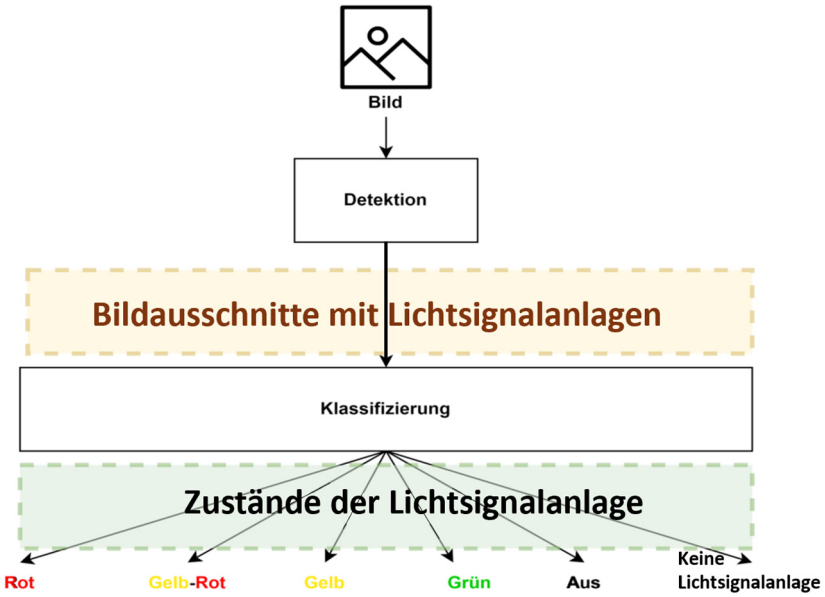

Bild 2: Modellarchitektur Lichtsignalanlagenerkennung.

In der ersten Stufe wird ein auf YOLOv8n basierendes Modell für die Detektion eingesetzt. Dieses Modell verarbeitet die Bilder der im Shuttle verbauten Kamera und identifiziert Bildausschnitte, die potenziell Lichtsignalanlagen enthalten. Dabei erfolgt die Detektion einer einzigen Klasse: *Lichtsignalanlagen*. Die Ausgabe des Modells besteht aus den Bildregionen, die als Eingabe für die zweite Verarbeitungsstufe dienen.

Die zweite Stufe übernimmt die Klassifizierung der erkannten Bildausschnitte. Hier kommt ein Modell zum Einsatz, das auf YOLOv8n-cls basiert. Es klassifiziert die erkannten Objekte in eine von neun Klassen: *Rot*, *Gelb-Rot*, *Gelb*, Grün, Aus, Auto, Natur, Fenster und Verkehrsschild. Die Klassen Aus, Auto, Natur, Fenster und Verkehrsschild werden bei der Auswertung zur Klasse *keine Lichtsignalanlage* zusammengefasst. Diese Klassen wurden anhand von erfahrungsgemäß falsch detektierten Objekten gewählt. Dadurch werden fehlerhaft detektierte Lischtsignalanlagen aus der ersten Stufe erkannt und entsprechend gefiltert. Dieser Ansatz erhöht die Robustheit des Systems, indem er eine zusätzliche Sicherheitsebene gegen falsch erkannte Lichtsignalanlagen schafft und die Gesamtzuverlässigkeit der Pipeline verbessert.

3.3 Training der KI-Modelle

In diesem Abschnitt werden das Trainingssetup, die Evaluationsmetriken und die Ergebnisse des Modelltrainings beschrieben. Das Training gliedert sich in mehrere Schritte, bei denen verschiedene die Datensätze genutzt werden, um die Modelle für Detektion und Klassifikation zu optimieren. Zur Bewertung der Modelle während des Trainings und der Validierung wurden folgende Metriken verwendet:

