

# Selbstlokalisierung für automatisierte Fahrzeuge mit tiefen neuronalen Netzen

Nico Engel und Klaus Dietmayer\*

**Zusammenfassung:** Für das automatisierte Fahren ist die Kenntnis über die eigene Fahrzeugpose (bestehend aus Position und Orientierung) unerlässlich, um ein hohes Maß an Sicherheit und eine robuste Fahrweise im Straßenverkehr zu garantieren. Weiterhin erfordern viele Softwaremodule, wie zum Beispiel die Trajektorienplanung, eine hoch genaue und verlässliche Schätzung der eigenen Pose.

In diesem Beitrag wird ein Verfahren vorgestellt, welches auf Basis einer hochgenauen digitalen Karte mit Landmarken, also leicht wiedererkennbaren Objekten in der Umgebung, und aktuellen Sensormessungen des automatisierten Fahrzeuges die Fahrzeugpose ermittelt. Dabei wird ein neuartiges Verfahren mit tiefen neuronalen Netzen präsentiert, welches aktuelle Messungen und Landmarken aus der Karte als 2D Punktlisten verarbeitet. Die Ergebnisse werden auf einem neu erstellten Datensatz in verschiedenen Szenarien in urbanen und ländlichen Gebieten evaluiert und es wird gezeigt, dass die vorgestellte Methode state-of-the-art Genauigkeiten im Bereich von 20 cm bei einer Rechenzeit von nur 2 ms erreicht. Weiterhin erfolgt ein Vergleich mit einer Netzwerkarchitektur, welche anstatt der Punktlisten auf Rasterkarten basiert, um die Vorteile der direkten Verarbeitung der Punkte aufzuzeigen.

**Schlüsselwörter:** Deep Learning, Digitale Karte, Lokalisierung, Punktwolken

## 1 Einleitung

Unter der Selbstlokalisierung eines automatisierten Fahrzeuges versteht man die Aufgabe, die eigene Position und die Ausrichtung innerhalb eines gegebenen Koordinatensystems zu bestimmen [14]. Diese Informationen sind essentiell für den sicheren und zuverlässigen Betrieb von automatisierten Fahrzeugen, da mit Hilfe der Positionsdaten unter anderem Informationen aus einer digitalen Karte mit in die Umgebungswahrnehmung integriert werden können [4]. Eine weit verbreitete Methode zur Bestimmung der Position ist die Verwendung von globalen Satellitennavigationssystemen (GNSS), wie beispielsweise das Global Position System (GPS). Die Genauigkeiten dieser Systeme hängt stark von Umwelteinflüssen (z.B. atmosphärischer Störungen) ab und kann in manchen Fällen durch Mehrwegausbreitungen in Häuserschluchten bei nur 10 m liegen [7]. Besonders in hoch komplexen und unstrukturierten Gebieten, zum Beispiel in der Stadt, ist jedoch ein präzises Lokalisierungsergebnis im Bereich von 20 – 50 cm erforderlich. Ein Lösungsansatz stellt die Nutzung von Korrekturdaten für die empfangenen GPS-Information dar (DGPS), womit Genauigkeiten im Zentimeterbereich erreicht werden können. Jedoch ist

---

\*Die Autoren sind vom Institut für Mess-, Regel- und Mikrotechnik, Universität Ulm, Albert-Einstein-Allee 41, 89081 Ulm, Deutschland. Email: nico.engel@uni-ulm.de, klaus.dietmayer@uni-ulm.de

auch ein DGPS System von Abschattungen in Häuserschluchten betroffen. Im Bereich der Fahrerassistenzsysteme und des automatisierten Fahrens sind Verfahren für die Lokalisierung verbreitet, die verschiedenen Sensoren, wie Lidar, Radar, Kamera und GPS kombinieren und mit Hilfe einer hochgenauen digitalen Karte die Fahrzeugpose schätzen. Typischerweise werden Lokalisierungsverfahren in zwei Gruppen eingeteilt: rasterkartenbasierte [15, 8] und landmarkenbasierte Methoden [2, 11]. Landmarken sind statische und wiedererkennbare Objekte in der Umgebung, wie zum Beispiel Lichtsignalanlagen, Bäume oder Straßenschilder, die mit verschiedenen Merkmalen (hochgenaue 2D Position, Größe, etc) in einer digitalen Karte gespeichert werden. Während der automatisierten Fahrt werden (vorverarbeitete) Sensormessungen mit der Karte abgeglichen um die eigene Position innerhalb des Kartenkoordinatensystems zu schätzen.

