

Schlussbericht

TWT GmbH Science & Innovation

Teil I - Kurzbericht
Teil II - Eingehende Darstellung

Berichtszeitraum
01. Juni 2024 – 31. August 2025

Autoren
Dr. Tobias Rößler, Franziska Ott, Hanna Ziegler, Dennis Otte

Zuwendungsempfänger



TWT GmbH Science & Innovation

Förderkennzeichen

19A24001B

Vorhabenbezeichnung und Laufzeit

NAOMI4Radar – Neuromorphe Algorithmen zur Optimierung der Radar-Sensorik in autonomen Fahrzeugen vom 1. Juni 2024 bis 31. August 2025



Finanziert von der
Europäischen Union
NextGenerationEU

Gefördert durch:



Bundesministerium
für Wirtschaft
und Energie

aufgrund eines Beschlusses
des Deutschen Bundestages

Projektträger:



I Inhaltsverzeichnis

II	Abbildungsverzeichnis	3
III	Tabellenverzeichnis	4
1	Teil I - Kurzdarstellung	5
1.1	Ausgangssituation und Motivation	5
1.2	Voraussetzung	5
1.3	Zielsetzung und angestrebter Nutzen	5
1.4	Wichtigste Ergebnisse und Erkenntnisse	6
1.4.1	Neuromorphe Pipeline mit Echtzeitverarbeitung.....	6
1.4.2	Evaluation neuromorpher Algorithmen.....	6
1.4.3	Radardatensatz	6
1.5	Bedeutung der Ergebnisse für Forschung, Wirtschaft und Gesellschaft	6
2	Teil II - Eingehende Darstellung	7
2.1	Wissenschaftlicher und technischer Stand zu Beginn und Ende des Vorhabens	7
2.1.1	Radarsensorik.....	7
2.1.2	Sparse Coding	8
2.1.3	Spiking Neural Networks	9
2.1.4	Resonate-and-Fire (RF)-Neuronen	9
2.1.5	Neuromorphic Computing	10
2.2	Wissenschaftlicher und technischer Stand zu Ende des Vorhabens	10
2.2.1	Radarsensorik.....	10
2.2.2	Sparse Coding	10
2.2.3	Spiking Neural Networks, Resonate-and-Fire (RF)-Neuronen, Neuromorphic Computing	12
2.3	Planung und Ablauf des Vorhabens	12
2.3.1	Strukturierung des Vorhabens	12
2.3.2	Erfolgskontrolle im Projektverlauf	13
2.3.3	Ressourcen	14
2.3.4	Zielanpassung	14
2.4	Projektergebnisse und Auswirkungen	14
2.4.1	AP 1 Algorithmenentwicklung	15
2.4.2	UAP 1.1 Erstellen eines Datensatzes	15
2.4.3	UAP 2.1 Hardwareaufbau der Ende-zu-Ende-Verarbeitungskette	18
2.4.4	UAP 2.2 Vorverarbeitung der Radarsensordaten zur Verarbeitung mit Intel Loihi 2	21
2.4.5	UAP 2.3 Implementierung einer Ende-zu-Ende-Verarbeitung von Radardaten nach Stand der Technik	23
2.4.6	UAP 2.4 Implementierung der neuromorphen Verarbeitungsmodelle zur Ausführung auf Intel Loihi 2	24

2.4.7	UAP 2.5 Echtzeit-Visualisierung der Radardatenverarbeitung	24
2.4.8	UAP 2.6 Technische Evaluation	24
2.4.9	AP 3 Projektkoordination und Ergebnisverbreitung.....	25
2.4.10	UAP 3.1 Projektmanagement	25
2.5	UAP 3.2 Publikation wissenschaftlicher Ergebnisse	25
2.6	Wichtigste Positionen des zahlenmäßigen Nachweises	26
2.7	Verwertbarkeit der Ergebnisse	26
2.8	Zusammenarbeit mit anderen Stellen oder außerhalb des Verbundprojekts	26
2.9	Darstellung des während des Vorhabens bekannt gewordenen Fortschritts auf diesem Gebiet bei anderen Stellen	27
2.10	Erfolgte und geplante Veröffentlichungen.....	27
VII	Literaturverzeichnis	28

II Abbildungsverzeichnis

Abbildung 1: Infineon Radar Sensor, verbaut in einem PKW von Mercedes-Benz	16
Abbildung 2: Hardwareaufbau der Ende-zu-Ende-Verarbeitungskette inkl. Referenzsystem	20
Abbildung 3: Setup für Echtzeitverarbeitung der Radardaten mit neuromorphem Prozessor Intel Loihi 2 im Kofferraum des Fahrzeugs	21
Abbildung 4: Rechenleistung Nvidia Jetson CPU vs. GPU für FFT- und Convolution-Berechnungen.....	23

III Tabellenverzeichnis

Tabelle 1: Planung des Vorhabens, Arbeitsaufwand nach Partner	13
Tabelle 2: Meilensteinplanung im Projekt NAOMI4Radar.....	14
Tabelle 3: Konfiguration und Leistungsparameter des Radarsensors	16
Tabelle 4: Energieverbrauchsmessungen State of the Art.....	25

1 Teil I - Kurzdarstellung

1.1 Ausgangssituation und Motivation

Das automatisierte Fahren zählt zu den zentralen Innovationsfeldern der deutschen Automobilindustrie. Mit der weltweit ersten für Endkunden zertifizierten Level-3-Fahrfunktion wurde durch Mercedes-Benz ein international beachteter Meilenstein gesetzt – ein Erfolg, zu dem auch wir als spezialisierter Entwicklungspartner im Bereich Simulation, Softwareentwicklung und Beratung maßgeblich beigetragen haben.

Unsere Mitwirkung an der Verifikation und Validierung dieser hochkomplexen Fahrfunktion hat gezeigt, dass klassische Rechenarchitekturen zunehmend an ihre Grenzen stoßen – insbesondere in Bezug auf Energieeffizienz, Latenz und Skalierbarkeit. Um den technologischen Vorsprung langfristig zu sichern und zukunftsfähige Lösungen zu schaffen, bedarf es neuer Ansätze in der Verarbeitung großer, heterogener Sensordatenströme in Echtzeit.

Ein vielversprechender Ansatz zur Steigerung von Effizienz und Reaktionsfähigkeit ist das Neuromorphic Computing (NC). Neuromorphe Hardware und Algorithmen orientieren sich am menschlichen Gehirn und können die Signalverarbeitung in intelligenten Fahrzeugsystemen erheblich effizienter machen.

Für uns als Entwicklungspartner eröffnet NC neue Möglichkeiten, um unsere Kunden bei der Realisierung robuster, skalierbarer und ressourcenschonender Fahrzeugintelligenz zu unterstützen – und damit einen aktiven Beitrag zu Sicherheit, Nachhaltigkeit und Reichweite automatisierter und elektrifizierter Fahrzeuge zu leisten.

1.2 Voraussetzung

TWT hat sich aufgrund der Unternehmensstrategie wissenschaftliche Erkenntnisse in technische Produkte zu integrieren als ein führender Innovationspartner in der Automobil- und Energiebranche etabliert und setzt sich aktiv mit aktuellen technologischen Herausforderungen wie KI, E-Mobilität, Nachhaltigkeit und autonomem Fahren auseinander und hat in diesem Zuge jahrzehntelange Erfahrung in der Durchführung von (inter-)nationalen Forschungsprojekten. Im Bereich automatisiertes und autonomes Fahren beteiligt sich TWT in vielen öffentlich geförderten Forschungsprojekten, wodurch unter anderem neue Features für Tronis® prototypisch entstanden. Tronis® ist eine Umgebung zum virtuellen Prototyping und zur Absicherung von Fahrerassistenzsystemen, welche reale Fahrsituationen und Verkehrsszenarien effizient abbildet und das für das Testen, z. B. für die kamera- und radarbasierte Umgebungserfassung, herangezogen werden kann. Tronis® verfügt hierzu über eine Radarsensorsimulation.

1.3 Zielsetzung und angestrebter Nutzen

Das Projekt NAOMI4Radar war mit einer Laufzeit von 15 Monaten auf eine agile Entwicklungsstrategie sowie eine zeitnahe Verwertung der Ergebnisse ausgerichtet. Ziel des Vorhabens war die Bewertung des Reifegrads neuromorpher Technologien und die Untersuchung ihres Potenzials für die Radarsignalverarbeitung im automobilen Umfeld.

Dazu wurden drei zentrale Ziele verfolgt:

- Integration einer neuromorphen Radarverarbeitungskette in ein Mercedes-Benz-Versuchsfahrzeug als Nachweis der technischen Umsetzbarkeit,
- Entwicklung und Evaluation neuromorpher Algorithmen zur Substitution und Verbesserung klassischer Verarbeitungsschritte,

- Erstellung und Veröffentlichung eines umfangreichen Radardatensatzes zur Validierung und Weiterentwicklung durch Forschung und Industrie.

1.4 Wichtigste Ergebnisse und Erkenntnisse

1.4.1 Neuromorphe Pipeline mit Echtzeitverarbeitung

Ein zentrales Ergebnis des Projekts ist der erfolgreiche Nachweis der technischen Machbarkeit einer vollständig neuromorphen Radarsignalverarbeitung im Fahrzeug. Hierzu wurde ein FMCW-Front-Radarsensor von Infineon in ein Versuchsträgerfahrzeug von Mercedes-Benz integriert und über eine dedizierte Datenverbindung mit einer Recheneinheit im Fahrzeugheck gekoppelt. Diese Recheneinheit ermöglicht die parallele Verarbeitung der Radarrohdaten mittels zweier unterschiedlicher Architekturen: einer neuromorphen Pipeline auf Basis des Intel Loihi 2 und einer konventionellen GPU-basierten Signalverarbeitungskette.

Die simultane Auswertung der Datenströme in Echtzeit sowie die Visualisierung der Verarbeitungsergebnisse im zentralen Fahrzeugdisplay erlaubten einen direkten Vergleich beider Ansätze unter realen Verkehrsbedingungen. Damit wurde erstmals demonstriert, dass eine neuromorphe Radarsignalverarbeitung sowohl technisch als auch betrieblich in einem Fahrzeugumfeld integrierbar und einsetzbar ist.

1.4.2 Evaluation neuromorpher Algorithmen

Im Projekt wurden neuromorphe Alternativen zu zentralen Verarbeitungsschritten der klassischen Radarsignalverarbeitung entwickelt – einschließlich Frequenzanalyse, Signaltransformation und Objektdetektion. Für die Frequenzanalyse kam ein auf dem „Resonate-and-Fire“-Prinzip basierender Ansatz zum Einsatz, der eine kontinuierliche Verarbeitung bei reduzierter Latenz ermöglicht. Ein Sparse-Coding-Verfahren wurde zur Verbesserung der Signalqualität und Unterdrückung von Nebenkeulen entwickelt und zeigte dabei eine signifikante Energieeinsparung im Vergleich zu konventionellen GPU-basierten Methoden. Zudem wurde eine neuromorphe Umsetzung eines YOLO-Netzwerks zur Detektion erfolgreich in die Pipeline integriert, welche in Echtzeit ausgeführt werden kann und eine Genauigkeit auf dem Niveau klassischer Verfahren erreicht.

Unser Projektbeitrag umfasste insbesondere die Implementierung und Optimierung der konventionellen GPU-basierten Verarbeitungskette als Referenz sowie die Einbringung domänenspezifischen Know-hows in der Radarsignalverarbeitung zur Entwicklung und Bewertung der neuromorphen Algorithmen.

1.4.3 Radardatensatz

Ein weiterer wesentlicher Projekterfolg ist die Erstellung und Veröffentlichung eines öffentlich zugänglichen Radardatensatzes, der eine Vielzahl realistischer statischer und dynamischer Verkehrsszenarien umfasst – darunter Fahrzeuge, Fußgänger und standardisierte Testobjekte. Der Datensatz enthält synchronisierte Rohdaten eines FMCW-Radarsystems sowie korrespondierende LiDAR- und GPS-Daten, wodurch eine präzise Referenzierung und Nachvollziehbarkeit der Szenen gewährleistet ist. Diese Datenbasis stellt eine Ressource für die Validierung, Entwicklung und Benchmarking neuartiger Algorithmen in der Radarsignalverarbeitung dar und unterstützt sowohl akademische als auch industrielle Forschungsaktivitäten.

1.5 Bedeutung der Ergebnisse für Forschung, Wirtschaft und Gesellschaft

Die erstmalige prototypische Integration einer neuromorphen Radardatenverarbeitung in ein Fahrzeug im Rahmen von NAOMI4Radar markiert einen wichtigen technologischen Fortschritt in der Sensorverarbeitung. Die Ergebnisse belegen, dass Radarsignale in Echtzeit mit drastisch reduziertem Energiebedarf verarbeitet werden können – ein Meilenstein auf dem Weg zu intelligenten, energieeffizienten Sensorsystemen.

Das Projekt eröffnet neue Potenziale für energieeffiziente Edge-Anwendungen in Mobilität, Robotik und Industrie. Für die Gesellschaft bedeutet dies Fortschritte in Sicherheit, Nachhaltigkeit und Ressourceneffizienz, insbesondere durch leistungsfähigere und energieeffizientere Fahrerassistenzsysteme.