- Box Loss (box_loss): Diese Metrik misst die Abweichung zwischen der vorhergesagten und der tatsächlichen Position sowie Größe der Bounding Boxes.
- Cls Loss (cls_loss) (Klassifikationsverlust): Bewertet die Abweichung zwischen den vorhergesagten und den tatsächlichen Klassen der Objekte.
- DFL Loss (dfl_loss) (Distributed Focal Loss): Verleiht schwer zu erkennenden Objekten eine höhere Gewichtung und misst, wie präzise diese erkannt werden.
- Precision: Der Anteil korrekt erkannter positiver Objekte (True Positives) an allen als positiv vorhergesagten Objekten. Eine hohe Precision deutet auf wenige False Positives hin.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

- Recall: Der Anteil korrekt erkannter positiver Objekte an allen tatsächlichen positiven Objekten. Eine hohe Recall-Rate zeigt, dass die meisten Objekte erkannt werden.
- mAP50: Der *mean Average Precision* bei einem Intersection-over-Union-(IoU)-Schwellenwert von 0,50. Diese Metrik gibt an, wie genau vorhergesagte Bounding Boxes mit den tatsächlichen Objekten übereinstimmen.
- mAP50-95: Durchschnittlicher mAP-Wert über IoU-Schwellenwerte von 0,50 bis 0,95 in Schritten von 0,05. Diese Metrik bietet ein umfassenderes Bild der Modellleistung.

3.3.1 Detektion

Das Grundmodell yolov8n.pt, bereitgestellt von Ultralytics, dient als Ausgangspunkt. Es wird über 100 Epochen mit dem BSTLD-Datensatz trainiert, wobei ein Teil eines selbst erstellten Datensatzes zur Validierung verwendet wird. Während des Trainings zeigen die Metriken box loss, cls loss und dfl loss eine kontinuierliche Verbesserung, was auf eine präzisere Vorhersage der Bounding Boxes, der Objektklassifikation und der Erkennung schwer identifizierbarer Objekte hinweist. Precision und Recall weisen jedoch starke Schwankungen auf, zeigen aber insgesamt eine leichte Verbesserung. Die Validierungsmetriken zeigen ähnliche Schwankungen, insbesondere bei box loss, cls loss und dfl loss, was darauf hindeutet, dass das Modell Schwierigkeiten hat, die Lichtsignalanlagen im Validierungsdatensatz zuverlässig zu erkennen. mAP50 zeigt eine leichte Verbesserung, während bei mAP50-95 keine signifikante Tendenz erkennbar ist (Bild 3).

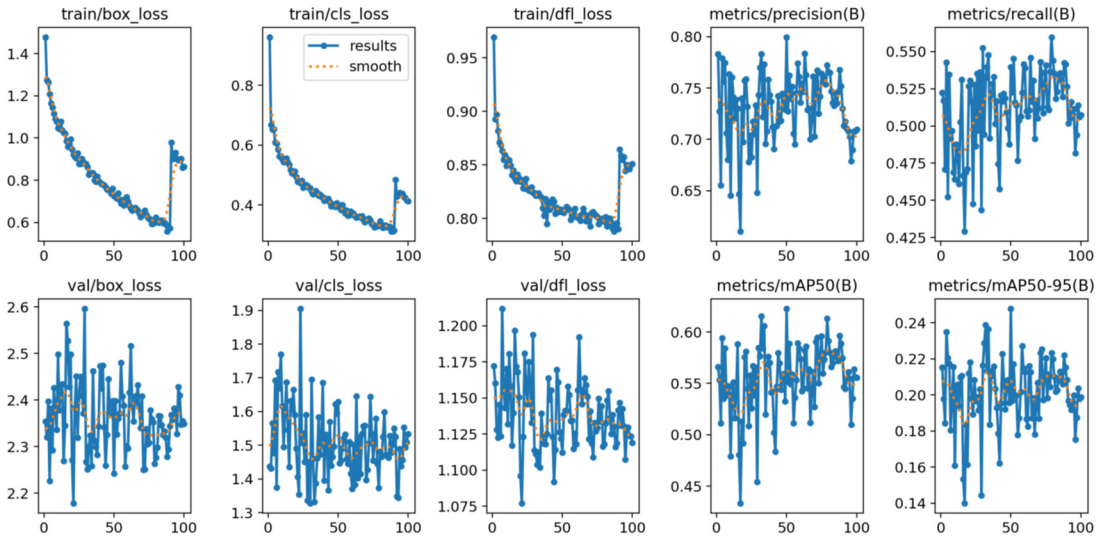

Bild 3: Trainingsmetriken – BSTDL Datensatz.