In diesem Beitrag wird ein Verfahren für die Selbstlokalisierung basierend auf Landmarken und multimodalen Sensormessungen vorgestellt, das zur Posenschätzung trainierte tiefe neuronale Netze verwendet. Die landmarkenbasierte Karte wird a-priori aufgenommen und als gegeben angesehen. Während der Fahrt werden aus den Messungen Features extrahiert, die mithilfe eines tiefen neuronalen Netzwerkes mit der Karte verglichen werden und die Fahrzeugpose geschätzt. Eine große Herausforderung hierbei ist die hohe Dynamik der Umgebung: Landmarken, die zum Zeitpunkt der Kartenerstellung vorhanden sind, können während der Lokalisierungsfahrt entfernt worden sein, oder statische Objekte (wie zum Beispiel parkende Autos) können neu auftauchen. Weiterhin ist sowohl die Anzahl der Landmarken aus der Karte, als auch die Anzahl der aktuellen Messungen für jeden Zeitschritt unbekannt. Traditionelle lernbasierte Ansätze wie Convolutional Neural Networks (CNN) sind für dieses Problem nicht geeignet, da die Eingangsdaten in einer geordneten und strukturierten Form vorliegen müssen, was bei 2D Punktlisten (Messungen und Landmarken) nicht gegeben ist. Eine Möglichkeit, dieses Problem zu umgehen besteht darin, die Punkte aus Messungen und Landmarken in einer Rasterkarte (Graustufenbild) einzutragen.

Zur Lösung des Problems der Selbstlokalisierung wird eine neuartige Netzwerkarchitektur vorgestellt, die auf der kürzlich eingeführten PointNet-Struktur basiert [9] und dahingehend modifiziert, dass zwei unabhängige Punktlisten verarbeitet und aufeinander abgebildet werden können. Das Ziel ist es, aus den Eingangsdaten Features zu generieren, die für die Posenschätzung verwendet werden können. Im Gegensatz zu dieser auf Punktlisten basierenden Methode wird zum Vergleich ein weiterer Ansatz auf Basis von Rasterkarten gezeigt, der als Eingang Bilder verwendet. Außerdem lassen sich die vorgestellten Ansätze mit bekannten Filtermethoden, wie einem Extended Kalman Filter (EKF), kombinieren, so dass eine noch genauere Lokalisierung ermöglicht werden kann.

## 2 Selbstlokalisierung mit Neuronalen Netzen

### 2.1 Digitale Karte und Messungen

Für die spätere Evaluierung wurde ein neuer Datensatz mit realen Szenarien in urbanen und ländlichen Gebieten erstellt. Dabei wurde für die Aufnahme ein Fahrzeug verwendet, welches mit einem Stereokamerasetup, Lidar- und Radarsensoren, sowie einem DGPS-System ausgestattet ist. Das DGPS-System dient der Generierung einer Ground-Truth Trajektorie, damit die Genauigkeit des vorgestellten Lokalisierungsverfahrens evaluiert

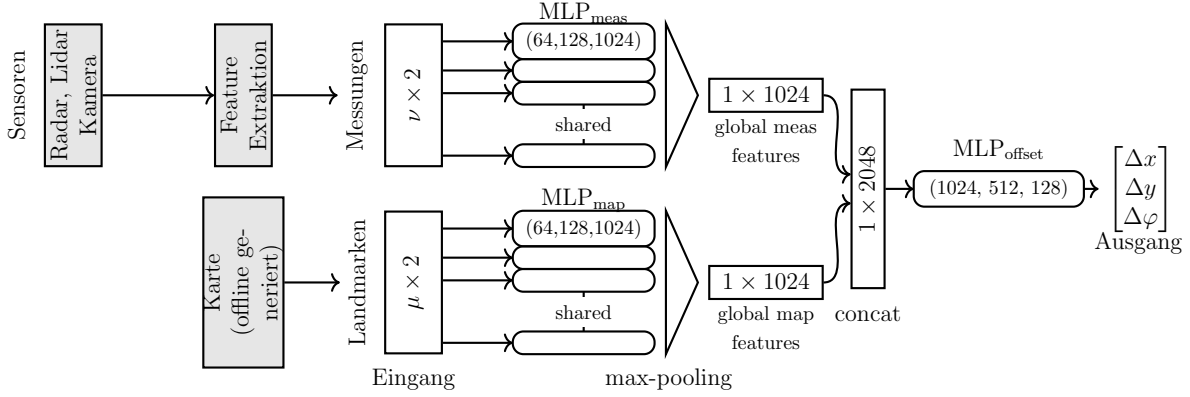


Abbildung 1: Übersicht der Netzwerkarchitektur auf Basis von Punktlisten [3]. Als Eingang dienen die vorverarbeitete Messungen und die Landmarken aus der digitalen Karten. Mittels Multilayerperzeptron, welches die Gewichte für jeden Punkt der Messungen und der Landmarken teilt, und einer maxpool Operation werden für beide Eingangspunktlisten globale Featurevektoren erzeugt. Anschließend werden diese zusammengeführt und mittels eines dritten MLP der Ausgang (Korrekturvektor) geschätzt.