2 Teil II - Eingehende Darstellung

2.1 Wissenschaftlicher und technischer Stand zu Beginn und Ende des Vorhabens

2.1.1 Radarsensorik

Seit den 1990er Jahren nehmen Radarsensoren eine immer wichtig werdende Rolle im Bereich der Advanced Driver Assistance Systems (ADAS) ein. Sie werden beispielsweise bei Adaptive Cruise Control (ACC) oder dem Automated Emergency Braking (AEB) eingesetzt. Hierbei handelt

es sich meist um Frequenzmodulierte Radare (FMCW-Radar) mit einer Trägerfrequenz von 77GHz, welche die Position sowie die Relativgeschwindigkeit von Objekten erfassen: Das Radar sendet eine elektromagnetische Welle mit linear ansteigender Frequenz und einer Dauer im Mikrosekunden-Bereich aus – ein sogenannter Chirp –, welche von dem zu erkennenden Objekt reflektiert und anschließend durch das Radar wieder empfangen wird. Das empfangene Signal wird mit dem gesendeten Signal gemischt, wodurch sich ein Inferenzsignal ergibt. Dieses wird nun mittels eines ADC digitalisiert (ca. 500-1000 samples). Für die weitere Signalverarbeitung wird eine Vielzahl dieser Chirps zu einem Frame (ca. 128-250 Stück) zusammengefasst. Für die Winkelbestimmung werden zudem mehrere Empfangsantennen im Sensor (mind. 3 Stück) benötigt. Die digitalisierten Signale aller Empfangsantennen eines Frames werden zu einem sogenannten Raw Data Cube zusammengefasst und ausgewertet. Mit Hilfe einer 2D-Fast-Fourier-Transformation (2D FFT) wird eine Range-Doppler-Map erstellt (Sun et al., 2020). Anhand des Constant-False-Alarm-Rate-Algorithmus (CFAR) werden Peaks innerhalb der Range-Doppler-Map detektiert, wodurch sich auf die Entfernung und die relative Radialgeschwindigkeit des Objektes schließen lässt (Rohling, 1983). Aufgrund der bereits für eine FFT-Auswertung optimierten Hardware wird häufig auch für die Winkelbestimmung auf diese zurückgegriffen (Gentilho et al., 2019) auch wenn Ansätze wie MUSIC (Schmidt, 1986) oder ESPRIT (Roy and Kaillath, 1990) bessere Ergebnisse liefern.

Aufbauend auf der resultierenden Punktwolke wird eine Objektklassifizierung und eine Objektverfolgung durchgeführt. Neuste Algorithmen für die Objektklassifizierung verwenden häufig Machine-Learning-Algorithmen wie zum Beispiel Convolutional-Neural-Networks (CNNs) (Schumann et al., 2017; Patel et al., 2019) oder Recurrent-Neural-Networks (RNNs) (Schumann et al., 2017; Klarenbeek et al., 2017). Für die Objektverfolgung hingegen wird weiterhin auf etablierte Algorithmen wie den Extended Kalman Filter (Ikram und Ali, 2013) oder Bayesian Filter (Gordon et al., 1993) zurückgegriffen. Die Größte Herausforderung stellt hierbei das Assoziierungsproblem der detektierten Objekte zu ihren entsprechenden Trajektorien dar. Mögliche Lösungsansätze bilden der Generalized-Nearest-Neighbour-Algorithmus (GNN), welcher die Distanz zwischen den Trajektorien und der Detektion minimiert, oder den rechenintensiveren Algorithmus Joint-Probabilistic-Data-Association (JPDA) (Gamba, 2020, Abschnitt 7.4).

FMCW-Radarsysteme sind aufgrund ihrer Fähigkeit, Distanzen und Geschwindigkeiten mit hoher Präzision zu messen, besonders wertvoll. Auf Grund der Wellenlänge im mm-Bereich sind Radarsensoren, im Gegensatz zu anderen Sensortypen wie zum Beispiel Lidar oder Kamera, größtenteils unabhängig von Wettereinflüssen. Stattdessen stellt die Komplexität der Signalverarbeitung eine große Herausforderung dar. Insbesondere Umgebungsruschen und Interferenzen können die Datenqualität stark beeinträchtigen. Besonders das Phasenrauschen kann hier großen Einfluss auf die Inferenz nehmen (Siddiq et al., 2018).

2.1.2 Sparse Coding

Um die Signalqualität zu verbessern kann das sogenannte Sparse Coding, eine Methode der Signalverarbeitung und des maschinellen Lernens, eine Lösung bieten. Via Sparse Coding können wesentliche Merkmale aus Daten durch Darstellung als Linearkombinationen über einen kleinen Satz von Basisfunktionen extrahiert werden.

Der Locally Competitive Algorithm (LCA) bietet eine Lösung für das Sparse Coding Problem (Rozell et al., 2008). Dies ist besonders bei rauschintensiven Signalen wie Radar von Vorteil, da ankommende Datenströme mit den gelernten Basisfunktionen vervollständigt und somit "ent-rauscht" werden können (Kasenbacher et al., 2025). Auf herkömmlicher Hardware zeigt sich der Algorithmus in seiner Ausführung als zeitintensiv und somit schlecht einsetzbar. Der neuartige neuromorphe Intel Loihi Chip hingegen ermöglicht eine sehr effiziente Implementierung des LCA (Parpart et al., 2023). Dadurch kann eine Anwendung des Algorithmus in Echtzeit realisierbar werden.

Die Anwendung von Sparse Coding und LCA auf FMCW-Radar im automobilen Kontext ist noch ein relativ unerforschtes Gebiet.

Ein Beispiel ist der Einsatz eines Sparse Sensing basierten Ansatzes zur Minderung von Interferenzen zwischen Automobil-Radarsystemen (Fei et al., 2021). Durch die Anwendung des Block Kronecker Compressed Sensing (BKCS) Algorithmus wird nicht nur die gegenseitige Störung signifikant reduziert, sondern auch die Leistungsfähigkeit der Radardetektion durch optimierte Signalverarbeitung verbessert. In dieser Arbeit wird gezeigt, wie Sparse Coding die Präzision und Zuverlässigkeit von Radarsystemen in Fahrzeugen verbessern kann.

Darüber hinaus zeigt die Forschung im Bereich der Synthetic Aperture Radar (SAR) Bilder, wie durch die Anwendung von Sparse Coding und spektraler Clusterung (Rahmani & Akbarizadeh, 2015) eine effektive Segmentierung und Interpretation von Radarbildern erreicht werden kann. Die Kombination von unsupervised feature learning mittels non-negative matrix factorisation (NMF) und Sparse Coding ermöglicht eine deutlich verbesserte Unterscheidung und Analyse von SAR-Bilddaten.

Zusätzlich zu FMCW-Radar und SAR-Bildsegmentierung bietet Dictionary Learning und Sparse Coding neue Ergebnisse zur Entrauschung (Liu et al., 2022). Diese Technik trainiert ein Set an Basisfunktionen, um Rauschen signifikant zu reduzieren, während die Bildtexturen erhalten bleiben.

2.1.3 Spiking Neural Networks

Spiking neural networks (SNNs) sind eine Form von künstlichen neuronalen Netzen (ANNs), die sich an biologischen Neuronen orientieren und Informationen durch das präzise Timing von Aktionspotentialen oder Spikes verarbeiten. Aufgrund ihrer ereignisgesteuerten Datenverarbeitung, bei der Berechnungen nur dann erforderlich sind, wenn sich Spikes im System ausbreiten, ermöglichen sie eine massive Parallelität und einen geringen Stromverbrauch (Pfeiffer and Pfeil, 2018). SNN-basierte Sensorverarbeitung für autonomes Fahren ist ein Thema das derzeit zunehmend Aufmerksamkeit erfährt. Tiefe SNNs haben sich als wesentlich energieeffizienter erwiesen und erzielen vergleichbare Ergebnisse wie herkömmliche ANNs, z.B. bei der Objekterkennung (Jin et al., 2023; Su et al., 2023) und semantischen Segmentierung (Kim et al., 2022). In Viale et al. (2021) wurden SNN unter Verwendung des Loihi-Chips eingesetzt, um Fahrzeuge mit dynamischen Bildsensoren zu erkennen. Dabei sind die erzielten Resultate, mit denen einer Offline-Implementierung vergleichbar ist. Die Radarsensorverarbeitung mit SNNs ist ein relativ neues Forschungsfeld mit viel Potenzial. Vogginger et al. (2022) skizzieren dies für die allgemeine Anwendung von SNNs in der automatischen Radarverarbeitung mit Aussichten für energieeffiziente Implementierungen in autonomen Fahrzeugen. Dies wird auch durch den geringen Energieverbrauch und die vorteilhafte Leistung ihrer tiefen rekurrenten SNNs begründet, die für die Objektklassifizierung auf (der ROI-Sequenz) der Range Doppler Maps des CARRADAR-Datensatzes implementiert wurden (Vogginger et al., 2022). Darüber hinaus implementierte López-Randulfe et al. (2022) ihr SNN, spiking FT (S-FT), in den Loihi-Chip für FMCW-Radarsignale für Kraftfahrzeuge und bestätigte, dass das Netzwerk mathematisch äquivalent zur Fourier-Transformation ist. In Kombination mit ihrem angepassten Leaky-Integrate-and-Fire-Neuronenmodell, das eine direkte Kodierung analoger Signale in phasenkodierte Spikes ermöglicht, gibt es einen vielversprechenden Ausblick auf eine End-to-End-Pipeline für die Radarsignalverarbeitung, wie zum Beispiel die direkte Einbindung des Spiking OS-CFAR oder eines Analog-to-Spike-Converters (López-Randulfe et al., 2021).

2.1.4 Resonate-and-Fire (RF)-Neuronen

Resonate-and-Fire (RF)-Neuronen sind oszillierende Spiking Neuronen (Izhikevich 2021; Higuchi et al. 2024a), die in der Lage sind, Frequenzmuster aus eingehenden Signalen im Zeitbereich zu extrahieren. In letzter Zeit haben sie besonders im Kontext einer effizienteren Signalverarbeitung an Bedeutung gewonnen. Frühere Arbeiten zeigen, dass RF-Neuronen, implementiert im Intel

Loihi 2 (Davies et al., 2018), für die Berechnung der Kurzzeit-Fourier-Transformation (STFT) von Signalen genutzt werden können (Frady et al., 2022), was deutlich effizienter ist als die herkömmliche STFT-Methode (Shrestha et al., 2024). Zudem wurden RF-Neuronen erfolgreich als Encoder für die Erkennung von Interferenzen eingesetzt, indem sie Frequenzen extrahieren und gleichzeitig das Rauschen digitaler radarsensorischer Signale filtern (Hille et al., 2022) und sogar analoge Signale in Spikes umwandeln, während sie Spektraltransformationen durchführen (Auge and Mueller, 2020; Lehmann et al., 2023).

Die jüngsten Fortschritte bei RF-Neuronen haben deren Anwendungsgebiete erweitert, indem sie nicht nur zur Kodierung von Eingangssignalen, sondern auch zum Erlernen aufgabenspezifischer Frequenzen genutzt werden. RF-Neuronen, eingesetzt als harmonische Oszillatoren in Feedforward-SNNs, haben ihre Effektivität in der Bildklassifizierung (AlKhamissi et al., 2021), der Schätzung optischer Flüsse und der Audioklassifizierung (Frady et al., 2022) unter Beweis gestellt. Ein methodischer Durchbruch wurde mit dem Balanced Resonate-and-Fire (BRF)-Neuron in rekurrenten SNNs erzielt, welches für Bild- und Audioklassifizierungsaufgaben das Erlernen entscheidender Resonanzfrequenzen ermöglicht und eine deutlich schnellere Konvergenz bei wesentlich weniger Spikes im Netz als mit konventionelle Integratormodelle zeigt (Higuchi et al., 2024). Die deutliche bessere Energieeffizienz sowie die Effektivität der BRF-Neuronen versprechen insbesondere im Hinblick auf die bereits erfolgreiche Verwendung von regulären RF-Neuronen in der Radarsignalverarbeitung ein erhebliches Potenzial für ihre zukünftige Implementierung in neuromorpher Hardware.

2.1.5 Neuromorphic Computing

Intels Loihi 2, die neueste Generation neuromorpher Computing-Chips, repräsentiert einen signifikanten Fortschritt in der Emulation der neuronalen Dynamik des menschlichen Gehirns durch erhöhte Neuronen- und Synapsenzahlen. Diese Weiterentwicklung ermöglicht eine präzisere und komplexere Modellierung von spikenden neuronalen Netzwerken (SNNs), wobei eine Architektur zum Einsatz kommt, die auf asynchronen Spiking-Neuronenmodellen basiert. Diese Modelle zeichnen sich durch ihre Fähigkeit aus, zeitbasierte Informationen effizient zu verarbeiten, indem sie nur bei der Datenverarbeitung Energie verbrauchen, was eine signifikante Steigerung der Energieeffizienz gegenüber traditionellen, taktgetriebenen Prozessoren darstellt. Loihi 2 unterstützt zudem On-Chip-Lernen durch lokale Lernalgorithmen, die eine Echtzeitdatenanpassung ohne Rückführung zur zentralen Verarbeitungseinheit ermöglichen, und bietet durch seine Skalierbarkeit und Programmierbarkeit innovative Möglichkeiten für die Entwicklung fortschrittlicher KI-Anwendungen.