Im zweiten Schritt wird das Modell mit Daten des DTLD-Datensatzes aus Dortmund trainiert. Die Gewichte aus dem ersten Training dienen als Ausgangspunkt. Während des Trainings über 100 Epochen zeigen die Fehlerwerte (box loss, cls loss, dfl loss) erneut eine kontinuierliche Verbesserung. Die Validierungsmetriken zeigen deutliche Schwankungen und höhere Fehlerwerte als im ersten Schritt, insbesondere bei der Klassifikation. mAP50 und mAP50-95 erreichen tendenziell höhere Werte, zeigen jedoch keinen stetigen Anstieg über die Epochen hinweg (Bild 4).

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

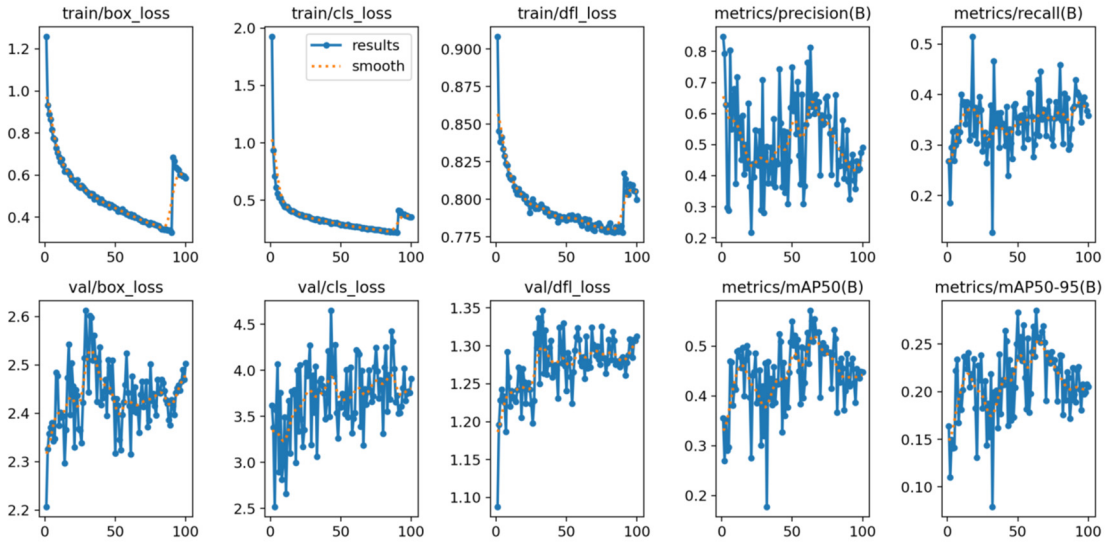

Bild 4: Trainingsmetriken – DTDL Datensatz.

Im finalen Schritt wird das Modell auf einem selbst erstellten Datensatz weitertrainiert. Hier zeigt das Training über 100 Epochen deutliche Verbesserungen bei Precision und Recall, die sich stabilisieren und auf einem höheren Niveau bleiben. Die Validierungsmetriken zeigen ebenfalls Verbesserungen. Box loss verläuft stabiler und erreicht niedrigere Endwerte. Cls loss und dfl loss weisen anfangs Schwankungen auf, stabilisieren sich jedoch im Verlauf des Trainings. mAP50 und mAP50-95 steigen kontinuierlich an und erreichen höhere Werte als in den vorherigen Trainingsschritten, zeigen jedoch weiterhin Schwankungen (Bild 5).

