

Abschlussbericht der TU München – Teil I

KI-ASIC: Adaptive neuronale Netzarchitekturen für energieeffiziente neuromorphe KI im autonomen Fahrzeug

Robin Dietrich, Nico Reeb, Javier López Randulfe, Negin Karimi, Alois C. Knoll

30. Oktober 2023

1 Einleitung

Über einen Zeitraum von 3,5 Jahren hat die Technische Universität München (TUM) zusammen mit der TU Dresden (TUD), der Hochschule Amberg-Weiden, der Infineon AG sowie BMW an dem Projekt KI-ASIC gearbeitet. Das übergeordnete Ziel des Projekts war es, „*adaptive neuronale Netzarchitekturen für energieeffiziente neuromorphe KI im autonomen Fahrzeug*“ zu erforschen und zu entwickeln. Die Rolle der TUM lag dabei darin, neuartige Ansätze zur Verarbeitung von Radardaten mittels gepulster neuronaler Netze (SNNs) zu entwickeln. Diese Netze sind ideal dafür geeignet, um sie auf der, von der TUD entwickelten, neuromorphen Hardware (SpiNNaker-2) zu evaluieren. Die Kombination aus den spärlich kommunizierenden, gepulsten neuronalen Netzen und der hochparallelen neuromorphen Hardware verspricht eine hohe Energieeffizienz und geringe Latenzzeiten verglichen mit herkömmlichen, tiefen neuronalen Netzen [1].

Diese Vorteile wurden jedoch bislang nur selten in der Praxis nachgewiesen, eine kommerzielle Nutzung dieser Hardware ist daher aktuell noch nicht möglich. In der Literatur fehlt es noch an passenden Anwendungen, unter anderem da diese Netzwerke sehr spärlich und asynchron arbeiten. Herkömmliche Sensoren wie Kameras, Lidar oder Radar stellen jedoch mit regelmäßiger Frequenz ganze Datenblöcke bereit, nicht nur spärliche Änderungen der Umgebung wie neuromorphe Sensoren (bspw. Eventkameras [2]). Es ist daher notwendig, neue Verarbeitungsmethoden für existierende Sensorik zu erforschen. Um diesem Ziel einen Schritt näher zu kommen hat die TUM im Projekt KI-ASIC den Fokus vor allem auf den ersten Teil der Sensordatenverarbeitung gelegt - der Signaldatenverarbeitung - mit dem Ziel eines neuromorphen Radarsystems. Zu Beginn des Projekts gab es noch keinerlei Literatur zur Verarbeitung von FMCW Radardaten mittels SNNs. Während des Projektes sind unter anderem Forschungsarbeiten zur Verarbeitung von automobilen Lidardaten mit gepulsten neuronalen Netzen veröffentlicht worden [3, 4]. Darüber hinaus sind im Projektzeitraum einige automobiler Radardatensets [5] sowie Ansätze zur Gestenerkennung mittels SNNs und Radardaten veröffentlicht worden [6–8]; jedoch keine Ansätze um automobiler FMCW Radardaten mit SNNs zu verarbeiten.

2 Projektablauf/-durchführung

Während des Förderzeitraums des KI-ASIC Projekts hat die TUM daran gearbeitet die gesamte herkömmliche Radarverarbeitungskette durch neuromorphe Implementierungen zu ersetzen. Im speziellen wurden die folgenden Algorithmen durch Netzwerke von gepulsten neuronalen Netzen ersetzt: Analog-Digital (A/D) Wandler, Frequenzanalyse (Fouriertransformation), Spitzenwert-Erkennung (CFAR), Objektdaten-gruppierung (Clustering), Objektverfolgung (Tracking).

Bei der Auswahl der Lernstrategien hat die TUM einen Fokus auf unüberwachte Lernverfahren gewählt. Dies hat mehrerlei Gründe. Zum einen werden hierfür keine (von Hand) gekennzeichneten Datensätze benötigt, zum anderen sind gepulste neuronale Netze und auch neuromorphe Hardware besonders für unüberwachte Lernverfahren geeignet, da angenommen wird, dass diese Verfahren so auch in biologischen neuronalen Netzen essenziell sind [9].