2.2 Wissenschaftlicher und technischer Stand zu Ende des Vorhabens

2.2.1 Radarsensorik

Zum Ende des Vorhabens lässt sich feststellen, dass sich der technologische Stand der Radarsensorik im Untersuchungszeitraum kaum verändert hat – dies entspricht dem ursprünglichen Ausgangspunkt der Arbeit, da die Entwicklung von RadarKomponenten nicht zu den Projektzielen gehörte. Zu Beginn des Vorhabens waren die wichtigsten Merkmale von Radarsystemen bereits durch hochauflösende 77 GHzFrontEnds, digitale BeamFormingArchitekturen und fortgeschrittene SignalProcessingAlgorithmen (z. B. CFAR, FMCWModulation) gekennzeichnet. Diese Technologien waren bereits in einer breiten Palette industrieller Anwendungen (AutomotiveADAS, industrielle Automation, Sicherheitsüberwachung) etabliert und boten eine Reichweite von bis zu mehreren hundert Metern bei einer Winkelauflösung von unter einem Grad.

2.2.2 Sparse Coding

Zum Abschluss des Vorhabens lässt sich festhalten, dass Sparse Coding als methodischer Kern für radarbezogene Signalaufbereitung vom konzeptionellen Ansatz zur praxistauglichen, hardwarebeschleunigten Verarbeitungskette gereift ist. Aufbauend auf den in Abschnitt 2.3.1.2 skizzierten

Grundlagen – Darstellung von Messdaten als Linearkombination weniger, charakteristischer Basisfunktionen – wurde ein Workflow etabliert, der (i) die lernbasierte Bestimmung geeigneter Basisfunktionen/Filter (Dictionary Learning), (ii) die schnelle Rekonstruktion mittels spärlicher Koeffizienten sowie (iii) die Einbettung in eine radartypische Auswertepipeline (u. a. CFAR) umfasst. Methodisch stützt sich der Rekonstruktionsschritt auf den Locally Competitive Algorithm (LCA; Rozell et al., 2008), dessen inhärente Dynamik (Schwellenbildung, laterale Inhibition und Schrumpfung) die gewünschten Sparsityeigenschaften zuverlässig erzwingt und zugleich robuste Entrauschungseffekte liefert.

Wissenschaftlich wurde im Projektverlauf insbesondere die konvolutionale Variante des Sparse Codings (CSC) konsolidiert: RadarSignale (Range–Angle bzw. Range–DopplerKarten) werden als Summe von Faltungsantworten zwischen gelernten Filtern (Dictionary) und spärlichen Aktivierungskarten modelliert. Dieses Prior- ermöglicht die effektive Dekonvolution der radartypischen Point-SpreadFunktion-, die Unterdrückung von Nebenkeulen sowie eine Trennung strukturierter Rückstreuung von Rauschen/Clutter. Damit werden drei zentrale Ziele adressiert: (1) SuperResolution- zur Separierung eng benachbarter Streuzentren, (2) Denoising zur Stabilisierung nachgelagerter CFARDetektoren- und (3) Compressive Sensing zur robusten Rekonstruktion aus unterabgetasteten ChirpFolgen. In exemplarischen Rekonstruktionen zeigte sich, dass -LCAbasierte -CSCAusgaben gegenüber konventionellen CFAR--Vorverarbeitungen deutlich bereinigte Range–Angle- -Karten liefern: Clutter, Rauschen und typische Radarnebenkeulen werden wirksam unterdrückt, Ziele erscheinen schärfer zentriert und von Artefakten entkoppelt. Ein repräsentatives Setup nutzte 140 Iterationen, Sparsity $\approx 0,75$ und $\tau = 2^{-8}$ bei einem CFAR-Schwellwert- von -75 dB; diese Parameter stellten in unseren Szenarien einen guten Kompromiss zwischen Rekonstruktionsgüte und Laufzeit her.

Technisch wurde ein wesentlicher Schritt durch die Übertragung des LCA auf neuromorphe Hardware (Intel Loihi 2) erreicht (vgl. Abschnitt 1.3.5). Durch die Abbildung von Filtern und Inhibitions-kernen auf die Kerne der Loihi--Architektur ergibt sich ein Solver, dessen Zeit- und Energiebedarf primär mit der Anzahl aktiver (nicht-null) Koeffizienten skaliert – ein Verhalten, das die Stärken gepulster neuronaler Netze auf dünnbesetzten Workloads direkt nutzt. In einem direkten Vergleich desselben Rekonstruktions--Tasks zeigte Loihi 2 eine deutlich günstigere Energiecharakteristik als eine Jetson -Orin GPU/CPU---Referenz: 19,5 J gesamt (14,5 J statisch + 4,9 J dynamisch) auf Jetson gegenüber 0,276 J gesamt (0,271 J statisch + 0,005 J dynamisch) auf Loihi 2. Parallel dazu fiel auch die Zeit zur- -Lösung auf Loihi 2 signifikant kürzer aus. Damit wird eine echtzeitnahe, energieeffiziente OnChip- -Denoising und Super- Resolution- Verarbeitung für FMCW--Radar auf Edge- Geräten- realistisch – insbesondere dort, wo niedrige Latenzen und strikte Leistungsbudgets entscheidend sind.

Aus Integrationssicht wurde eine radarspezifische Pipeline etabliert: Nach der FFT-basierten Vorverarbeitung (Range/Angle bzw. Range/Doppler) folgt die CSC--Rekonstruktion und eine angepasste CFAR--Detektion, deren Schwellenwahl von den verbesserten statistischen Eigenschaften (reduzierte Varianz, weniger Ausreißer) profitiert. Die Implementation adressiert praxisrelevante Aspekte wie I/O---Pfad und Quantisierung, die Abbildung komplexwertiger Radargrößen (I/Q) auf reelle Rechenpfade, die Ausnutzung (lokaler) Nachbarschaften in den Aktivierungskarten und die Kerndistribution über mehrere LoihiCores. Auf herkömmlicher CPU/GPU- -Hardware bleibt LCA aufgrund der iterativen Dynamik rechenzeitintensiv.

Zusammenfassend ist der Stand zu Ende des Vorhabens wie folgt: Sparse Coding – konkret als konvolutionales Sparse Coding mit LCA--Solver – ist wissenschaftlich fundiert, technisch in eine lauffähige, echtzeitnahe Pipeline überführt und auf neuromorpher Hardware hochgradig energieeffizient demonstriert. Die Methode liefert sichtbar verbesserte Radarrekonstruktionen (Entrauschung, Nebenkeulenunterdrückung, Schärfung), erhöht die Stabilität klassischer Detektionsverfahren und eröffnet kompressive Betriebsmodi mit reduzierten Messumfängen. Offene Arbeiten betreffen v. a. die Domänenrobustheit, die skalierende Hardwarezuordnung für

großdimensionale MIMO---Setups sowie die automobilgerechte Verifikation. Auf dieser Basis ist Sparse Coding als Schlüsselbaustein für niedriglatente, energiearme -Radar Perzeption- am Edge etabliert und anschlussfähig für weiterführende Produkt- und Serienentwicklungen.

2.2.3 Spiking Neural Networks, Resonate-and-Fire (RF)-Neuronen, Neuromorphic Computing

Zum Ende des Vorhabens hat sich das Verständnis über die Möglichkeiten und Grenzen der neuromorphen Radarverarbeitung deutlich erweitert. Durch die im Projekt entwickelten Modelle und Experimente konnte gezeigt werden, dass gepulste Neuronenmodelle wie das Balanced Resonate-and-Fire (B-RF)-Modell prinzipiell zur Abbildung von Reichweiteninformationen in Radarsignalen geeignet sind. Die Implementierung auf Intel Loihi 2 bestätigte die Funktionsfähigkeit des Ansatzes für die Range-Dimension und lieferte wertvolle Erkenntnisse zu Energieeffizienz, Synchronisation und numerischer Stabilität neuromorpher Systeme. Darüber hinaus wurde im Verlauf des Projekts erfolgreich eine End-to-End-Radarverarbeitung auf neuromorpher Hardware demonstriert. Durch die Kombination aus schneller Fourier-Transformation (FFT) und einem Sigma-Delta-YOLO-Netzwerk (TinyYOLO) konnte gezeigt werden, dass eine Pipeline von der Radarsensorik bis zur Objekterkennung auf Loihi 2 ausgeführt werden kann. Damit wurde ein neuer technischer Referenzpunkt geschaffen, der klassische Deep-Learning-Verfahren mit neuromorpher Verarbeitung vereint.

Zusätzlich lieferte die begleitende Forschung zu Trainingsregularisierung in Spiking-Netzen (Label Smoothing) grundlegende Einsichten in die Optimierung von SNNs: Sie zeigte, dass durch weiche Zielverteilungen der Energieverbrauch während des Lernens signifikant gesenkt und die Aktivitätsverteilung im Netzwerk verbessert werden kann. Diese Ergebnisse tragen dazu bei, neuromorphe Modelle künftig effizienter und stabiler zu trainieren.

Insgesamt hat sich der wissenschaftliche und technische Stand von anfänglich konzeptionellen Ansätzen zu praktisch validierten, hardwarebasierten Demonstrationen weiterentwickelt. Neuromorphe Signalverarbeitung kann heute als technisch realisierbar und wissenschaftlich fundiert gelten, wenngleich weitere Fortschritte in Hardwareunterstützung und algorithmischer Anpassung erforderlich bleiben.

2.3 Planung und Ablauf des Vorhabens

2.3.1 Strukturierung des Vorhabens

Die Projektplanung gliederte sich in drei Arbeitspakete (AP) mit klar definierten Verantwortlichkeiten (siehe Tabelle 1).

AP 1 umfasste die Entwicklung neuromorpher Algorithmen zur Verarbeitung von Radardaten eines Frequency-Modulated-Continuous-Wave-Sensors (FMCW) und lag in der Verantwortung der Universität zu Lübeck. Ziel war die Erprobung und Weiterentwicklung ausgewählter neuromorpher Verfahren für die Radarsignalverarbeitung. Zudem wurde im Rahmen dieses Arbeitspaketes ein Datensatz zur Evaluation der Algorithmen aufgenommen.

AP 2 beinhaltete die Entwicklung und Integration des Demonstrators im Fahrzeug und wurde von der TWT GmbH verantwortet. In diesem Arbeitspaket wurde die vollständige Ende-zu-Ende-Verarbeitungskette vom FMCW-Radarsensor über die Vorverarbeitung, die neuromorphe Verarbeitung mit Intel Loihi 2 bis hin zur Echtzeitvisualisierung im Fahrzeug umgesetzt. Der Demonstrator bildet damit das zentrale Bindeglied zwischen Forschung und industrieller Anwendung.

AP 3 war unter der Verantwortung der Mercedes-Benz AG und umfasste die Projektkoordination sowie die Verwertung und Ergebnisverbreitung. Mercedes-Benz übernahm die übergeordnete Steuerung, stellte das Integrationsfahrzeug bereit und gewährleistete die Einbindung der Ergebnisse in bestehende Entwicklungsprozesse. Zudem wurden die Ergebnisse durch wissenschaftliche Publikationen und eine frei zugängliche technische Dokumentation öffentlich gemacht.

Arbeitspaket	Mercedes-Benz	Univ. zu Lübeck	TWTGmbH Science & Innovation	Kurzbeschreibung
1. Algorithmenentwicklung	12	10,5	9	Entwicklung und Evaluation neuromorpher Algorithmen für die Verarbeitung von Radarsignalen
2. Demonstrator	6	3	20	Ende-zu-Ende-Integration der Verarbeitungskette mit Radarsensor, neuromorphem Chip und Visualisierung, im Fahrzeug
3. Projektkoordination und Ergebnisverbreitung	6	1,5	2	Sicherstellen des Projektfortschritts
Σ PM:	24	15	31	

Tabelle 1: Planung des Vorhabens, Arbeitsaufwand nach Partner

2.3.2 Erfolgskontrolle im Projektverlauf

Der Projektfortschritt wurde durch drei Meilensteine strukturiert (siehe Tabelle 2).

Meilenstein 1 – Aufbau im Fahrzeug:

Die Hardware wurde vollständig in das Fahrzeug integriert, die Datenkommunikation zwischen den Komponenten erfolgreich getestet und die Infrastruktur zur synchronisierten Datenerfassung eingerichtet.

Meilenstein 2 – Erstes methodisches Ziel:

Alle Verarbeitungsmodelle wurden in Software umgesetzt, auf der neuromorphen Plattform Intel Loihi 2 implementiert und in die Gesamtkette integriert. Messungen zu Laufzeit, Energieverbrauch und Verarbeitungsqualität bestätigten das Funktionsprinzip. Die Ergebnisse wurden im Rahmen des VDA-Kongresses ELIV 2025 vorgestellt.

Meilenstein 3 – Fertigstellung der Implementierung:

Der Demonstrator erreichte volle Ende-zu-Ende-Funktionalität mit Echtzeitverarbeitung und Visualisierung. Die Präsentation des Demonstrators erfolgt im Rahmen der Abschlussveranstaltung.

Insgesamt entsprach der Projektverlauf im Wesentlichen der ursprünglichen Planung. Einzelne inhaltliche Anpassungen wurden im Verlauf vorgenommen, um das Potenzial der Technologie gezielt auszuschöpfen.