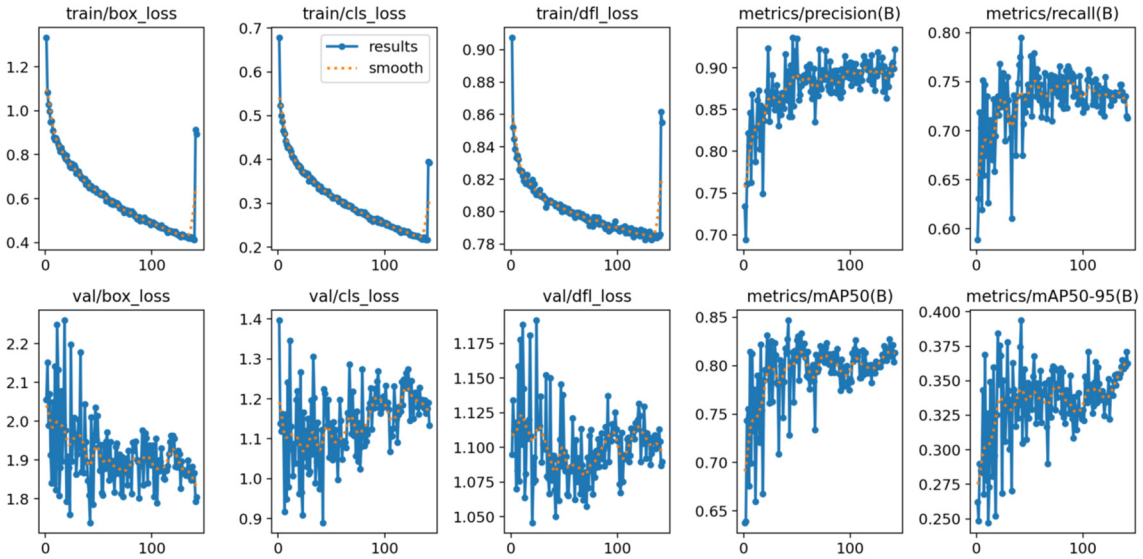

Bild 5: Trainingsmetriken – Selbsterstellter Datensatz.

In Bild 6 ist die Performance des Modells anhand von Beispielbildern abgebildet. Links werden vier Bilder des Validierungsdatensatzes mit den dazugehörigen markierten Lichtsignalanlagen. Rechts wird dargestellt, welche Vorhersagen das Modell bei den vier Bildern macht. Es ist zu erkennen, dass das Modell in den gezeigten Beispielen alle Lichtsignalanlagen erkennt. Neben den eingezeichneten Bounding Boxes sind auch die vom Modell berechnete Wahrscheinlichkeiten angegeben. Diese liegen zwischen 0.5 und 0.9, was für eine solide Erkennung spricht.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Bild 6: Validierung Labels und Detektion.

3.3.2 Klassifikation

Das Training der Klassifikationsmodelle basiert auf dem vortrainierten Modell yolov8n-cls.pt. Das Modell wird auf neun Klassen trainiert, darunter die verschiedenen Zustände der Lichtsignalanlagen und Klassen für typische Fehlklassifikationen (*keine Lichtsignalanlage*). Die Trainingsdaten werden aus den BSTLD- und DTLD-Datensätzen extrahiert, indem Bounding Boxes ausgeschnitten und entsprechend ihrer Klassen gespeichert werden. Der Trainingsprozess über 100 Epochen zeigt stetige Verbesserungen bei der accuracy top1, die die korrekte Klassifizierung angibt, sowie bei der accuracy top5, die eine nahezu perfekte Zuverlässigkeit aufweist.

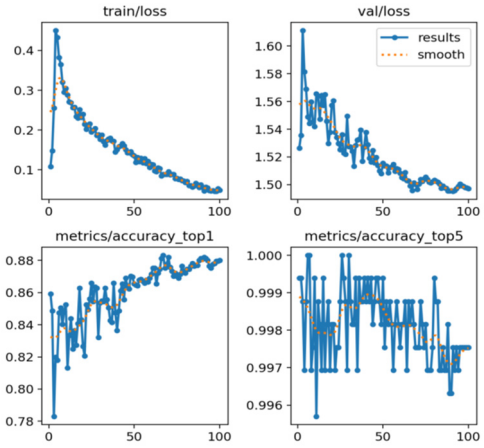

Bild 7: Trainingsmetriken – Klassifizierung.