3 Ergebnisse

Der erste Schritt in der Radardatenverarbeitung ist die Konvertierung von analogen in digitale Signale. Den üblich verwendeten Analog-Digital-Wandler hat die TUM in einem Versuch durch einen Schaltkreis ersetzt,

welcher die eingehenden Radarsignale verarbeitet und unmittelbar Pulse erzeugt, welche anschließend mit neuromorpher Hardware verarbeitet werden können [10]. Die nachfolgende Fouriertransformation wurde in einem gepulsten Netzwerk exakt nachgebaut, d.h., dass die Verbindungen des Netzwerks sowie die dazugehörigen Gewichte aufgrund der mathematischen Formeln der Fouriertransformation gewählt wurden [11, 12]. Als Ersatz für die Objekterkennung (CFAR) wurde sowohl von der TUM eine Variante (gepulste OS-CFAR) als auch von der TUD eine Variante (CA-CFAR) entwickelt. Sowohl die Fouriertransformation als auch die beiden CFAR-Varianten wurden auf dem neuromorphen Chip der TUD - SpiNNaker 2 - implementiert und evaluiert. Die Effektivität der Netzwerke wurde damit nachgewiesen - es gab keinerlei Unterschied der berechneten Ergebnisse gegenüber den herkömmlichen Algorithmen. Die Effizienz hingegen konnte noch nicht abschließend belegt werden, da die Hardware sich aktuell noch in der Entwicklung befindet.

Für den anschließenden Teil der Objektdatenverarbeitung wurden zwei verschiedene Netzwerke entwickelt und evaluiert. Mittels zeitkodierten Werten und unüberwachtem Lernen wurde ein Netzwerk für das Gruppieren von Radardatenpunkten trainiert. Das Netzwerk ist sehr effizient, da pro Eingangswert nur zwei Pulse erzeugt werden (x und y) und ist fähig die Datenpunkte mit einer Genauigkeit von ca. 60 % korrekt zu gruppieren. Für das Verfolgen von dynamischen Objekten wurde außerdem ein kontinuierliches Attraktornetzwerk basierend auf ratenkodierten Werten implementiert. Durch die Ratenkodierung ist das Netzwerk fähig sehr schnell auf Veränderungen zu reagieren und kann im Mittel ein Objekt zu 79 % und mit einer Genauigkeit von 3,22 m erkennen. Dieses Netzwerk wurde auf einer SpiNNaker 1 Platine evaluiert und erreicht eine Verarbeitungsrate von 100 Bildern/Sekunde.

Über diese Ergebnisse hinaus hat die TUM gemeinsam mit allen beteiligten Partnern am Projekt eine Veröffentlichung zum Stand der Technik sowie Herausforderungen der automobilen Radardatenverarbeitung mit Ergebnissen des Konsortiums veröffentlicht [13].