Meilenstein 1 (M1): Aufbau im Fahrzeug	M7
Hardwareaufbau in Fahrzeug eingebaut	
Implementierung der vergleichenden Ende-zu-Ende-Implementierung nach Stand der Technik funktioniert	
Signalkommunikation evaluiert (I/O-Datenrate zwischen den Komponenten getestet)	
Technische Möglichkeit zur Datensatzerstellung im Straßenverkehr geschaffen	

Meilenstein 2 (M2): Erstes methodisches Ziel erreicht	M10
Trainiertes Verarbeitungsmodell in Software vorhanden	
HW-Limitationen sind berücksichtigt	
Bericht mit Abschätzungen zu Verarbeitungsfehler, Laufzeit und Energieverbrauch	
Meilenstein 3 (M3): Fertigstellung Implementierung	M12
Setup funktioniert Ende-zu-Ende mit Echtzeitverarbeitung der Radardaten	
Verarbeitung wird in Echtzeit visualisiert	
Datensatz ist fertiggestellt und veröffentlicht	
Projektende:	M15
Wissenschaftliche Ergebnisse sind publizierbar aufgearbeitet	
Technische Dokumentation ist fertiggestellt und veröffentlicht	

Tabelle 2: Meilensteinplanung im Projekt NAOMI4Radar

2.3.3 Ressourcen

TWT hat für die im Projekt durchgeführten Forschungsarbeiten ausschließlich Experten mit akademischen Abschlüssen in den Fachrichtungen Ingenieurwesen und Informationstechnologie eingesetzt. Trotz einiger personeller Veränderungen im Projektverlauf, die zu reduzierten Kapazitäten führten, konnten die wesentlichen Entwicklungsziele erreicht und die geplanten Demonstrationen erfolgreich umgesetzt werden.

Im Projekt kamen verschiedene Hardwareplattformen zum Einsatz, darunter ein neuromorpher Intel Loihi 2 Chip, ein NVIDIA Jetson Orin Development Kit sowie zwei ASUS NUCs der Generationen 12 und 14. Der Schwerpunkt lag auf dem Aufbau einer verteilten Rechenarchitektur sowie der Entwicklung robuster Kommunikationsschnittstellen zwischen den Systemkomponenten. Ein weiterer wesentlicher Arbeitsbereich war die Integration heterogener Sensorsysteme, bestehend aus Kamera, LiDAR, Radar und GPS, in ein Versuchsträgerfahrzeug. Zur Validierung der entwickelten Komponenten wurden mehrere Datenerhebungsfahrten auf einer definierten Testfläche durchgeführt.

2.3.4 Zielanpassung

Im Projektverlauf zeigte sich, dass das Resonate-and-Fire-(RF)-Verarbeitungsmodell sein volles Potenzial insbesondere im Zusammenspiel mit einer speziell optimierten neuromorphen Plattform entfalten kann. Das Modell wurde erfolgreich implementiert und im Gesamtsystem evaluiert. Von der ursprünglich geplanten Weiterentwicklung zum Balanced-Resonate-and-Fire-(B-RF)-Modell sowie der Integration von Sparse-Coding-Verfahren wurde aufgrund der erzielten Systemergebnisse abgesehen. Stattdessen wurden ergänzende neuromorphe Modelle integriert und wissenschaftlich untersucht, darunter die Implementierung klassischer Radarverarbeitungsverfahren sowie einer Objekterkennung auf dem neuromorphen Chip Intel Loihi 2.

Durch die vorgenannten Anpassungen konnte der Projektumfang gezielt konsolidiert und der technologische Reifegrad im Hinblick auf eine industrielle Anwendbarkeit deutlich erhöht werden.

Das Labeln des aufgenommenen Datensatzes war aus zeitlichen Gründen nicht möglich, da der Hardwareaufbau zur Messdatenaufnahme unerwartete Herausforderungen, insbesondere hinsichtlich der Zeitsynchronisation, aufwies.

2.4 Projektergebnisse und Auswirkungen

Die entwickelte Verarbeitungskette des Demonstrators verarbeitet Radarrohdaten bis zur Live-Visualisierung im Fahrzeug. Sie ist funktionsfähig und umfasst zwei parallele Verarbeitungsstränge:

- eine neuromorphe Verarbeitung auf Intel Loihi 2,

- eine vergleichende State-of-the-Art-Verarbeitung auf einem Jetson Orin.

Beide Streams werden im Fahrzeug in Echtzeit visualisiert. Damit wurde das angestrebte Technology Readiness Level TRL 7 erreicht – nämlich die Demonstration eines Prototypensystems in einer realen Einsatzumgebung, ausgehend von TRL 3.

Im Rahmen des Projekts wurden drei neuromorphe Verarbeitungsmodule entwickelt, die alternative Ansätze zu den drei klassischen Schritten der FMCW-Radarverarbeitung darstellen:

- Frequenzanalyse: Ersatz der herkömmlichen FFT-basierten Frequenzdetektion durch Resonance-and-Fire-Neuronen (RF), die spektrale Informationen direkt im Zeitbereich kodieren.
- Detektion: Ablösung der klassischen CFAR-Methode durch Compressed-Sensing- und Sparse-Coding-Verfahren, die insbesondere bei niedrigem Signal-Rausch-Verhältnis verbesserte Ergebnisse liefern.
- Objektklassifikation: Implementierung einer für neuromorphe Hardware optimierten DNN-Variante, die Objekte wie Fußgänger, Fahrräder und Fahrzeuge in Echtzeit erkennt.

Ergänzend wurde ein erweitertes Sensorenset im Versuchsfahrzeug integriert, welche die zeitlich synchronisierte Aufzeichnung von Radar- (Rohsignale von Infineon), Lidar-, Kamera- und hochgenaue GPS-Daten ermöglicht. Dieses Setup erlaubt umfassende multimodale Auswertungen und stellt eine wertvolle Basis für die weitere Entwicklung und Validierung neuromorpher Sensordatenverarbeitung dar.

2.4.1 AP 1 Algorithmenentwicklung

Neuromorphic Computing geht von einem neuartigen, event-basierten Datenverarbeitungsmodell aus. Um das volle Potential entfalten zu können ist eine gemeinsame Weiterentwicklung neuromorpher Hardware im Zusammenspiel mit neuartigen neuromorphen Algorithmen erforderlich. Die Anpassung erfolgt dabei domänenspezifisch in Bezug auf die Datenmodalität sowie in Bezug auf die spezifischen Anforderungen des Einsatzzwecks. Ziel ist die Weiterentwicklung und Evaluation ausgewählter neuromorpher Algorithmen (RF + SC) unter Berücksichtigung der Limitationen einer Ende-zu-Ende-Verarbeitung mit Intel Loihi 2.

Die wissenschaftlichen Arbeiten zur Entwicklung der neuromorphen Algorithmen erfolgte durch die Mercedes-Benz AG und der Universität zu Lübeck. TWT beteiligte sich durch technische Unterstützung hinsichtlich des Radars sowie des Hardwareaufbaus. Zudem lag die Verantwortung für die Aufnahme des Datensatzes bei der TWT.

2.4.2 UAP 1.1 Erstellen eines Datensatzes

Für das Training der neuartigen neuromorphen Algorithmen, sowie für die Evaluation, sollte ein Datensatz generiert und in drei Stufen zur Verfügung gestellt werden, als (i) Rohdaten mit im Fahrzeug montiertem Sensor bei Fahrt im Straßenverkehr, (ii) vorverarbeitete Daten für die Weiterverarbeitung in Intel Loihi und (iii) annotierte Daten mit Position, Größe, Geschwindigkeit und Objektklasse.

Aufgrund von Verzögerungen im Hardwareaufbau, welche nachfolgend näher erläutert werden, konnten weniger Daten aufgenommen werden als zuerst beabsichtigt.

Hardwareaufbau für die Datensatzaufnahme

Für die Aufnahme der Radardaten wurde ein FMCW-Radar der Firma Infineon verwendet (vgl. Abbildung 1). Eine Auflistung der wichtigsten Radarparameter sind in Tabelle 3 aufgelistet.



Abbildung 1: Infineon Radar Sensor, verbaut in einem PKW von Mercedes-Benz

Parameters	Values
Modulation	FMCW TDM MIMO
Chirp-Dauer	40.96 us
Chirp-Anzahl	1024
RF-Bandbreite	414 MHz
Max. eindeutige Distanz	240 m
Max. eindeutige Geschwindigkeit	± 8.5 km/h
Entfernungsauflösung	0.46 m
Geschwindigkeitsauflösung	0.07 m/s
Winkelauflösung	2°

Tabelle 3: Konfiguration und Leistungsparameter des Radarsensors

Zum Vergleich mit den aufgenommenen Radardaten sollen Lidar- sowie Kameradaten verwendet werden. Zu diesem Zweck wurden ein Lidar (Velodyne, Alpha Prime VLS-128) und eine Kamera (Teledyne, Blackfly S, BFS-U3-5025) im Demonstrationsfahrzeug verbaut, mit dem die Referenzdaten aufgenommen werden. Damit ein Vergleich der Daten sowie eine Übertragung von beispielsweise detektierten Objekten von einer Datenquelle auf die andere möglich ist, muss eine Zeitsynchronisation der Daten aus unterschiedlichen Datenquellen erfolgen. Diese Synchronisierung wird anhand eines RTK-GPS-Systems (ArduSimple, SimpleRTK3B Pro) vorgenommen. Das hochpräzise Pulse-per-Second-Signal (PPS) des GPS-Empfängers ermöglicht durch das PTPv2 (Precision Time Protocol) die Synchronisation der einzelnen Sensoren. Während der Lidar die Möglichkeit bietet die Daten direkt mit einem synchronisierten Zeitstempel zu versehen, müssen die Kameradaten bei Dateneingang auf dem Messrechner mit dem entsprechenden Zeitstempel markiert werden.

Zum Zweck der Datenaufnahme wurden die Sensoren an dem Demonstratorfahrzeug von Mercedes-Benz befestigt und der gesamte Hardware-Aufbau dort integriert.

Aufnahme des Datensatzes

Zur Algorithmenentwicklung sowie zur Algorithmenevaluation wurden unterschiedliche statische und dynamische Szenarien in einer dedizierten Testumgebung aufgenommen. An insgesamt drei Terminen wurden Datenaufnahmen auf dem Testgelände der Mercedes-Benz AG in Immendingen durchgeführt. Auf Grund von Stabilitätsproblemen der Sensoren, insbesondere des GPS-Systems und folglich Synchronisationsproblemen der einzelnen Sensoren, konnten die Daten der ersten beiden Aufnahmetermine nicht weiterverwendet werden. Nach der Implementierung eines Trigger-Skripts, welches die Sensoren startet und die Datenaufnahme aller Sensoren zentral triggert, sowie einer Anpassung des GPS-Setups, konnten am letzten Termin die folgenden Szenarien aufgenommen werden:

Kalibrierungstests

Zu Beginn wurden gezielte Szenarien zur Kalibrierung des Radarsensors durchgeführt. Diese umfassen unter anderem statische Messungen eines Corner-Reflektors bei 0° Einfallswinkel in unterschiedlichen Entfernungen sowie dynamische Fahrten auf einer Kurvenstrecke mit Bäumen, Leitplanken und metallischen Zaunelementen im Sichtfeld des Sensors. Ergänzend erfolgte eine Messung, bei der sich das Versuchsfahrzeug mit einer Geschwindigkeit von 6 km/h geradlinig auf den in Fahrzeuglängsachse positionierten Corner-Reflektor zubewegt.

Dynamische Szenarien mit stillstehendem Versuchsfahrzeug

Zur Erweiterung des Datensatzes wurden verschiedene dynamische Szenarien aufgenommen, in denen sich unterschiedliche Zielobjekte bewegten, während das Versuchsfahrzeug in statischer Position verblieb. Ziel dieser Aufnahmen war die Erfassung realistischer Bewegungssituationen zur Validierung und zum Vergleich neuromorpher und konventioneller Verarbeitungsansätze. Die aufgezeichneten Szenarien umfassten:

- Die Slalomfahrt eines PKWs um zwei Corner-Reflektoren sowohl in Richtung des Versuchsfahrzeugs als auch in entgegengesetzter Fahrtrichtung,
- die Durchfahrt eines PKWs durch die Mitte zweier Corner-Reflektoren sowohl in Richtung des Versuchsfahrzeugs als auch in entgegengesetzter Fahrtrichtung,
- die Fahrt eines PKWs in einer „Acht“-förmigen Trajektorie um zwei Corner-Reflektoren rechtwinklig zur Ausrichtung des Versuchsfahrzeugs,
- das zeitweilige Verdecken eines Fußgängers hinter einem PKW bei Annäherung von links oder rechts,
- das Hervortreten eines Fußgängers hinter einem PKW in das Sichtfeld des Radarsensors,
- die Bewegung einer Person quer zur Ausrichtung des Versuchsfahrzeugs mit zwei in der Nähe positionierten Corner-Reflektoren,
- sowie die Bewegung einer Person entlang der Fahrzeuglängsachse, sowohl in Richtung des Versuchsfahrzeugs als auch davon weg.