3.3.3 Interpretation

Trotz Verbesserungen in den Metriken zeigen die Trainings- und Validierungsergebnisse einige Schwächen. Potenzielle Ursachen umfassen: Ungleichgewicht der Daten: Ein unausgewogenes Verhältnis zwischen Bildern mit und ohne Lichtsignalanlagen kann zu Verzerrungen führen; Fehlerhafte Annotationen: Unvollständige oder fehlerhafte Markierungen beeinträchtigen die Genauigkeit; Redundante Bildinhalte: Wiederholte Muster (z. B. Fahrzeugdetails) können das Modell in die Irre führen; Ungleich verteilte Klassen: Häufige Zustände wie *Rot* oder *Grün* können zu einer Überrepräsentation führen; Zu starke Bildaugmentation: Übermäßige Änderungen an den Trainingsbildern können die Generalisierungsfähigkeit des Modells beeinträchtigen.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Die Ergebnisse zeigen, dass eine sorgfältige Datensatzaufbereitung sowie eine konsistente und ausgewogene Datenverteilung entscheidend für den Trainingserfolg sind. Zudem beeinflussen die Wahl der Trainingsparameter und die Qualität der Annotationen die Modellleistung erheblich.

3.4 Zuordnung der Lichtsignalanlagen

Nach der Erkennung einer Lichtsignalanlage muss geprüft werden, ob diese für die aktuelle Fahrsituation des Shuttles relevant ist. Da das Shuttle stets dieselbe Route befährt, sind die Positionen der Anlagen im globalen Koordinatensystem bekannt. Ziel der Validierung ist es, die erkannte Lichtsignalanlage einer vordefinierten Position entlang der Strecke zuzuordnen. Diese Zuordnung erfolgt auf Basis des Winkels zwischen der Ausrichtung des Shuttles und der Position der Lichtsignalanlage. Die Position und Orientierung des Shuttles werden von Lokalisierungsmodul auf dem Shuttle berechnet und sind zu jedem Zeitpunkt bekannt. Basierend darauf wird der Winkel zwischen der Fahrtrichtung des Shuttles und der Position der Anlage berechnet. Aus der Ausrichtung des Shuttles im globalen Koordinatensystem φ_{0S} wird der Vektor $\vec{v}$ abgeleitet und es wird der Winkel zwischen $\vec{v}$ und $\overrightarrow{SA}$ berechnet. Lichtsignalanlagen werden nur dann als relevant betrachtet, wenn dieser Referenzwinkel kleiner als 90° ist, da nur solche Analgen innerhalb des Sichtfelds der Kamera liegen können. Zudem wird eine Lichtsignalanlage nur bis zu einem bestimmten maximalen Abstand als relevant eingestuft, basierend auf dem Betrag $|\overrightarrow{SA}|$. Im nächsten Schritt wird der Winkel zwischen der aktuellen Shuttleausrichtung und der erkannten Anlage auf Basis der Kameraposition berechnet (siehe Bild 8). Die Mitte des Kamerabildes entspricht der Fahrtrichtung des Shuttles, und die Breite des Bildes (W=640 Pixel) wird auf den Aufnahmewinkel der Kamera $\varphi_K$skaliert. Unter der Annahme eines linearen Zusammenhangs zwischen dem Winkel und der x-Koordinate der Lichtsignalanlage im Bild ergibt sich der Winkel $\varphi_A$wie folgt: $\varphi_A = \frac{x}{W} \cdot \varphi_K$.