References

- ¹M. Davies et al., "Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook", *Proceedings of the IEEE* 109, 911 (2021).
- ²A. Lakshmi, A. Chakraborty, and C. S. Thakur, "Neuromorphic vision: From sensors to event-based algorithms", *WIREs Data Mining and Knowledge Discovery* 9, e1310 (2019).
- ³W. Wang et al., "Temporal Pulses Driven Spiking Neural Network for Fast Object Recognition in Autonomous Driving", en, *arXiv:2001.09220 [cs, eess]* (2020).
- ⁴S. Zhou et al., "Deep SCNN-Based Real-Time Object Detection for Self-Driving Vehicles Using LiDAR Temporal Data", *IEEE Access* 8, 76903 (2020).
- ⁵A. Ouaknine et al., "CARRADA Dataset: Camera and Automotive Radar with Range-Angle-Doppler Annotations", Feb. 8, 2021.
- ⁶A. Safa et al., "A 2- μ J, 12-class, 91% Accuracy Spiking Neural Network Approach For Radar Gesture Recognition", Aug. 24, 2021.
- ⁷I. J. Tsang et al., "Radar-Based Hand Gesture Recognition Using Spiking Neural Networks", *Electronics* 10, 1405 (2021).
- ⁸A. Shaaban et al., "Spiking Neural Networks for Gesture Recognition Using Time Domain Radar Data", in *2022 19th European Radar Conference (EuRAD)* (Sept. 2022), pp. 33–36.
- ⁹W. Gerstner and W. M. Kistler, "Mathematical formulations of Hebbian learning", *Biological Cybernetics* 87, 404 (2002).
- ¹⁰J. Lopez-Randulfe, N. Reeb, and A. Knoll, "Integrate-and-fire circuit for converting analog signals to spikes using phase encoding*", *Neuromorphic Computing and Engineering* 3, 044002 (2023).
- ¹¹J. López-Randulfe et al., "Spiking neural network for fourier transform and object detection for automotive radar", *Frontiers in Neurobotics* 15, 10.3389/fnbot.2021.688344 (2021).
- ¹²J. López-Randulfe et al., "Time-coded spiking fourier transform in neuromorphic hardware", *IEEE Transactions on Computers* 71, 2792 (2022).
- ¹³B. Vogginger et al., "Automotive Radar Processing With Spiking Neural Networks: Concepts and Challenges", *Frontiers in Neuroscience* 16, 851774 (2022).

Abschlussbericht der TU München – Teil II

KI-ASIC: Adaptive neuronale Netzarchitekturen für energieeffiziente neuromorphe KI im autonomen Fahrzeug

Robin Dietrich¹, Nico Reeb¹, Javier López Randulfe¹, Negin Karimi¹, Alois C. Knoll¹

¹School of Computation, Information and Technology, Technische Universität München (TUM), Boltzmannstr. 3, 85748 Garching, Deutschland
 30. Oktober 2023

Projektübersicht

Förderungskennzeichen	16ES0995
Projektname	KI-ASIC
Vorhabensbezeichnung GVB	KI-Prozessorarchitekturen für Radarmodule im autonomen Fahrzeug
Vorhabensbezeichnung TVB	Adaptive neuronale Netzarchitekturen für energieeffiziente neuromorphe KI im autonomen Fahrzeug
Koordinator Teilvorhaben	Alois Knoll – knoll@in.tum.de – +49 289 18116
Administrative Ansprechpartnerin	Yvonne Lechner – lechnery@in.tum.de – +49 289 17171
Laufzeit des Vorhabens	01.08.2019 - 30.04.2023
Berichtszeitraum	01.08.2019 - 30.04.2023

Verbundpartner

Nr.	Verbundpartner	Akronym	Adresse	Ansprechpartner
1	Infineon Technologies AG	IFAG	Am Campeon 1-12 85579 Neubiberg	Dr. Cyprian Grassmann
2	Infineon Technologies Dresden GmbH	IFD	Königsbrücker Str. 180 01099 Dresden	Dr. Oliver Pyper
3	BMW Group	BMW	Parkring 19 85748 Garching	Dr. Matthias Traub
4	Technische Universität Dresden	TUD	Helmholtzstraße 10 01069 Dresden	Prof. Christian Mayr
5	Ostbayerische Technische Hochschule Amberg-Weiden	OTH(-AW)	Kaiser-Wilhelm-Ring 23 92224 Amberg	Prof. Alfred Höß

Datum

Unterschrift

Inhaltsverzeichnis

1	Einführung	2
2	Arbeitspakete der TUM	3
2.1	Arbeitspaket 1	3
2.2	Arbeitspaket 2	5
2.3	Arbeitspaket 3	7
2.4	Arbeitspaket 4	8
2.4.1	Signaldatenverarbeitung	9
2.4.2	Objektdatenverarbeitung	11
2.4.3	Neuromorphe Radarverarbeitungsketten	13
2.5	Arbeitspaket 5	14
3	Weiterführende Informationen	17
3.1	Vorhaben, Ergebnisse & Zuwendungsverwendung	17
3.2	Wichtige Positionen des zahlenmäßigen Nachweises	18
3.3	Notwendigkeit & Angemessenheit der Projektarbeiten	18
3.4	Nutzen & Verwertbarkeit des Ergebnisses	18
3.5	Fortschritt auf dem Gebiet bei anderen Stellen	19
3.6	Veröffentlichungen der TUM	19
3.6.1	Erfolgte Veröffentlichungen	19
3.6.2	Geplante Veröffentlichungen	20
	Literatur	20