Dynamische Szenarien mit Fahrzeugbewegung

In diesen Szenarien bewegt sich auch das Versuchsfahrzeug aktiv. Ziel dieser Aufnahmen war die Analyse komplexer Bewegungskonstellationen zwischen Fahrzeugen und Fußgängern. Die durchgeführten Szenarien umfassten:

- die Annäherung des Versuchsfahrzeugs an einen stehenden PKW, während sich eine Person vor bzw. hinter dem Fahrzeug befindet und die Fahrbahn zu überqueren beabsichtigt,
- die Annäherung des Versuchsfahrzeugs an einen PKW, gefolgt von einem Wendemanöver und einer erneuten Annäherung,
- sowie eine schlangenförmige Annäherung des Versuchsfahrzeugs an einen stehenden PKW.

Diese Aufnahmen ermöglichen eine detaillierte Analyse der Radarsignaturen unter variierenden Bewegungs-, Abschattungs- und Separationssituationen und bilden eine wesentliche Grundlage für die Validierung der entwickelten Signalverarbeitungsalgorithmen.

Datenaufnahmen im offenen Straßenverkehr konnten aus Zeitgründen nicht durchgeführt werden.

Durchführung der Datensatzaufbereitung für den Einsatz in Intel Loihi 2 und der State-of-the-Art Umsetzung

Ursprünglich war vorgesehen, Lidar- und Kameradaten als Referenz für die Objekterkennung zu nutzen. Aufgrund des Mangels an zuverlässigen, universell einsetzbaren Algorithmen zur Objekterkennung in Lidardaten und der Tatsache, dass vorhandene Ansätze meist modellabhängig trainiert sind, wurde aus Kapazitätsgründen auf die Entwicklung einer entsprechenden semi-automatischen Pipeline verzichtet. Stattdessen wurde eine Zusammenarbeit mit dem VERANO-Projekt bei der Datenerfassung angestrebt. Dadurch standen zwei mobile GPS-Empfänger (GeneSys ADMA-Slim) zur Verfügung, mit denen die Bewegung dynamischer Objekte (z. B. Fußgänger) präzise erfasst werden konnte.

Da die Datenerfassung in einer kontrollierten Umgebung stattfinden sollte und sämtliche Objekte in der Szene mit einem GPS-Gerät ausgestattet waren, war keine Objekterkennung auf Basis der Lidardaten erforderlich. Stattdessen wurde ein Framework implementiert, welches die GPS-Daten aller Objekte einliest, in das Radarkoordinatensystem des Versuchsfahrzeugs transformiert und daraus die Radar-Ground-Truth-Informationen – radiale Entfernung, Relativgeschwindigkeit, Winkel sowie Objektklasse – berechnet.

Aufgrund der beschriebenen Synchronisationsprobleme konnten die Daten der ersten beiden Aufnahmetermine nicht verwendet werden. Da die VERANO-Projektpartner am dritten Aufnahmetag nicht anwesend waren, standen die mobilen GPS-Empfänger an diesem Tag nicht zur Verfügung. Folglich konnten keine GPS-Daten der Objekte in den Szenarien zur Erstellung von Labels genutzt werden, sodass am Ende kein Labeling der Daten umgesetzt werden konnte.

Um die Verwertung von Forschungsergebnissen für die Entwicklung in Serienfahrzeugen zu ermöglichen, muss jeder Verarbeitungsschritt, ausgehend von Sensorik und Vorverarbeitung, über die Haupt-Sensordatenverarbeitung, bis zur Ausgabe des verarbeiteten und interpretierten Signals nachweislich funktionieren. Die Verarbeitung der einzelnen Module müssen in der System-sicht zusammenpassen und harmonisch ineinandergreifen. Um dies nachzuweisen, wurde die gesamte Ende-zu-Ende-Verarbeitungskette in einem Demonstratorfahrzeug dargestellt, ausgehend vom Frequency-Modulated-Continuous-Wave-Sensor (FMCW) mit Vorverarbeitungseinheit auf einem NVIDIA Jetson Orin Development Kit, über die Sensordatenverarbeitung mit neuartigen neuromorphen Algorithmen unter Verwendung von Intel Loihi 2, bis zur Echtzeit-Darstellung der Verarbeitung auf einer Anzeigeeinheit in der Fahrkabine des Fahrzeugs. Durch Betrachtung der gesamten Verarbeitungskette als ein System und dementsprechend durchgeführten Tests wurden Hindernisse, Flaschenhälse und weitere Verbesserungsvorschläge herausgearbeitet.

Die Ende-zu-Ende Verarbeitungskette wurde zunächst als Tischaufbau konzipiert, da die Einrichtung eines Fernzugriffs per Internet sowie die Fehlersuche auf diese Weise unkomplizierter waren und dieser Tischaufbau auch früher zur Verfügung stand als die im Fahrzeug verbaute Ende-zu-Ende Verarbeitungskette.

2.4.3 UAP 2.1 Hardwareaufbau der Ende-zu-Ende-Verarbeitungskette

Evaluierung des Loihi 2 Development Kits sowie Schnittstellen- und Anforderungsdefinition für die Ende-zu-Ende-Verarbeitungskette

Loihi 2 ist ein neuromorpher Chip von Intel, der für die effiziente Echtzeitverarbeitung von Spiking Neural Networks (SNNs) mit geringem Stromverbrauch entwickelt wurde und in der Robotik, in autonomen Systemen und intelligenten Infrastrukturen Anwendung findet. Der

neuromorphe Chip verfügt über ein vollständig programmierbares Neuronenmodell, das sowohl graduierte als auch binäre Spikes unterstützt und somit eine Vielzahl von Rechenaufgaben ermöglicht. Graduierte Spikes, die eine 32-Bit-Nutzlast tragen, bieten eine präzisere Modulation der synaptischen Gewichte als binäre Spikes. Das Verhalten der Neuronen wird durch Mikrocode-Befehle gesteuert, die Operationen wie bitweise Logik, Arithmetik, Verzweigungen und Spike-Generierung ermöglichen.

Das Development Kit des Chips ermöglicht mit seinem flexiblen Inter-Chip-Protokoll und Standardschnittstellen wie Ethernet, GPIO und SPI die Systemintegration. Intel bietet außerdem Lava an, ein Open-Source-Software-Framework für die Entwicklung neuromorpher Anwendungen auf Loihi 2. Lava unterstützt das Training, die Integration mit Frameworks wie TensorFlow und PyTorch und bietet Python-basierte Bibliotheken.

Im Rahmen des Projekts wurde das Loihi 2 Development Kit in Betrieb genommen, welches einen Host-PC benötigt, auf dem die Softwareumgebung Lava ausgeführt wird. Lava übernimmt hierbei die zentrale Koordination der Kommunikation und Orchestrierung des Loihi 2-Chips. Ein besonderer technischer Meilenstein war die Inbetriebnahme der 10-Gigabit-Ethernet-Schnittstelle (SFP+ Anschluss), die in dieser Form zuvor noch nie in Deutschland verwendet worden war. In enger Zusammenarbeit mit Partnern von Intel konnte die Schnittstelle erfolgreich in Betrieb genommen und in das Gesamtsystem integriert werden.

Ursprünglich war vorgesehen, die 10-Gigabit-Schnittstelle als Eingangsverbindung sowie eine 1-Gigabit-Ethernet-Schnittstelle für die benötigte Ausgangs- und Host-Kommunikation zu nutzen. Diese Architektur ließ sich jedoch in der Praxis nicht realisieren. Stattdessen werden die Ausgabedaten nun zurück an den Host-PC übertragen und von dort an das Visualisierungssystem weitergeleitet.

Die entwickelte Host-Software wurde zunächst für AMD64-Architekturen konzipiert. Im Projektverlauf wurden die benötigten Anpassungen und Erweiterungen, um auch eine Ausführung auf Jetson-Systemen mit ARM-Architektur zu ermöglichen, durchgeführt.

Entwicklung des Hardwareaufbaus der ganzheitlichen Ende-zu-Ende-Verarbeitungskette für einen Tischaufbau unter Berücksichtigung der Kommunikation zwischen allen Verarbeitungseinheiten sowie der Stromversorgung und der Möglichkeit für einen Fernzugriff per Internet

Die Ende-zu-Ende-Verarbeitungskette beginnt bei dem FMCW-Radarsensor von Infineon. Die Entscheidung für diesen Sensor wurde getroffen, um Synergien mit dem öffentlich geförderten Forschungsprojekt VERANO, bei welchem die Mercedes-Benz AG ebenfalls beteiligt, zu schaffen. Die Radar-Rohdaten werden am Radarsensor über vier Kanäle serialisiert und über Koaxial-Kabel zur weiteren Verarbeitung an einen NVIDIA Jetson AGX Orin Development Kit geschickt. Dort werden die Daten mithilfe des von Infineon zur Verfügung gestellten Boards deserialisiert und über das CSI-2 (Camera Serial Interface 2) des Jetsons direkt in dessen RAM gespeichert. Bei CSI-2 handelt es sich um ein weit verbreitetes Hochgeschwindigkeitsprotokoll, um unbewegte sowie bewegte Bilder von einem Sensor, z.B. einer Kamera auf einen Prozessor zu übertragen. Das Betriebssystem auf dem Jetson ist ein Ubuntu 20.04, zusätzlich wurde die JetPack 5.1.2 Version installiert.

Nach einer Vorverarbeitung werden die Daten zur parallelen Evaluation mit neuromorphen Algorithmen und Stand-der-Technik weitergegeben. Hierbei stellt der NVIDIA Jetson den Host-PC für den Intel Loihi 2 Chip dar. Die Rohdaten werden hier über die 10-Gigabit-Ethernet Schnittstelle übertragen, um eine Echtzeitfähigkeit zu gewährleisten. Nach der Verarbeitung auf Intel Loihi 2 werden die Daten über die bidirektionale Verbindung zurück an das Host-System geschickt und von dort an die Visualisierung weitergeleitet.

Für die Auswertung nach aktuellem Stand-der-Technik war ursprünglich das NXP S32R274RRU-EVB Microcontroller Kit vorgesehen. Nach dem Erwerb des Microcontroller Kits wurde jedoch festgestellt, dass die Schnittstellen des Boards nicht wie geplant angesteuert werden konnten. Da

auch der technische Hersteller-Support nicht weiterhelfen konnte, wurde entschieden die Radardatenauswertung nach Stand der Technik auf der CPU/GPU des NVIDIA Jetson Orin durchzuführen.

Sowohl der neuromorphe Prozessor Intel Loihi 2 als auch der Jetson für die Auswertung der Radardaten nach Stand der Technik wurden per Ethernet mit einem ASUS NUC 12 PC verbunden, auf welchem die Visualisierung der Daten durchgeführt wurde. Dieser NUC wurde daher mit einem Bildschirm verbunden, um die Visualisierungssoftware anzeigen zu können. Das Betriebssystem, welches auf dem NUC installiert wurde, ist ebenfalls Ubuntu 20.04.

Da der Radarsensor auf einem Tischaufbau keine sinnvollen Daten liefern würde, wurde entschieden die Kosten für die Anschaffung beim Tischaufbau zu ersparen und stattdessen die anschließende Verarbeitungskette durch simulierte bzw. aufgenommene Daten zu testen.

Zur Kommunikation der einzelnen Komponenten untereinander werden Protokolle des TCP/IP-Stacks verwendet. Hierzu befinden sich alle Komponenten in demselben privaten Netzwerk und haben festgelegte statische IP-Adressen. Sowohl beim Tischaufbau der TWT GmbH als auch im Demonstrationsfahrzeug ist ein Remote-Zugriff per SSH auf den NUC, welcher für die Visualisierung zuständig ist, möglich. Über diesen NUC kann sowohl der Intel Loihi 2 als auch der Jetson angesteuert werden. Eine Skizze des Hardwareaufbaus inklusive des Referenzsystems zur Datenaufnahme ist in Abbildung 2 dargestellt.

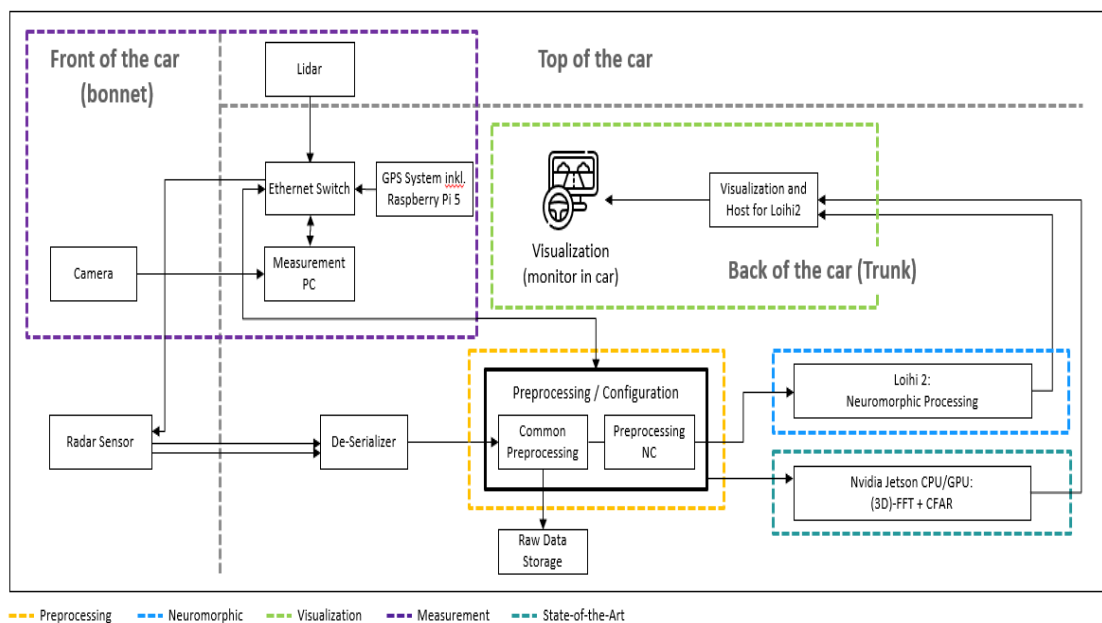


Abbildung 2: Hardwareaufbau der Ende-zu-Ende-Verarbeitungskette inkl. Referenzsystem

Anpassung des Hardwareaufbaus hinsichtlich der Anforderungen an die Integration in das Versuchsfahrzeug

Der Hardwareaufbau der Ende-zu-Ende-Verarbeitungskette wurde von Anfang an bewusst mit Hinblick auf die Integration im Fahrzeug geplant. So wurde beispielsweise der Switch so gewählt, dass er mit einer 12V-Spannungsversorgung betrieben werden kann.