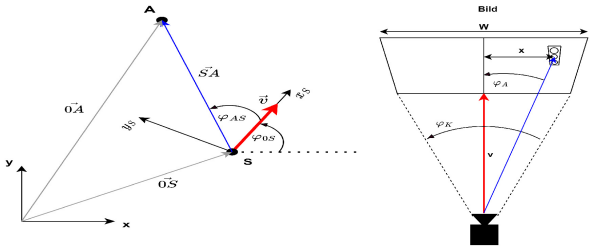

Bild 8: Referenzwinkelbestimmung anhand von Lichtsignalanlage- und Shuttleposition; Winkelbestimmung anhand des Kamerabildes.

Hierbei ist x der Abstand der erkannten Anlageposition zum Bildmittelpunkt in Pixeln. Die Differenz zwischen dem Referenzwinkel und dem Kamerawinkel wird berechnet. Liegt der Betrag dieser Differenz innerhalb einer definierten Toleranz, wird die Lichtsignalanlage als gültig und ihrer vordefinierten Position zugeordnet. Tests haben gezeigt, dass eine Toleranz von 15° einen guten Kompromiss darstellt. Diese Einstellung minimiert sowohl Fehlklassifikationen als auch das Risiko, relevante Anlagen zu übersehen, und gewährleistet eine zuverlässige Zuordnung der erkannten Anlagen zu den vordefinierten Positionen.

4 Validierung des Verfahrens

Zur Validierung und Analyse des Modells wurden mehrere Debug- und Auswertehilfen implementiert, die eine detaillierte Untersuchung des Modellverhaltens ermöglichen. Während der Ausführung wird das aktuelle Kamerabild angezeigt, wobei erkannte Lichtsignalanlagen durch blaue

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

Bounding Boxes hervorgehoben werden. Innerhalb jeder Bounding Box werden der erkannte Zustand der Lichtsignalanlage, der Winkel zur Lichtsignalanlage sowie die x-Koordinate eingeblendet. Zusätzlich werden zwei Fahrmanöver visualisiert, die oben links im Bild dargestellt sind: *Aktuelles Manöver*: Dieses basiert auf dem aktuell erkannten Bild und zeigt die Fahrentscheidung, die aus der momentanen Erkennung der Lichtsignalanlage resultiert.; *Stabiles Manöver*: Dieses berücksichtigt die Ergebnisse der letzten Bilder, um eine stabilere Fahraktion zu berechnen, die weniger anfällig für kurzfristige Erkennungsfehler ist. Die genannten Visualisierungen und Ausgaben unterstützen eine detaillierte Überprüfung der Modellleistung und tragen zur Analyse der Genauigkeit sowie Stabilität der Lichtsignalanlageerkennung bei. Die Evaluierung der Erkennung wurde mithilfe realer Daten durchgeführt, die mit den Shuttles erhoben wurden.

Die Evaluierung zeigt, dass das entwickelte Modell in der Lage ist, alle Lichtsignalanlagen auf den beiden getesteten Strecken zuverlässig zu erkennen und korrekt zuzuweisen. Jedoch wurden qualitative Unterschiede zwischen den Erkennungen einzelner Lichtsignalanlagen auf den unterschiedlichen Strecken beobachtet. Diese Diskrepanzen lassen sich vermutlich auf die Eigenschaften der verwendeten Trainingsdatensätze zurückführen, insbesondere auf unterschiedliche Beleuchtungsverhältnisse, Betrachtungswinkel oder andere variierende Merkmale in den Trainingsbildern. Erste Erkennung von Lichtsignalanlagen sind bereits aus einer Distanz von etwa 100 Metern möglich. Die Zuverlässigkeit der Erkennung und Zuordnung verbessert sich jedoch deutlich mit abnehmender Distanz. Ab einer Entfernung von etwa 50 Metern arbeitet das Modell konsistent und liefert zuverlässige Ergebnisse. Vereinzelt treten noch Fehlklassifikationen auf. Diese beschränken sich jedoch hauptsächlich auf einzelne Bilder und zeigen kein systematisches Muster. Häufig entstehen solche Fehlerkennungen durch Bildstrukturen, die den Konturen einer Lichtsignalanlagen stark ähneln, wie beispielsweise Fenster oder Fassaden. Diese Fehlklassifikationen haben jedoch nur einen begrenzten Einfluss auf die Gesamtleistung des Modells und könnten durch gezielte Erweiterung der Trainingsdaten weiter minimiert werden. Außerdem werden Bildsequenzen ausgewertet, sodass die einzelnen Ausreißer erkannt und eliminiert werden. Die Zuordnung der detektierten Lichtsignalanlagen zu den im System hinterlegten Referenzlichtsignalanlagen entlang der Strecke funktioniert zuverlässig. Bild 9 stellt Beispielhaft der Erkennung der Lichtsignalanlagen in den zwei Testsites.