werden kann. Die landmarkenbasierte Karte wurde im Dezember 2017 aus insgesamt drei Rundfahrten auf der Teststrecke in Ulm-Lehr an verschiedenen Tagen erstellt und enthält 3860 Landmarken, wobei 1731 aus Lidarmessungen, 1411 aus Kamerabildern und 718 aus Radarmessungen kommen. Unabhängig davon wurden acht Messfahrten im November 2018 auf der Teststrecke durchgeführt, um Messungen zu generieren. Insgesamt enthält der Datensatz 145 000 Zeitschritte, die neben den Sensormessungen auch eine Ground-Truth Pose des Fahrzeugs beinhalten.

## 2.2 Methode auf Basis von Punktlisten

Das Ziel ist es, die Fahrzeugpose  $\mathbf{p} = [x, y, \varphi]^T$  in Bezug auf die Landmarken in der Karte  $\{\mathbf{z}_1, \dots, \mathbf{z}_\nu\}, \mathbf{z}_{\{\cdot\}} \in \mathbb{R}^2$  und durch die Nutzung der Messungen  $\{\mathbf{m}_1, \dots, \mathbf{m}_\mu\}, \mathbf{m}_{\{\cdot\}} \in \mathbb{R}^2$  zu bestimmen. Da eine direkte Bestimmung der Pose in einem globalen Koordinatensystem, wie zum Beispiel dem Universal Transverse Mercator (UTM), eine große Herausforderung darstellt, wird in diesem Beitrag vorgeschlagen, den Offset  $\Delta\mathbf{p}$ , also die Differenz, zu einer vorherigen, bekannten Pose  $\mathbf{p}_{t-1}$  zu schätzen. Die Idee besteht darin, eine Abbildung der Messungen und den Landmarken auf den Offset der Fahrzeugpose zu lernen [3]

$$f(\{\mathbf{z}_1, \dots, \mathbf{z}_\nu\}, \{\mathbf{m}_1, \dots, \mathbf{m}_\mu\}) \rightarrow [\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{\varphi}]. \quad (1)$$

Ist die Pose des vorherigen Zeitschritts  $\mathbf{p}_{t-1}$  bekannt, so lässt sich mit Hilfe des Netzausgangs auf die aktuelle Pose schließen

$$\mathbf{p}_t = \mathbf{p}_{t-1} + [\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{\varphi}]. \quad (2)$$

Das Lernziel ist die Minimierung der Differenz von dem Ground-Truth Pose Offset  $[\Delta x, \Delta y, \Delta \varphi]$  zum prädizierten Offset  $[\Delta\hat{x}, \Delta\hat{y}, \Delta\hat{\varphi}]$ . Da der translatorische Offset und

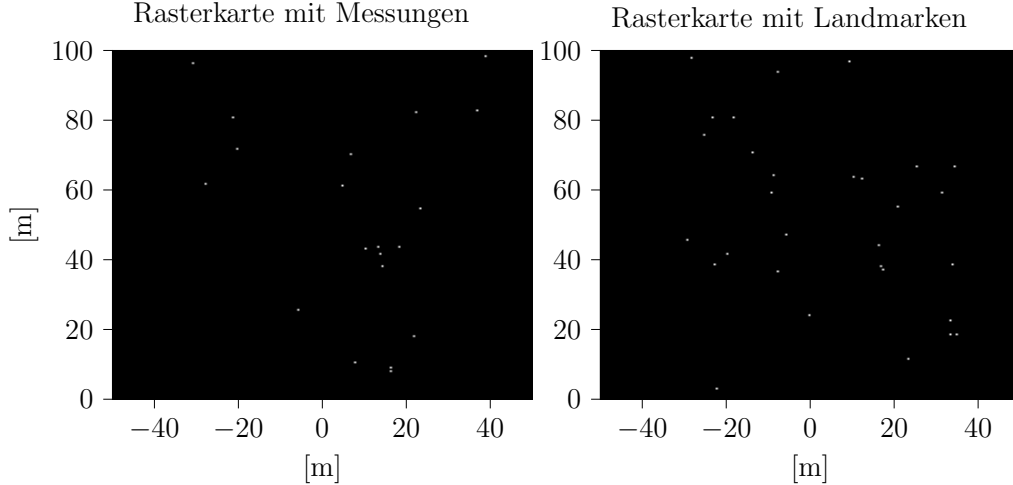


Abbildung 2: Darstellung der Landmarken und Messungen in einer Rasterkarte.