Allerdings musste aufgrund der Versorgungsspannung der NUCs von 19V eine 230V-Stromquelle dem Demonstrationsaufbau hinzugefügt werden, da die benötigte Eingangsspannung durch die 12V nicht ausreichend abgedeckt werden konnte.

Einbau eines zweiten Hardwareaufbaus in ein Versuchsfahrzeug, inklusive Konstruktionsaufwände für die Sensorintegration

Um die Anpassbarkeit sowie die einfache Zugänglichkeit des Hardwareaufbaus während der Projektarbeit gewährleisten zu können, wurden alle Komponenten mit Ausnahme der Sensoren und des Visualisierungsbildschirms im Kofferraum integriert (siehe Abbildung 3).

Die Antenne des GPS-Empfängers und der Lidar wurden auf dem Dach des Demonstratorfahrzeugs angebracht. Die Kamera hingegen wurde oben-mittig hinter der Windschutzscheibe montiert.

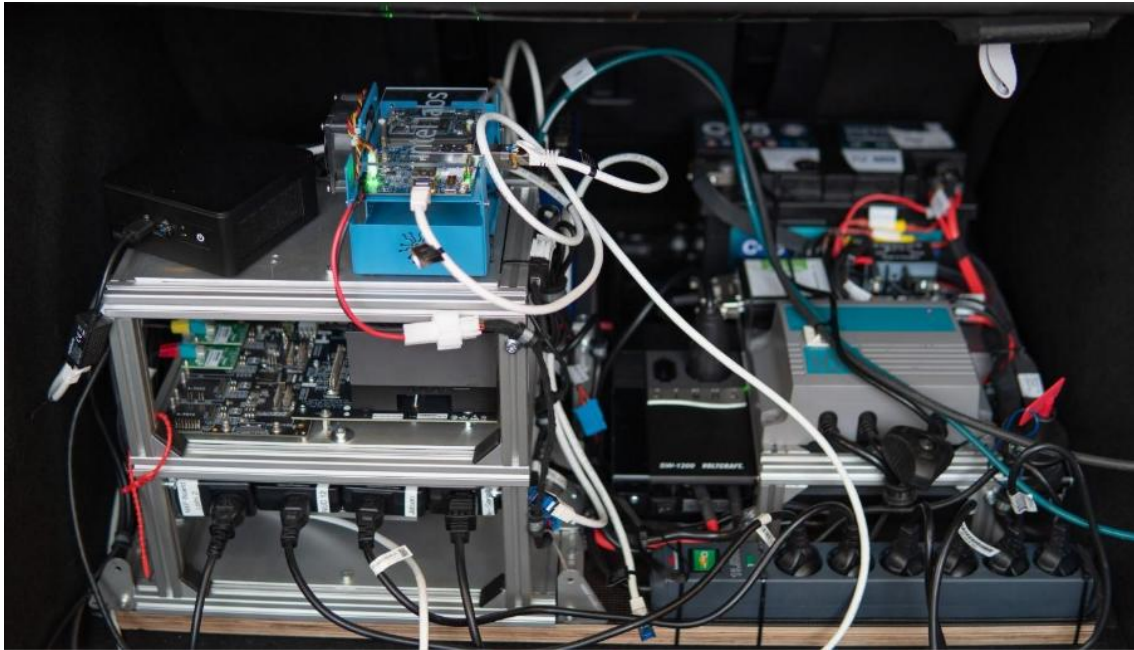


Abbildung 3: Setup für Echtzeitverarbeitung der Radardaten mit neuromorphem Prozessor Intel Loihi 2 im Kofferraum des Fahrzeugs

2.4.4 UAP 2.2 Vorverarbeitung der Radarsensordaten zur Verarbeitung mit Intel Loihi 2

Die von der Hardware (Radar, Loihi 2) vorgegebenen Rahmenbedingungen erfordern eine Vielzahl unterschiedlicher Expertisen. Für die Initialisierung der Modelle auf Loihi 2, sowie die Einspeisung und das Auslesen der rohen oder vorverarbeiteten Messdaten wird eine tiefe Kenntnis der Python-API von Loihi 2 und von Signalverarbeitungsstrategien benötigt. Das Auslesen der rohen Messdaten via Direkt-Memory-Access hingegen bedarf Performanz optimierter, hardwarenaher Programmierung.

Hinzu kommt, dass die Modelle unterschiedliche Ansprüche an die Vorverarbeitung der Modellinputs stellen. Einige arbeiten mit der Ausgabe einer Fourier-Transformation oder der CFAR (Constant False Alarm Rate), andere direkt auf den rohen Messungen des Radars. Um die Vergleichbarkeit mit der State-Of-the-Art Verarbeitung zu gewährleisten, sollen die Komponenten zudem möglichst auf demselben Eingabe-Stream arbeiten.

Um die im Projekt vorhandenen Kenntnisse optimal zu nutzen und die notwendige Flexibilität zu gewährleisten wurde eine Architektur aus mehreren separaten Prozessen erarbeitet, die über eine geteilte Bibliothek kommunizieren. Dies bringt einige technische und soziale Vorteile bei der Implementierung mit sich:

1. Die Komponenten können autonom von dem jeweiligen Experten programmiert und weiterentwickelt werden.
2. Die Ein- und Ausgaben der einzelnen Komponenten können leicht auf verschiedene Quellen und Konsumenten umgeleitet werden.
3. Nicht alle Komponenten müssen auf derselben Hardware laufen. Die Echtzeit-Visualisierung finden z.B. auf einem separaten PC statt.
4. Für die Orchestrierung und Überwachung der Prozesse können etablierte Tools des OS verwendet werden.

Da die meisten Prozesse mit Python implementiert wurden, konnte die Kommunikation zwischen den Komponenten in ein gemeinsames Package ausgelagert werden. In diesem Package wurden Container für alle Ein- und Ausgaben der Modelle, sowie deren (De-)Serialisierung implementiert. Ebenso wurden Funktionen zur Übertragung via TCP/IP und Unix-Pipes bereitgestellt. Diese strikt definierte Schnittstelle erleichtert eine Umleitung der Datenströme auf unterschiedliche Komponenten und dient als Kommunikationsgrundlage zwischen den Partnern.

Im Laufe des Projektes wurde auch ein Message Broker wie z. B. ROS1/2 in Erwägung gezogen. Jedoch erschien die Umstellung auf ein neues OS mit spezifischen Einschränkungen bei der Integration mit spezieller Software, wie sie vom Loihi 2 benötigt wird, zu riskant.

Implementierung einer echtzeitfähigen Übertragung der Rohdaten von Sensor zur Rechen-einheiten der Stand-der-Technik

Der Sensor stellt die Messdaten über ein Kernelmodul via Shared Memory bereit. Eine Implementierung in C++ zum Auslesen dieser Rohdaten wurde von Infineon bereitgestellt und für NAOMI4Radar angepasst. Aufgrund der hohen Datenrate (~40 MB pro Frame) kam Python für diese Aufgabe nicht in Frage. Das Programm stellt die strukturierten Daten über stdout bereit; die Serialisierung erfolgt mit Flatbuffers. Dies erlaubt die deklarative Definition der Datenstruktur mit der Pythonbibliothek zu teilen. Der Ausgabestrom kann mit den Mitteln des OS (Ubuntu/Linux) auf mehrere Konsumenten verteilt werden; die Bandbreite stellt Unix-Pipes vor keine besondere Herausforderung. Dieser Strom aus Rohdaten wird dann vom Prozess für die Vorverarbeitung konsumiert. Die so vorverarbeiteten Daten wurden dann an die Pipeline für die Auswertung nach Stand-der-Technik mit Hilfe des TCP-Protokolls weiter gegeben.

Implementierung einer echtzeitfähigen Übertragung der Rohdaten von Sensor zu Intel Loihi 2

Die Umsetzung für die Übertragung der Radardaten an die neuromorphe Auswertung ist nahezu äquivalent zu den Schritten für die Übertragung zur Stand-der-Technik-Auswertung. Die vorverarbeiteten Radardaten werden nicht nur an den Auswerteprozess nach Stand-der-Technik auf dem Jetson weitergeleitet, sondern auch an den Host-Prozess des Loihi 2. Auch hier folgt die Kommunikation mittels TCP-Protokolls.

Implementierung notwendiger Vorverarbeitungsschritte der Radarsensordaten zur Echtzeit-Verarbeitung mit Intel Loihi 2 unter Berücksichtigung dessen Rahmenbedingungen wie beispielsweise der I/O-Geschwindigkeit, dem I/O-Protokoll, sowie dem geringstmöglichen durchschnittlichen Verarbeitungszeitschritt

Die Vorverarbeitung muss die Rohdaten des Sensors aus der Unix-Pipe lesen und je nach angestrebtem neuromorphen Algorithmus entweder bestimmte Antennen oder Chirps aus dem Rohdatenstream filtern oder direkt die Range-Doppler bzw. Range-Angle-Map berechnen.

Um je nach Bedarf auf verschiedene (Vor)verarbeitungsschritte zurückgreifen zu können, wurde ein Python-Package „naomi4radar_datahandler“ zur Radardatenverarbeitung implementiert. Dieses unterstützt drei verschiedene Inputmodi: Simulation, Messung sowie Unix-Pipe. Darüber hinaus wurden alle wichtigen Signalverarbeitungsschritte der Radardatenverarbeitung implementiert. Hierzu gehören verschiedene Rohdaten-Umformungen, um die Azimuth- und Elevations-Informationen der Empfangsantenne zu extrahieren, die 2D- und 3D-FFT-Berechnung mit der Möglichkeit eine Fensterung durchzuführen und die OS-CFAR Implementierung.

2.4.5 UAP 2.3 Implementierung einer Ende-zu-Ende-Verarbeitung von Radardaten nach Stand der Technik

Implementierung der Signalverarbeitungskette von FMCW-Radardaten zur Lokalisierung, Verfolgung und Klassifizierung von Objekten nach Stand-der-Technik

Die Implementierung der Stand-der-Technik wurde auf dem Nvidia Jetson Orin Development Kit umgesetzt. Hierzu wurde das Python-Package „naomi4radar_datahandler“ um die Möglichkeit erweitert die Berechnungsschritte der FFT auf die GPU des Jetson auszulagern. Eine detaillierte Analyse dieser Erweiterung wird im nächsten Abschnitt aufgeführt.

Integration der Stand-der-Technik-Auswertung der Radardaten in den Hardwareaufbau des Demonstrators

Auf Grund der Tatsache, dass die Stand-der-Technik Auswertung auf dem Jetson durchgeführt wird, mussten keine größeren Integrationstätigkeiten durchgeführt werden, welche über die Testung der resultierenden Skripte für die Demonstration hinausgehen. Aus diesem Grund wurden zur Bewertung der Rechenleistung des Nvidia Jetson Convolution und FFT sowohl auf der CPU als auch auf der GPU ausgeführt. Dafür wurden zufällig generierte Eingangsdaten unterschiedlicher Größe verwendet, konkret Matrizen mit den Dimensionen (100×100) , (200×200) , (2000×2000) , (5000×5000) sowie (20000×20000) . Die Berechnung auf der GPU erfolgten mithilfe von CUDA und des Python-Pakets Cupy, um die parallelen Recheneinheiten optimal zu nutzen. Durch den Vergleich der Ausführungszeiten auf CPU und GPU lässt sich aufzeigen, wie sich die Skalierung des Problems auf die Performanz auswirkt und ab welcher Problemgröße die GPU einen deutlichen Vorteil gegenüber der CPU bietet.