1 Einführung

Die Autonomie von Maschinen und Fahrzeugen schreitet in den letzten Jahren immer mehr voran. Gleichermaßen steigt auch der Energieverbrauch der Rechner, welche für die Wahrnehmung und Intelligenz eines solchen Systems zuständig sind aufgrund von immer höheren Anforderungen. Zudem ist bei der Entwicklung von Prozessoren eine physikalische Grenze erreicht und es können nicht immer mehr Transistoren auf kleinerem Raum verbaut werden. Neue Prozessorarchitekturen, wie Quantencomputer oder neuromorphe Prozessoren, sind damit von grundlegender Bedeutung für die Entwicklung von schnelleren und energiesparenderen Rechenarchitekturen der Zukunft.

Das Gesamtziel des Projekts KI-ASIC war daher die Erforschung, Implementierung und Evaluierung von neuromorphen Prozessoren (ASICs), einschließlich der gesamten, dem menschlichen Gehirn nachempfundenen Signalflosskette für gepulste neuronale Netze. Dieses Ziel wurde für das Anwendungsgebiet des autonomen Fahrens auf öffentlichen Straßen verfolgt. Mittels der neuromorphen Prozessoren sollte sowohl die Rechenleistung (Latenzzeit), verbessert, als auch die Leistungsaufnahme (Energieeffizienz) verringert werden. Die Teilziele, welche für die Erreichung dieses Gesamtziels notwendig sind, umfassen dabei zunächst die Entwicklung eines neuromorphen Mikroprozessors für die Integration in die Radarsensorik. Die Arbeit an diesem Ziel wurde primär von der TU Dresden (TUD) geleitet und durch Feedback der TUM, basierend auf den gesammelten Erfahrungen mit gepulsten Netzen und neuromorpher Hardware, ergänzt. Die für die Entwicklung von gepulsten Netzen benötigten Daten wurden in dem Projekt von der OTH-AW aufgenommen. Die TUM hat an diesem Prozess maßgeblich Einfluss genommen, um die Struktur und die Korrektheit der Daten zu verifizieren. Das hierbei verwendete Demonstratorfahrzeug wurde von der BMW AG gestellt und die Sensorik von der Infineon AG. Die Hauptaufgabe der TUM lag in der Entwicklung neuromorpher Algorithmen, die auf gepulsten Netzen aufbauen. Hierbei konnte die TUM maßgebliche Erfahrungen aus anderen Forschungsprojekten zur künstlichen Intelligenz sowie neurobiologisch inspirierte Algorithmen einbringen. Bei der Entwicklung wurde Wert auf effiziente und stromsparende Ansätze gelegt. Neben der TUM, arbeitete die TUD und Infineon AG ebenfalls an neuromorphen Algorithmen.

2 Arbeitspakete der TUM

Das Projekt KI-ASIC gliedert sich in 5 technische und 1 administratives Arbeitspaket, die die zielgerichtete Erforschung neuromorpher ASICs, einschließlich der zu erforschenden und darauf zu implementierenden Algorithmen für gepulste neuronale Netze für den Einsatz in autonomen Fahrzeugen ermöglichen. Abbildung 1 zeigt diese Projektstruktur. Der vollständige Arbeitsplan des Projektverbunds ist in Kapitel 3 der Gesamtvorhabenbeschreibung dokumentiert. Im Folgenden wird die durchgeführte Arbeit sowie die daraus entstandenen Ergebnisse der Technischen Universität München (TUM) im Bezug auf die einzelnen Arbeitspakete beschrieben. Ein besonderer Fokus liegt hierbei auf dem AP4 "KI-SW: Netzmodelle, Lernverfahren, Signal-Processing", bei welchem die TUM die Leitung übernahm. Das Arbeitspaket beinhaltet die Entwicklung von neuartigen Netzmodellen und deren Implementierung mit neuen bzw. existierenden Lernverfahren zur Signal- bzw. Objektdatenverarbeitung von Radardaten (siehe Abbildung 3).