der rotatorische Offset unterschiedliche Einheiten besitzen, werden sie für die Minimierung unabhängig behandelt

$$\mathcal{L}_{\text{tran}} = \mathbb{E} [(\Delta\hat{x} - \Delta x)^2] + \mathbb{E} [(\Delta\hat{y} - \Delta y)^2], \mathcal{L}_{\text{rot}} = \mathbb{E} [(\Delta\hat{\varphi} - \Delta\varphi)^2]. \quad (3)$$

Zusätzlich wird die Gewichtung der beiden Loss-Terme mit gelernt [6], so dass für die Loss Funktion folgt

$$\mathcal{L} = \mathcal{L}_{\text{tran}} e^{-s_{\text{tran}}} + s_{\text{tran}} + \mathcal{L}_{\text{rot}} e^{-s_{\text{rot}}} + s_{\text{rot}}, \quad (4)$$

wobei  $s_{\text{tran}}$  und  $s_{\text{rot}}$  die jeweilige Unsicherheit der Aufgabe angibt und als Gewichtungsfaktor verstanden werden kann. Die verwendete Netzwerkarchitektur ist in Abb. 1 dargestellt. Die Messungen und die Landmarken aus der Karte liegen als 2D Punktlis-ten vor. Jeder Punkt aus der Liste der Messungen und jeder Punkt aus der Liste der Landmarken wird mit Hilfe mit jeweils einem unabhängigen Multilayerperzeptron (MLP) von  $\mathbb{R}^2$  in einen höher dimensional-ten Raum transformiert, hier  $\mathbb{R}^{1024}$ . Wichtig dabei ist, dass die Parameter der MLPs für die Messungen ( $\text{MLP}_{\text{meas}}$ ) geteilt werden. Das gleiche gilt für die MLPs der Landmarken ( $\text{MLP}_{\text{map}}$ ) (siehe Abb. 1). Es folgt eine *maxpool* Operation über die 1024 Dimensionen, so dass ein Vektor der Größe  $1 \times 1024$  folgt, der globaler Featurevektor genannt wird. Es ist ersichtlich, dass durch diese Operation unabhängig von der Anzahl der Eingangspunkte als auch unabhängig von der Reihenfolge der selbe Featurevektor entsteht. Somit sind die eingangs erwähnten Probleme der Permutationsinvarianz (Punktlis-ten sind unsortiert) und der unbekannt-ten Anzahl der Datenpunkte zu jedem Zeitschritt gelöst. Die beiden globalen Featurevektoren werden anschließend zusammengeführt und mit Hilfe eines dritten MLPs ( $\text{MLP}_{\text{offset}}$ ) wird der Offset der Pose geschätzt.

### 2.3 Methode auf Basis von Rasterkarten

Die Idee der Selbstlokalisierung von automatisierten Fahrzeugen auf Basis von Rasterkarten und neuronalen Netzen orientiert sich an der Struktur von RegNet [10]. Der Eingang besteht dabei aus zwei Rasterkarten (Graustufenbilder), die das Sichtfeld (engl.: field of

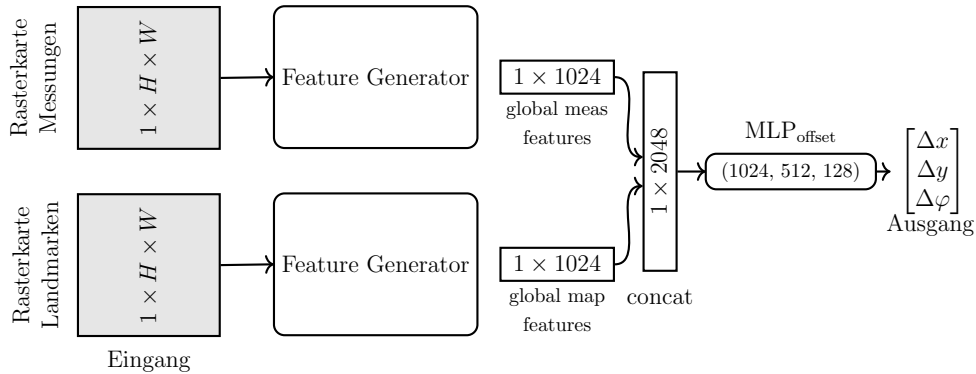


Abbildung 3: Übersicht der Netzwerkarchitektur auf Basis von Rasterkarten. Als Eingang dienen die Rasterkarten der Landmarken und Messungen. Um die Features zu erzeugen, werden jeweils die Featuregeneratoren von GoogleNet [13], InceptionV4 [12] und ResNet [5] verwendet, die im nächsten Schritt zusammengeführt und mit Hilfe eines MLPs ein Korrekturvektor geschätzt.