Bild 9: Zuverlässige Erkennung von Lichtsignalanlagen in den Testsites Bad Wimpfen (obere Reihe) und Heilbronn (untere Reihe). Das mittlere Bild in der unteren Reihe zeigt eine vereinzelte Fehldetektion. Solche Bilder sind vereinzelt und beeinflussen nicht die Systemperformance.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

5 Schlussfolgerung und weitere Schritte

Das Modell zeigt eine insgesamt zuverlässige und robuste Leistung, wobei alle Lichtsignalanlagen auf den getesteten Strecken erfolgreich erkannt und korrekt zugeordnet wurden. Dennoch wurden Unterschiede in der Erkennungsqualität zwischen den Strecken festgestellt, was auf ein Optimierungspotenzial bei den Trainingsdatensätzen hinweist. Eine bessere Balance und Diversifizierung der Datengrundlage kann dazu beitragen, die Leistung des Modells in verschiedenen Szenarien zu harmonisieren. Die Erkennung funktioniert zuverlässig bis zu einer Distanz von etwa 100 Metern, wobei bis zu 50 Metern eine hohe Genauigkeit und Konsistenz erreicht wird. Schwächen zeigt das Modell insbesondere bei der Unterscheidung eng beieinanderliegender Lichtsignalanlagen und vereinzelt bei Fehlklassifikationen in einzelnen Bildern. Trotz dieser Einschränkungen bietet das Modell eine solide Basis für eine verlässliche Erkennung der Anlagen und rechtfertigt die Weiterentwicklung dieses Ansatzes.

In den nächsten Schritt soll das Verfahren in den Zielgebieten eingesetzt und evaluiert werden. Im Projekt Shuttle2X [12] wird zusätzlich eine Kommunikation mit der Infrastruktur in Testsite Bad Wimpfen realisiert. Hier wird Infrastrukturperzeption basierend auf einer Fusion zwischen Kamera und Lidar Verfahren [13], [14], [15] in der shuttleseitigen Perzeption [16] integriert. Da die Lichtsignalanlagen in Bad Wimpfen keine technische Möglichkeit für die Übermittlung ihres Zustands haben, ist das hier vorgestellte Verfahren relevant. Eine zuverlässige Lichtsignalanlageerkennung ist ein entscheidender Faktor für die Sicherheit und Effizienz autonomer Fahrzeuge und trägt wesentlich zur gesellschaftlichen Akzeptanz dieser Technologie bei, da sie Vertrauen in deren Fähigkeit schafft, komplexe Verkehrssituationen sicher zu bewältigen, während sie gleichzeitig die Grundlage für innovative Mobilitätskonzepte wie Service-on-Demand-Anwendungen [17] schafft. Die Erweiterung auf nutzerfreundlichen on-Demand Dienste die individuelle und flexible Transportlösungen ermöglichen, ähnlich wie in [18] und [19] für autonome Paketlieferungen, wird in der Zukunft in beiden Testsites angestrebt.