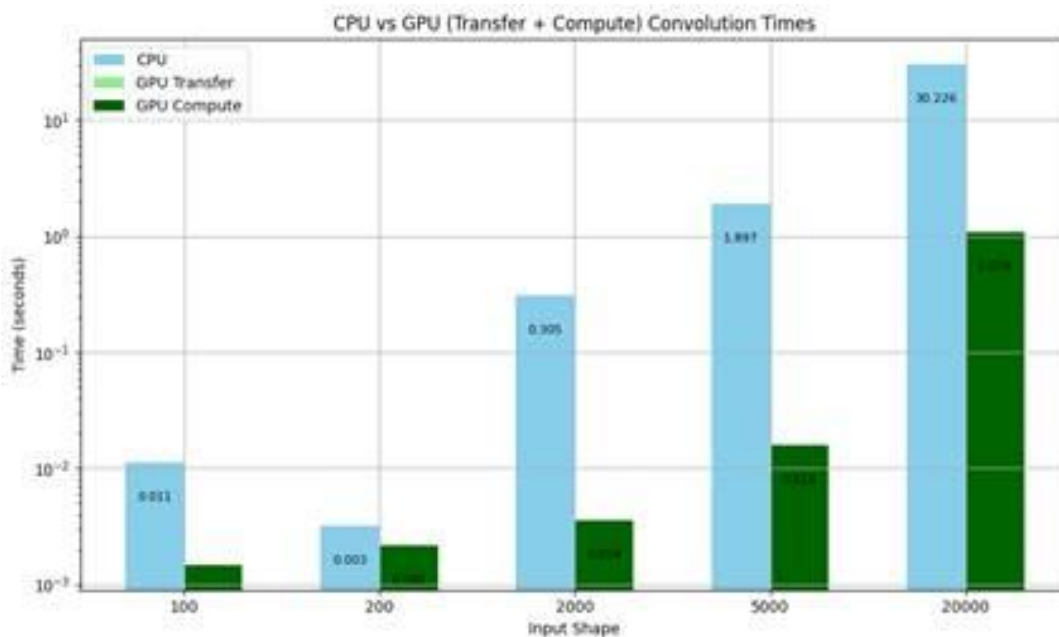


Abbildung 4: Rechenleistung Nvidia Jetson CPU vs. GPU für FFT- und Convolution-Berechnungen

In der obigen Abbildung (logarithmische Zeitskala) sind die gemessenen Ausführungszeiten für die CPU-Berechnung, den Datentransfer zur GPU sowie die GPU-Berechnung dargestellt. Auffällig ist, dass die für den Transfer benötigte Zeit auf dem Nvidia Jetson vernachlässigbar klein ist. Dies liegt daran, dass CPU und GPU über ein gemeinsames Arbeitsspeicher-Subsystem (Shared Memory) verfügen. Im Gegensatz zu klassischen Desktop-GPUs entfällt somit die aufwendige Kopie großer Datenmengen. Dadurch lohnt sich die Nutzung der GPU bereits bei deutlich kleineren Datengrößen. Während auf typischen Systemen oft erst bei großen Matrizen ein signifikanter

Performance-Gewinn gegenüber der CPU erreicht wird, zeigt der Jetson durch die minimierte Transferzeit schon bei mittleren Dimensionen einen klaren Vorteil

2.4.6 UAP 2.4 Implementierung der neuromorphen Verarbeitungsmodelle zur Ausführung auf Intel Loihi 2

Die Implementierung der neuromorphen Verarbeitungsmodelle zur Ausführung auf Intel Loihi 2 Chip wurde durch die Mercedes-Benz AG sowie der Universität zu Lübeck umgesetzt. Diese beinhaltet neuronale Verarbeitungsmodelle zur Fourier-Transformation mit Resonate-and-Fire-Neuronen (F-RF) für die Radardatenverarbeitung sowie zu Sparse Coding basierend auf dem Locally Competitive Algorithm (LCA).

2.4.7 UAP 2.5 Echtzeit-Visualisierung der Radardatenverarbeitung

Entwicklung der Echtzeit-Visualisierung der Radardatenverarbeitung, welche die Verarbeitungsgüte des neuromorphen Ansatzes mit dem Stand der Technik vergleicht

Zur Echtzeit-Visualisierung der Radardatenverarbeitung wurde ein Programm in Python unter Verwendung der Bibliothek Tkinter (vis4naomi) sowie ein Package zur einheitlichen Kommunikation der unterschiedlichen Systeme mit dem Visualisierungsprogramm (com4naomi) geschrieben. Die Visualisierung ermöglicht die parallele Darstellung der Auswertung der Radardaten nach Stand der Technik und mit neuromorphen Methoden. Sie umfasst die Darstellung der Fourier-Transformation der Radardaten in einer Dimension (Abstand) und in zwei Dimensionen (wahlweise Abstand-Azimuthwinkel-Darstellung oder Abstand-Geschwindigkeit-Darstellung). Zudem können 2 Video-Streams verglichen werden. Diese findet bei der Darstellung der Objekterkennung mit verschiedenen TinyYolo- (You only look once) Modellen Anwendung.

Die Echtzeit-Visualisierung empfängt direkt die Ausgabe der Modelle und der State-of-the-Art Algorithmen via TCP/IP. Ausgeführt wird der Visualisierungs-Prozess auf einem separaten PC, um die Performance der Signalverarbeitung auf dem Jetson nicht zu verfälschen.

Sowohl das Programm für die Visualisierung als auch das Package zur Kommunikation zwischen den Systemen sind objektorientiert aufgebaut.

Anpassung der Visualisierung an die Anforderungen zur Integration in das Versuchsfahrzeug

In der Entwicklung der Visualisierung wurde die Integration in das Versuchsfahrzeug von Beginn an berücksichtigt, sodass die Darstellung nicht angepasst werden musste. Bei der dahinterliegenden Logik war ebenfalls keine Anpassung notwendig.

2.4.8 UAP 2.6 Technische Evaluation

Zur Ermittlung des Energieverbrauchs des Demonstrators wurde das Messverfahren in zwei Schritten durchgeführt. Zunächst wurde der Stromverbrauch des Nvidia Jetson im Idle-Zustand über einen Zeitraum von 30 Minuten aufgezeichnet, um einen stabilen Referenzwert zu erhalten. Anschließend erfolgte die Messung während der Ausführung des zu evaluierenden Prozesses. Der tatsächliche Verbrauch des Prozesses ergibt sich dabei aus der Differenz zwischen der unter Last gemessenen Leistung und dem zuvor bestimmten Idle-Durchschnitt. Die Messungen wurden sowohl mit den Nvidia-Tool tegrastats als auch mit der Power GUI durchgeführt, wobei zwischen Instantaneous Power (aktuelle Momentanwerte) und Average Power (gemittelte Werte über den Messzeitraum) unterschieden werden muss. Zur Veranschaulichung wird im Folgenden ein Beispielergebnis in Tabellenform dargestellt.

	Gesamt	Idle	Verbrauch vom Prozess
CPU In	1841 mW	800 mW	1041 mW
CPU Avg	1174 mW	800 mW	374 mW
GPU In	3418 mW	2958 mW	460 mW
GPU Avg	3268 mW	3093 mW	175 mW

Tabelle 4: Energieverbrauchsmessungen State of the Art

Um den Verbrauch des Demonstrators beispielhaft zu ermitteln, wurde die Leistungsaufnahme (die vier abgebildeten Kennwerte CPU In, CPU Avg, GPU In, GPU Avg) über einen Zeitraum gemessen, welcher zum Ausführen des Demonstrators benötigt wird. Die Messabstände waren 1000 ms und die Messwerte wurden dann über den Zeitraum gemittelt. Wie beschrieben werden die Idle Messwerte vom Gesamt abgezogen, um den Verbrauch vom Demonstrator abzuschätzen.

Der Vergleich zwischen den neuromophen Algorithmen und der State-of-the-Art Implementierung erfolgte durch die Entwickelnden der neuromophen Algorithmen.

2.4.9 AP 3 Projektkoordination und Ergebnisverbreitung

Als Konsortialführung übernahm die Mercedes-Benz AG die Gesamtkoordination des Projektes. TWT war verantwortlich für die Leitung des Arbeitspakets „Demonstrator“ und leistete einen wesentlichen Beitrag zur Erstellung der technischen Dokumentation sowie zur Berichterstattung. Durch die koordinierte Zusammenarbeit im Konsortium konnten die Projektziele innerhalb des vorgegebenen Zeitrahmens vollständig erreicht und die Ergebnisse erfolgreich in Forschung und Industrie kommuniziert werden.

2.4.10 UAP 3.1 Projektmanagement

Im Rahmen des Verbundprojekts übernahm TWT die Verantwortung für das Management des Arbeitspakets AP2 „Demonstrator“. In diesem Zusammenhang war TWT für die technische Gesamtkoordination des Arbeitspakets sowie für die Erstellung der zugehörigen Zwischen- und Endberichte für den Projektträger zuständig.

Zu den Aufgaben von TWT gehörten die Planung, Abstimmung und Überwachung sämtlicher technischer Aktivitäten innerhalb des Arbeitspakets. Dies umfasste die Koordination der Schnittstellen zwischen den beteiligten Partnern, die Sicherstellung der technischen Kohärenz der Arbeitsergebnisse sowie die Begleitung der Umsetzung des Demonstrators gemäß den definierten Projektzielen und Meilensteinen.

Darüber hinaus unterstützte TWT die Konsortialführung bei der technischen Dokumentation und der Berichterstattung auf Gesamtprojektebene. Durch eine enge Abstimmung mit den beteiligten Partnern konnte ein effizienter Informationsfluss gewährleistet und ein wesentlicher Beitrag zum fristgerechten Erreichen der Projektziele geleistet werden.

2.5 UAP 3.2 Publikation wissenschaftlicher Ergebnisse

Im Rahmen des Vorhabens wurde ein wissenschaftlicher Beitrag auf der ELIV Conference 2025 präsentiert. Die ELIV (European Laboratory for Intelligent Vehicles) ist ein jährlich stattfindendes Fachforum, das führende Forschungsarbeiten und industrielle Entwicklungen in den Bereichen autonomes Fahren, Sensorik und Künstliche Intelligenz zusammenführt und somit einen intensiven Austausch zwischen Wissenschaft und Industrie ermöglicht.

Der Beitrag mit dem Titel „NAOMI4Radar: Advancing Autonomous Driving with High-Efficiency Neuromorphic Radar“ wurde am 15. Oktober 2025 in der Session „Automotive AI – From Sensor

to Validation“ vorgestellt. Das Paper beschreibt den im Projektverlauf entwickelten Demonstrator sowie einen Vergleich zwischen neuromorphen und konventionellen Verfahren zur Auswertung von Radardaten. Darüber hinaus werden die Potenziale und Grenzen neuromorpher Berechnungsansätze für die Radar-Signalverarbeitung evaluiert.

Die Präsentation erfolgte durch Geoffrey Kasenbacher (PhD Student, Group Research, Future Technologies, Mercedes-Benz AG, Böblingen) und Franziska Ott (Principal ADAS, Technological Competence Center, TWT GmbH Science & Innovation, Stuttgart). Das Fachpublikum zeigte großes Interesse an den vorgestellten Ergebnissen, insbesondere an der praktischen Umsetzung und dem innovativen Ansatz der neuromorphen Radarverarbeitung.

Neben dieser Publikation sind seitens TWT keine weiteren Veröffentlichungen zu NAOMI4Radar geplant.

2.6 Wichtigste Positionen des zahlenmäßigen Nachweises

Der zahlenmäßige Nachweis erfolgt in einem gesonderten Dokument.

2.7 Verwertbarkeit der Ergebnisse

Der Einsatz von Neuromorphic Computing (NC) ermöglicht eine präzisere Erfassung und Verarbeitung von Radarsignalen und trägt damit wesentlich zur Verbesserung der Umfeldwahrnehmung für automatisierte Fahrfunktionen bei. Gleichzeitig führt die auf ereignisbasierten Prozessen beruhende Architektur zu einer deutlichen Reduzierung des Energiebedarfs bei der Signalverarbeitung.

Diese Technologie besitzt das Potenzial, die Leistungsfähigkeit und Energieeffizienz zukünftiger Fahrzeugsysteme maßgeblich zu steigern und damit die Wettbewerbsfähigkeit des Standorts Deutschland im Bereich sicherer und effizienter Fahrzeugtechnologien weiter zu stärken.

Als Innovationspartner nahezu aller deutschen Automotive-OEMs, sowie renommierter Tier-1-Zulieferer beabsichtigt TWT, die in NAOMI4Radar angestrebten Ergebnisse in Form verschiedener Industrieprojekte in die Anwendung zu bringen. Es war geplant, dass die im Rahmen des Forschungsvorhabens NAOMI4Radar angestrebten Projektergebnisse einen erheblichen Beitrag zur Weiterentwicklung der TWT- Kernkompetenzen, im Bereich der Radardatenauswertung generell und in der Anwendung von Neuromorphic Computing auf dedizierter Hardware im Speziellen, leisten sollen. Dieser Kompetenzaufbau ist im Rahmen des Projekts erfolgt. Bereits während der Projektlaufzeit wurde das Projekt in zahlreichen Akquiseterminen vorgestellt und hat so dazu beigetragen die Kompetenz der TWT im Bereich ADAS zu unterstreichen und dadurch Projekte zu gewinnen. TWT plant außerdem auf Basis der im Projekt gewonnenen Erkenntnisse und Erfahrungen im Bereich der Entwicklung der Labeling Pipeline ein Produkt zu entwickeln. Aufgrund der gewonnen Expertise konnte TWT ebenfalls bereits ein Angebot für einen Industrieauftrag bei einem Automotive Kunden abgeben und gewinnen.

2.8 Zusammenarbeit mit anderen Stellen oder außerhalb des Verbundprojekts

Im Verlauf des Projekts arbeitete TWT eng mit verschiedenen externen Partnern sowie thematisch angrenzenden Forschungsinitiativen zusammen, um technische Synergien zu nutzen und die Überführung der Projektergebnisse in die industrielle Anwendung vorzubereiten.