view - FOV) des Fahrzeuges und die Umgebung darstellen. Jedes Pixel der Rasterkarte modelliert einen Bereich des Sichtfeldes der Größe  $10 \text{ cm} \times 10 \text{ cm}$ . Befindet sich innerhalb dieses Bereichs entweder eine Landmarke oder eine Sensormessung zum aktuellen Zeitpunkt, so wird der Bereich der zugehörigen Rasterkarte weiß eingefärbt. Freiraum wird dabei durch ein schwarzes Pixel dargestellt. Beispielhaft sind in Abbildung 2 für einen Zeitschritt die generierten Rasterkarten für die Landmarken im FOV des Fahrzeuges sowie die aktuellen Sensormessungen dargestellt. Anhand der gezeigten Rasterkarte wird ersichtlich, dass in dem Graustufenbild, welches bei einer Rastergröße von  $10 \text{ cm} \times 10 \text{ cm}$  und einem FOV von  $100 \text{ m}$  eine Größe von  $1000 \times 1000$  Pixel besitzt, im Durchschnitt über 99.9% der Rasterzellen Freiraum sind und somit keine Informationen enthalten. Die grundlegende Struktur ist ähnlich zu Abbildung 1 mit dem Unterschied, dass die Messungen und Landmarken wie bereits erwähnt zu Rasterkarten transformiert werden und somit in einer geordneten Struktur vorliegen. Für die Generierung der globalen Features der Landmarken (*global map feat*) und der Messungen (*global meas feat*) werden bekannte Netzwerkarchitekturen verwendet. In diesem Beitrag werden jeweils die Featuregeneratoren von GoogleNet [13], InceptionV4 [12] und ResNet [5] verwendet und miteinander verglichen. Der Unterschied zum Netzwerk mit Punktlisten besteht darin, dass die Features anstatt mit MLPs und einer maxpool Operation mit Hilfe von Convolutional Neural Networks (CNN) erstellt werden. Diese Featurevektoren werden anschließend verknüpft um einen Zusammenhang zwischen den beiden Eingangsrasterkarten zu lernen und mittels des  $\text{MLP}_{\text{offset}}$  zu einem Korrekturvektor zusammengefasst. Die resultierende Architektur ist in Abbildung 3 dargestellt. Für das Training und die Inferenz wurde die Größe der Rasterkarten auf  $320 \times 320$  Pixel reduziert um auf Kosten der Genauigkeit Rechenzeit und Rechenleistung einzusparen.

## 2.4 Filterung des Netzausgangs

Da das Netzwerk ohne eine zeitliche Komponente trainiert und eingesetzt wird, kann es zum Beispiel vorkommen, dass eine Position präzisiert wird, die entgegen der eigentli-

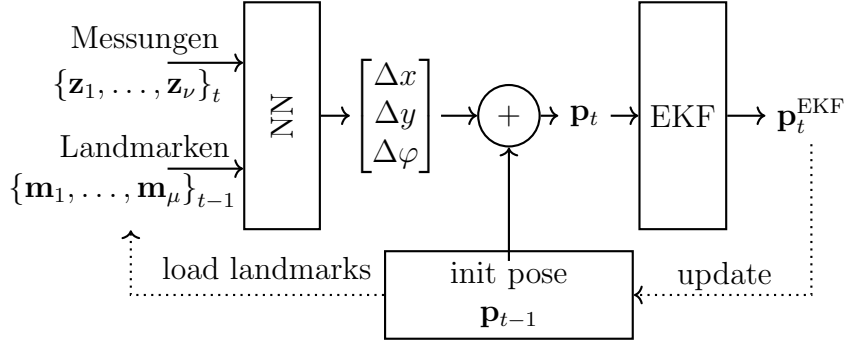


Abbildung 4: Implementierung des EKF [3]. Mit Hilfe der Messungen und Landmarken wird mittels des vorgestellten neuronalen Netzwerkes (NN) ein Offset-Vektor geschätzt, der eine initiale Pose korrigiert. Die korrigierte Pose  $\mathbf{p}_t$  dient als Eingang eines Extended Kalman Filters mit einem CTRV Bewegungsmodells. Der Ausgang ist eine geglättete Posenschätzung  $\mathbf{p}_t^{\text{EKF}}$ .

chen Fahrtrichtung liegt. Um diesen Effekt zu verhindern, wird zusätzlich ein Extended Kalman Filter (EKF) implementiert, welches als Messung die um den Netzausgang korrigierte Pose des Fahrzeuges erhält und eine zeitlich geglättete Trajektorie ausgibt. Die resultierende Pose wird dabei als neuer Startzustand für den nächsten Zeitschritt verwendet. Die Struktur ist in Abbildung 4 dargestellt. Als Bewegungsmodell wird ein Modell konstanter Geschwindigkeit und konstanter Drehrate (engl. constant turn rate and velocity - CTRV) eingesetzt. Die Rauschparameter wurde dabei experimentell bestimmt und die Updatefrequenz beträgt 20 ms.