6 Literatur

[1] J. Redmon, A. Farhadi, “Yolov3: An incremental improvement”, CoRR, vol. abs/1804.02767, 2018.
[2] W. Liu et. all “SSD: Single Shot MultiBox Detector”, European Conference on Computer Vision, 2016
[3] M. Islam, N. Bryant, “Faster R-CNN based Traffic Sign Detection and Classification” In WSEAS TRANSACTIONS ON SIGNAL PROCESSING. 18. 1-10. 10.37394/232014.2022.18.1.
[4] M. Diaz et. all “A Survey on Traffic Light Detection”, In New Trends in Image Analysis and Processing - ICIAP Workshops 2015.
[5] S. Pavlitska, et. all, “Traffic Light Recognition using Convolutional Neural Networks: A Survey.”, 10.48550/arXiv.2309.02158.
[6] A. Fregin, J. Muller, U. Krebel, K. Dietmayer, “The driveu traffic light dataset: Introduction and comparison with existing datasets” In IEEE International Conference on Robotics and Automation (ICRA), pages 3376‑3383, 2018.
[7] K. Behrendt, L. Novak, R. Botros, "A deep learning approach to traffic lights: Detection, tracking, and classification," In IEEE International Conference on Robotics and Automation (ICRA), Singapore, pp. 1370-1377, doi: 10.1109/ICRA.2017.7989163, 2017.
[8] M. Weber, P. Wolf, J. M. Zöllner, “Deeptlr: A single deep convolutional network for detection and classification of traffic lights”, In IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 2016, pp. 342-348, doi: 10.1109/IVS.2016.7535408.
[9] M. Weber, M. Huber, J. M. Zöllner, “ HDTLR: A CNN based Hierarchical Detector for Traffic Lights“, *21st International Conference on Intelligent Transportation Systems (ITSC)*, USA, pp. 255-260, 2018.
[10] Q. Morgan et. all “ROS: an open-source Robot Operating System” In ICRA Workshop on Open Source Software. 3, 2009.
[11] S. Alexandrova, Z. Tatlock, M. Cakmak, Maya, “RoboFlow: A flow-based visual programming language for mobile manipulation tasks” Proceedings - IEEE International Conference on Robotics and Automation. 2015. 5537-5544. 10.1109/ICRA.2015.7139973.
[12] M. Yazgan et. all “Shuttle2X-Overcoming Operational Borders of Autonomous Shuttles by Infrastructure Support", In IEEE Intelligent Vehicles Symposium (IV), pp. 390-396, 2024.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +

[13] J. Buyer, M. Vollert, M. Kocsis, N. Sußmann, R. Zoellner, "Image-based multi-target tracking using a multi-layer particle filter and extended em clustering", In IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 620-625, 2017.

[14] J. Buyer, M. Vollert, A. Haas, M. Kocsis, R. Zöllner, "An adaptive multi-layer particle filter for tracking of traffic participants in a roundabout", In IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 2625-2631, 2016.

[15] J. Buyer, M. Vollert, M. Kocsis, N. Sußmann and R.D. Zöllner, "Multi-object tracking based on a multi-layer particle filter for unclustered spatially extended measurements" Lecture Notes in Electrical Engineering, Cham:Springer, vol. 501, pp. 219-238, 2018.

[16] M. Sautter, M. Kocsis, J. Buyer, R.D. Zöllner, " Realtime Lidar-Based Detection and Tracking of Dynamic Objects for an Autonomous Vehicle on Public Roads", In International Congress of Automotive and Transport Engineering, pp. 299-312, 2024.

[17] M. Kocsis, J. Winckler, N. Sußmann, R. Zöllner, " Interactive mission planning system of an autonomous vehicle fleet that executes services", In 23rd International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6, 2020.

[18] N. Marsden et. all "Research for Responsible Innovation: A Living-Lab Approach for Last-Mile Logistics Using a Self-Developed Autonomous Transport System", In International Conference on Human-Computer Interaction (HCI), pp. 316-333, 2023.

[19] M. Kocsis, R. Zöllner, G. Mogan, „Interactive system for package delivery in pedestrian areas using a self-developed fleet of autonomous vehicles", Electronics 2022, 11, 748. https://doi.org/10.3390/electronics11050748.

Open Access. © 2025 M.Kocsis, B. Hartmann, E. Heinrich, R. Zöllner, N. C. Stache
This work is licensed under the Creative Commons Attribution 4.0 License. +