Durch die Kontaktaufnahme seitens der Konsortialführung mit dem öffentlich geförderten Forschungsprojekt VERANO, an welchem unter anderem Mercedes-Benz und Infineon beteiligt sind,

konnten mittels regelmäßiger Abstimmungen und gemeinsamer Workshops inhaltliche Überschneidungen zwischen NAOMI4Radar und VERANO identifiziert und von diesen beidseitig profitiert werden. Schwerpunkte lagen auf dem gegenseitigen Wissenstransfer, der Definition gemeinsamer Systemanforderungen sowie der Harmonisierung von Leistungskennzahlen (KPIs) für neuromorphe Verarbeitungssysteme. Darüber hinaus erfolgte eine wechselseitige Nutzung von Fahrzeugmessdaten, wodurch eine konsistente Datengrundlage und ein effizienter Ressourceneinsatz gewährleistet wurden.

In Zusammenarbeit mit Intel erfolgte eine intensive technische Abstimmung zur Integration der neuromorphen Plattform Loihi 2. Intel stellte die erforderliche Hardware bereit und unterstützte bei der Anpassung und Optimierung der entwickelten Algorithmen für den Echtzeitbetrieb. Der kontinuierliche Austausch ermöglichte wertvolle Erkenntnisse über die praktischen Anforderungen an neuromorphe Hardware im automobilen Umfeld, die in die Weiterentwicklung der Plattform einfließen.

Auch mit Infineon bestand eine enge Kooperation. Das Unternehmen lieferte den im Demonstrator verbauten Radarsensor und begleitete dessen Einbindung in die Gesamtsystemarchitektur. Neben der Bereitstellung technischer Unterstützung trug Infineon durch die Abstimmung von Schnittstellen und die Definition von Signalverarbeitungspfaden maßgeblich zum Projekterfolg bei.

Insgesamt bildeten diese Kooperationen wesentliche Bausteine, die Praxisrelevanz der Projektergebnisse zu erhöhen, den Wissenstransfer zwischen Industrie und Forschung zu fördern und eine solide Basis für zukünftige Entwicklungen im Bereich neuromorpher Sensorsysteme zu schaffen.

2.9 Darstellung des während des Vorhabens bekannt gewordenen Fortschritts auf diesem Gebiet bei anderen Stellen

Im Verlauf des Projekts wurde bekannt, dass das Air Force Research Laboratory (AFRL) in den USA ein Forschungsprojekt zur neuromorphen Radarverarbeitung an BrainChip Holdings in Kooperation mit Raytheon vergeben hat. Ziel dieses Vorhabens ist die Implementierung komplexer Signalverarbeitungsverfahren – darunter insbesondere die Analyse von Micro-Doppler-Signaturen – auf neuromorpher Hardware. Die von BrainChip entwickelten Akida-Prozessoren zeichnen sich durch einen sehr niedrigen Energieverbrauch aus und ermöglichen eine leistungsfähige Signalverarbeitung auf energie- und platzbeschränkten Plattformen wie unbemannten Luftfahrzeugen oder mobilen Systemen. Dieses Beispiel verdeutlicht das wachsende internationale Interesse an neuromorpher Technologie für Radar- und KI-Echtzeitanwendungen in unterschiedlichen Einsatzfeldern.

2.10 Erfolgte und geplante Veröffentlichungen

Im Rahmen des Vorhabens wurde die gemeinsame Publikation „NAOMI4Radar: Advancing Autonomous Driving with High-Efficiency Neuromorphic Radar“ auf dem ELIV-Kongress (Electronics in Vehicles) vorgestellt. Der Beitrag, erarbeitet von Mercedes-Benz, TWT, der Universität zu Lübeck, Intel, Infineon und der TU München, präsentiert die Umsetzung neuromorpher Radarsignalverarbeitung in einem Mercedes-Benz-Versuchsfahrzeug. Im Mittelpunkt stehen die energieeffiziente Verarbeitung radarbasierter Sensordaten auf dem Intel Loihi 2 mithilfe gepulster neuronaler Netze sowie die Definition von Key Performance Indicator (KPIs) zur Bewertung des Potenzials neuromorpher Ansätze für zukünftige autonome Fahrzeugsysteme. Neben dieser Publikation sind seitens TWT keine weiteren Veröffentlichungen zu NAOMI4Radar geplant. Die Doktorandinnen und Doktoranden der Partner bereiten jeweils eigene Publikationen zu ihren gewählten Algorithmen vor. Datenerhebung/-aufbereitung, experimentelle Auswertung und Manuskripterstellung laufen derzeit; die Einreichungen sind für die kommenden Projektphasen vorgesehen.

VII Literaturverzeichnis

- AlKhamissi, B., ElNokrashy, M., & Bernal-Casas, D. (2021). Deep spiking neural networks with resonate-and-fire neurons. arXiv preprint arXiv:2109.08234.
- Auge, D., Hille, J., Mueller, E., & Knoll, A. (2021, December). Hand gesture recognition in range-doppler images using binary activated spiking neural networks. In 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021) (pp. 01-07). IEEE.
- Barnell, M., Raymond, C., Loomis, L., Vidal, F., Brown, D., & Isereau, D. (2024, September). Spike-Driven YOLO: Ultra Low-Power Object Detection with Neuromorphic Computing. In 2024 IEEE High Performance Extreme Computing Conference (HPEC) (pp. 1-5). IEEE.
- Davies, M., Srinivasa, N., Lin, T. H., Chinya, G., Cao, Y., Choday, S. H., ... & Wang, H. (2018). Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1), 82-99.
- Fei, T., Guang, H., Sun, Y., Grimm, C., & Warsitz, E. (2021, January). An efficient sparse sensing based interference mitigation approach for automotive radar. In 2020 17th European Radar Conference (EuRAD) (pp. 274-277). IEEE.
- Fraday, E. P., Sanborn, S., Shrestha, S. B., Rubin, D. B. D., Orchard, G., Sommer, F. T., & Davies, M. (2022). Efficient neuromorphic signal processing with resonator neurons. *Journal of Signal Processing Systems*, 94(10), 917-927.
- Gamba, J. (2020). Target Filtering and Tracking. In: *Radar Signal Processing for Autonomous Driving. Signals and Communication Technology*. Springer, Singapore. https://doi.org/10.1007/978-981-13-9193-4_7
- Gentilho Jr, E., Scalassara, P. R., & Abrão, T. (2020). Direction-of-arrival estimation methods: A performance-complexity tradeoff perspective. *Journal of Signal Processing Systems*, 92(2), 239-256.
- Gordon, N. J., Salmond, D. J., & Smith, A. F. (1993, April). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE proceedings F (radar and signal processing)* (Vol. 140, No. 2, pp. 107-113). IEE.
- Higuchi, S., Kairat, S., Bohte, S., & Otte, S. (2024, July). Balanced Resonate-and-Fire Neurons. In *International Conference on Machine Learning* (pp. 18305-18323). PMLR.
- Higuchi, S., & Otte, S. (in press). On the Effectiveness of Label Smoothing in Spiking Neural Networks. *Proceedings in Austrian Symposium on AI, Robotics, and Vision (AIRoV) 2025*
- Hille, J., Auge, D., Grassmann, C., & Knoll, A. (2022, July). Resonate-and-fire neurons for radar interference detection. In *Proceedings of the International Conference on Neuromorphic Systems 2022* (pp. 1-4).
- Ikram, M. Z., & Ali, M. (2013, December). 3-D object tracking in millimeter-wave radar for advanced driver assistance systems. In 2013 IEEE Global Conference on Signal and Information Processing (pp. 723-726). IEEE.
- Izhikevich, E. M. (2001). Resonate-and-fire neurons. *Neural networks*, 14(6-7), 883-894.
- Jin, X., Zhang, M., Yan, R., Pan, G., & Ma, D. (2023). R-SNN: Region-based spiking neural network for object detection. *IEEE Transactions on Cognitive and Developmental Systems*, 16(3), 810-817.

- Kasenbacher, G., Ehret, F., Ecke, G., & Otte, S. (2025). WARP-LCA: Efficient convolutional sparse coding with Locally Competitive Algorithm. *Neurocomputing*, 130291. Elsevier.
- Kim, Y., Chough, J., & Panda, P. (2022). Beyond classification: Directly training spiking neural networks for semantic segmentation. *Neuromorphic Computing and Engineering*, 2(4), 044015.
- Klarenbeek, G., Harmanny, R. I. A., & Cifola, L. (2017, October). Multi-target human gait classification using LSTM recurrent neural networks applied to micro-Doppler. In *2017 European Radar Conference (EURAD)* (pp. 167-170). IEEE.
- Kosuge, A., Suehiro, S., Hamada, M., & Kuroda, T. (2022). mmWave-YOLO: A mmWave imaging radar-based real-time multiclass object recognition system for ADAS applications. *IEEE Transactions on Instrumentation and Measurement*, 71, 1-10.
- Lehmann, H. M., Hille, J., Grassmann, C., & Issakov, V. (2023). Direct signal encoding with analog resonate-and-fire neurons. *IEEE Access*, 11, 50052-50063.
- Liu, S., Pu, N., Cao, J., & Zhang, K. (2022). Synthetic aperture radar image despeckling based on multi-weighted sparse coding. *Entropy*, 24(1), 96.
- López-Randulfe, J., Duswald, T., Bing, Z., & Knoll, A. (2021). Spiking neural network for fourier transform and object detection for automotive radar. *Frontiers in Neurorobotics*, 15, 688344.
- López-Randulfe, J., Reeb, N., Karimi, N., Liu, C., Gonzalez, H. A., Dietrich, R., ... & Knoll, A. (2022). Time-coded spiking fourier transform in neuromorphic hardware. *IEEE Transactions on Computers*, 71(11), 2792-2802.
- Parpart, G., Risbud, S., Kenyon, G., & Watkins, Y. (2023, August). Implementing and benchmarking the locally competitive algorithm on the loihi 2 neuromorphic processor. In *Proceedings of the 2023 International Conference on Neuromorphic Systems* (pp. 1-6).
- Patel, K., Rambach, K., Visentin, T., Rusev, D., Pfeiffer, M., & Yang, B. (2019, April). Deep learning-based object classification on automotive radar spectra. In *2019 IEEE Radar Conference (Radar-Conf)* (pp. 1-6). IEEE.
- Pfeiffer, M., & Pfeil, T. (2018). Deep learning with spiking neurons: Opportunities and challenges. *Frontiers in neuroscience*, 12, 409662.
- Rahmani, M., & Akbarizadeh, G. (2015). Unsupervised feature learning based on sparse coding and spectral clustering for segmentation of synthetic aperture radar images. *IET Computer Vision*, 9(5), 629-638.
- Raimondi, M., Ciattaglia, G., Nocera, A., Senigaglia, L., Spinsante, S., & Gambi, E. (2024). mmDetect: YOLO-based processing of mm-wave radar data for detecting moving people. *IEEE Sensors Journal*, 24(7), 11906-11916.
- Rebut, J., Ouaknine, A., Malik, W., & Pérez, P. (2022). Raw high-definition radar for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 17021-17030).
- Rohling, H. (1983). New procedure for disturbance-adaptive target recognition in a CFAR processor. *Ortung und Navigation*, 15-1.

- Roy III, R. H., & Kailath, T. (1990). ESPRIT--Estimation of signal parameters via rotational invariance techniques. *Optical Engineering*, 29(4), 296-313.
- Rozell, C. J., Johnson, D. H., Baraniuk, R. G., & Olshausen, B. A. (2008). Sparse coding via thresholding and local competition in neural circuits. *Neural computation*, 20(10), 2526-2563.
- Schmidt, R. (1986). Signal subspace approach to high-resolution signal processing: The music algorithm. In *SEG Technical Program Expanded Abstracts 1986* (pp. 669-669). Society of Exploration Geophysicists.
- Schumann, A., & Stiefelhagen, R. (2017). Person re-identification by deep learning attribute-complementary information. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 20-28).
- Shrestha, S. B., Timcheck, J., Frady, P., Campos-Macias, L., & Davies, M. (2024, April). Efficient video and audio processing with loihi 2. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 13481-13485). IEEE.
- Siddiq, K., Hobden, M. K., Pennock, S. R., & Watson, R. J. (2018). Phase noise in FMCW radar systems. *IEEE Transactions on Aerospace and Electronic Systems*, 55(1), 70-81.
- Su, Q., Chou, Y., Hu, Y., Li, J., Mei, S., Zhang, Z., & Li, G. (2023). Deep directly-trained spiking neural networks for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 6555-6565).
- Sun, S., Petropulu, A. P., & Poor, H. V. (2020). MIMO radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges. *IEEE Signal Processing Magazine*, 37(4), 98-117.
- Viale, A., Marchisio, A., Martina, M., Masera, G., & Shafique, M. (2022, October). Lanesnns: Spiking neural networks for lane detection on the loihi neuromorphic processor. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 79-86). IEEE.
- Vogginger, B., Kreutz, F., López-Randulfe, J., Liu, C., Dietrich, R., Gonzalez, H. A., ... & Mayr, C. (2022). Automotive radar processing with spiking neural networks: Concepts and challenges. *Frontiers in neuroscience*, 16, 851774.