### 3 Experimente und Evaluation

In diesem Kapitel werden die durchgeführten Experimente evaluiert. Insgesamt werden für den auf Punktlisiten basierenden Ansatz zwei verschiedene Versuche mit vergleichbaren state-of-the-art Methoden verglichen. Zum Einem wird der Fall untersucht, dass eine ungenaue (z.B. durch atmosphärische Störungen verrauschte) GPS-Position als initiale Lösung angenommen wird und die Aufgabe des neuronalen Netzes besteht darin, diese mit Hilfe der zum aktuellen Zeitpunkt registrierten Messungen und der digitalen Karte zu korrigieren (GPS-basierter Ansatz). Zusätzlich wird die in Kapitel 2.4 vorgestellte Filter-basierte Lokalisierung evaluiert. Die Besonderheit liegt darin, dass nur eine einzige initiale (und möglicherweise verrauschte) GPS-Messung notwendig ist. Alle weiteren Zeitschritte nehmen als initiale Position die vorherige durch Netzausgang und EKF korrigierte Position an. Für die auf Rasterkarten basierte Methode werden, wie in Kapitel 2.3 beschrieben, verschiedene Featuregeneratoren (GoogleNet, InceptionV4 und ResNet) verwendet und auf Genauigkeit des Lokalisierungsergebnisses sowie Rechenzeit untersucht. Als Evaluationsmetrik wird der *Root Mean Square Error* (RMSE) verwendet. Außerdem werden ähnliche, auf Landmarken basierende Ansätze auf dem vorgestellten Datensatz zum Vergleich evaluiert. Die Ergebnisse sind in Tabelle 1 dargestellt. Für die auf Punktlisiten basierenden Ansätze werden für die GPS-basierte Methode Positionsgenauigkeiten im Bereich von 23 – 27 cm erreicht bei einer Rechenzeit von nur 1.7 ms. Der Nachteil dieser Methode liegt darin, dass der Algorithmus in jedem Zeitschritt von einem GPS-Signal

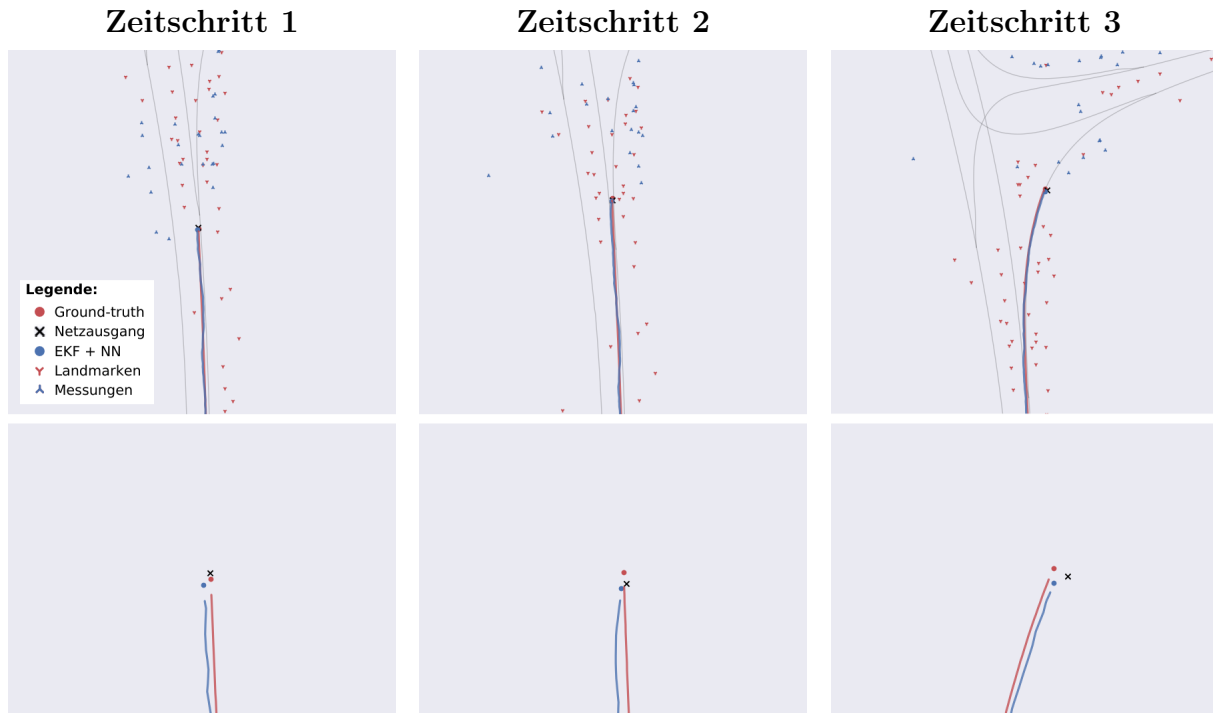


Abbildung 5: Ergebnisse der Testsequenz auf dem Ulm-Lehr Datensatz. Der Netzausgang kommt aus dem Netzwerk auf basis von Punktlisten. Ein roter Punkt (●) symbolisiert die Ground-Truth Position, ein blauer Punkt (●) der vom EKF geglättete Netzausgang (×). Weiterhin sind die Landmarken in den oberen Grafiken als rote Kreuze und die aktuellen Messungen als blaue Kreuze dargestellt.

abhängt, welches beispielsweise in städtischen Szenarien durch Abschattungen nicht immer gegeben ist. Um dieses Problem zu umgehen wird der Netzausgang gefiltert und die geläutete Pose als neuer Startzustand gewählt (NN + EKF und NN + EKF + GPS in Tabelle 1). Somit ist der Algorithmus nur noch von einer initialen GPS-Messung abhängig und erreicht ohne weitere GPS-Messungen Genauigkeiten im Bereich von 30 – 50 cm. Werden zusätzlich vorhandene und verrauschte GPS-Messung im EKF fusioniert, erhöht sich die Genauigkeit auf  $\approx 20$  cm. Der Einsatz eines EKF hat auf die Rechenzeit einen vernachlässigbaren Einfluss.

Für die auf Rasterkarten basierenden Ansätze werden die drei vorgestellten Featuregeneratoren verglichen. GoogleNet, InceptionV4 und ResNet erreichen ähnliche Genauigkeiten im Bereich von 40 – 60cm. Der Nachteil dieser Methoden liegt in der hohen Rechenzeit von 20 – 80 ms. Dies ist darauf zurückzuführen, dass durch die Umwandlung der Landmarken und Messungen in Rasterkarten ein großer Überschuss an Daten generiert wird, der zur Lösung der Aufgabe keinen Vorteil bringt (unbelegte Zellen). Dieses Ergebnis legt nahe, dass die Verwendung von Punktlisten für die Selbstlokalisierung mit Landmarken besser geeignet ist.

Der Iterative Closest Point (ICP) Algorithmus [1] versucht mittels Rotation und Translation zwei Punktwolken aufeinander abzubilden, indem die Abstände zwischen den Punkten der beiden Punktwolken iterativ minimiert werden. Da die Messungen und die Landmarken in dem vorgestellten Datensatz sich in einigen Zeitschritten stark unterscheiden, erreicht der ICP Algorithmus eine durchschnittliche Genauigkeit von nur 1.17 m. Weiter-

Tabelle 1: Übersicht der durchgeführten Experimente mit den zugehörigen Genauigkeiten (RMSE) und Rechenzeiten.

Experiment	RMSE			Rechenzeit
	$x$ [m]	$y$ [m]	$\varphi$ [°]	[ms]
<b>Punktlisten-basiert:</b>				
NN (GPS-basiert)	0.27	<b>0.23</b>	0.85	1.7
NN + EKF	0.56	0.33	1.20	2.1
NN + EKF + GPS	<b>0.27</b>	0.24	<b>0.82</b>	2.3
<b>Rasterkarten-basiert:</b>				
GoogleNet [13]	0.47	0.40	1.32	79.9
InceptionV4 [12]	0.61	0.52	1.22	62.6
ResNet [5]	0.52	0.45	1.7	20.9
<b>Vergleichbare Ansätze:</b>				
ICP [1]	1.17	1.4	4.96	7.6
EKF + GPS	0.59	0.54	6.58	<b>0.1</b>
RFS-MCL + GPS [11]	0.28	0.26	1.99	28

hin wurde die Pose mittels Extended Kalman Filter (EKF) geschätzt, wobei ungenaue GPS-Positionen als Messungen verwendet wurden (RMSE  $\approx 0.5$  m und  $\approx 6.5^\circ$ ). Als state-of-the-art Methode wird die Random Finite Set Monte-Carlo Lokalisierung (RFS-MCL) von Stübler et al. [11] herangezogen, die vergleichbare Ergebnisse erreicht. Der Vorteil des vorgestellten Lokalisierungsverfahrens auf Basis tiefer neuronaler Netze liegt neben der hohen Genauigkeit in der sehr schnellen Rechenzeit von nur 2 ms im Vergleich zu den 28 ms bei der RFS-MCL.

Eine kurze Sequenz der Ergebnisse des Netzwerkes auf Basis von Punktlisten auf der Teststrecke ist exemplarisch in Abb. 5 dargestellt. Die obere Reihe zeigt einen größeren Ausschnitt mit dem Straßenverlauf, den in der Karte gespeicherten Landmarken, den Messungen sowie der Ground-Truth und der geschätzten Trajektorie (und Pose). In der unteren Reihe ist lediglich ein vergrößerter Ausschnitt gezeigt, um die Genauigkeit der Lokalisierung zu beurteilen. Es ist ersichtlich, dass die vorgestellte Methode ausreichende Genauigkeiten für das automatisierte Fahren liefert und mit Hilfe der Filterung (EKF) des Netzausgangs die Ground-Truth Trajektorie auf wenige Zentimeter genau geschätzt wird.

## 4 Zusammenfassung

In diesem Beitrag wurde ein Verfahren für die Selbstlokalisierung von autonomen Fahrzeugen auf Basis von tiefen neuronalen Netzen vorgestellt. Dabei wurden zwei Ansätze gewählt, die auf verschiedenen Eingangsformen basieren. Die Besonderheit liegt darin, dass durch die an PointNet angelegte Netzwerkarchitektur zwei unabhängige und ungeordnete 2D Punktlisten verarbeitet und aufeinander abgebildet werden können. Das



Netzwerk ist dabei ein Regressor, der den Offset der Fahrzeugpose zu einer bekannten, vorherigen Pose schätzt. Um das Lokalisierungsergebnis zu verbessern und zu glätten, wird ein Extended Kalman Filter mit einem Bewegungsmodell eingesetzt, welches als Messung den Ausgang des vorgestellten Netzwerkes erhält. Die Genauigkeit der vorgestellten Methode liegt im Bereich der Anforderungen für den sicheren Fahrbetrieb in städtischen sowie ländlichen Regionen und ist mit aktuellen bekannten Methoden vergleichbar. Zusätzlich kann die Methode mit nur einer einzigen initialen (und möglicherweise) ungenauen GPS Messungen zufriedenstellende Ergebnisse liefern. Weiterhin wurde gezeigt, dass die Rechenzeit des neuronalen Netzwerkes in Verbindung mit dem EKF bei etwa 2 ms liegt.

## Literatur

- [1] P. J. Besl and N. D. McKay. Method for registration of 3-D shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–607. International Society for Optics and Photonics, 1992.
- [2] H. Deusch, S. Reuter, and K. Dietmayer. The labeled multi-bernoulli SLAM filter. *IEEE Signal Processing Letters*, 22(10):1561–1565, 2015.
- [3] N. Engel, S. Hoermann, M. Horn, V. Belagiannis, and K. Dietmayer. Deeplocalization: Landmark-based self-localization with deep neural networks. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 926–933, Oct 2019.
- [4] F. Gies, A. Danzer, and K. Dietmayer. Environment perception framework fusing multi-object tracking, dynamic occupancy grid maps and digital maps. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3859–3865. IEEE, 2018.
- [5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [6] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.
- [7] T. Kos, I. Markežic, and J. Pokrajcic. Effects of multipath reception on GPS positioning performance. In *Proceedings ELMAR-2010*, pages 399–402. IEEE, 2010.
- [8] J. Levinson, M. Montemerlo, and S. Thrun. Map-based precision vehicle localization in urban environments. In *Robotics: Science and Systems*, volume 4, page 1. Citeseer, 2007.
- [9] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR)*, IEEE, 1(2):4, 2017.
- [10] N. Schneider, F. Piewak, C. Stiller, and U. Franke. Regnet: Multimodal sensor registration using deep neural networks. In *2017 IEEE intelligent vehicles symposium (IV)*, pages 1803–1810. IEEE, 2017.

- [11] M. Stübler, S. Reuter, and K. Dietmayer. Consistency of feature-based random-set Monte-Carlo localization. In *2017 European Conference on Mobile Robots (ECMR)*, pages 1–6. IEEE, 2017.
- [12] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [14] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. MIT press, 2005.
- [15] J. Wiest, H. Deusch, D. Nuss, S. Reuter, M. Fritzsche, and K. Dietmayer. Localization based on region descriptors in grid maps. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 793–799. IEEE, 2014.