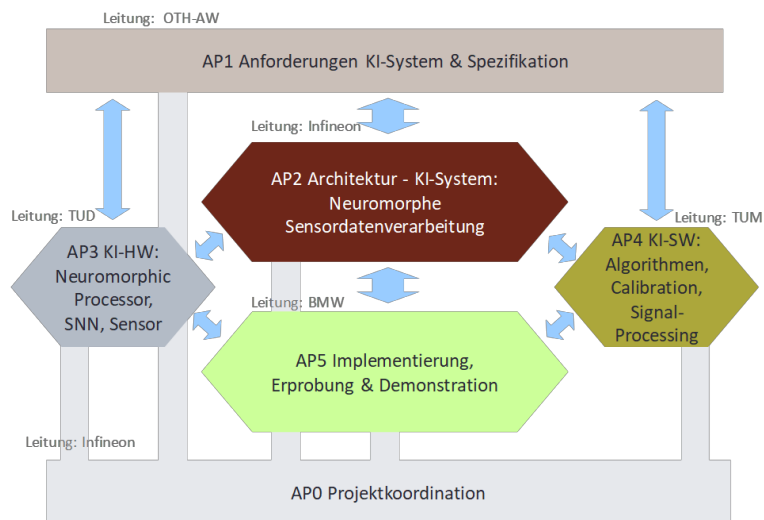


Abbildung 1 Struktur und Arbeitspakete des Verbundprojekts KI-ASIC

Für die Entwicklung und Einbettung dieser Netzmodelle in das Gesamtsystem wurde in AP2 gemeinsam mit den Partnern des Projekts eine Softwarearchitektur entworfen. Ergänzend wurden in AP1 grundsätzliche Anforderungen an die Netzwerke, verwendete Datenstrukturen, (externe) Software und verbaute Hardware von der TUM mitdefiniert. Zur Evaluierung der Netzwerke hat die TUM im Rahmen des AP3 einen mobilen Roboter als lokales Testsystem angeschafft und verwendet. Die vielversprechendsten bzw. passendsten Netzwerke wurden dann in AP5 in enger Kooperation mit den Kollegen der TUD auf der neuromorphen Hardware Plattform SpiNNaker-2 implementiert und getestet. Die Ergebnisse der von der TUM geleisteten Forschungs- und Entwicklungsarbeit innerhalb der einzelnen Pakete werden nachfolgend ausführlich beschrieben.

2.1 Arbeitspaket 1

Im Arbeitspaket 1 wurden die Anforderungen und Spezifikationen an das Signalverarbeitungssystem definiert. Hierbei erarbeitete die TUM insbesondere die Anforderungen der einzelnen Prozessschritte der neuronalen Datenverarbeitung (T1.1) sowie die zu verwendenden Datenschnittstellen und das Datenspeicherformat (T1.2). Die jeweiligen Anforderungen (T1.1) hängen von der finalen Anwendung des Algorithmus ab, und können nicht allgemein definiert werden. Daher fasste die TUM vorab folgende Systemparameter zusammen, die die Anforderungen beschreiben und klassifizieren: Latenz, Signalrauschen, Anwendung (Ziel des Algorithmus), Datenstruktur und Energieverbrauch.

Da die entwickelten Algorithmen auf Spiking Neuron Networks (SNNs), zu Deutsch gepulste neuronale Netzwerke, beruhen, wurden zudem die Parameter der SNNs in Tabelle 1 zusammengefasst, die Einfluss auf die Systemparameter haben.

Tabelle 1 Parameter von gepulsten neuronalen Netzwerk, die Latenz, Signalrauschen, Anwendung (Ziel des Algorithmus), Datenstruktur und Energieverbrauch beeinflussen.

Parameter	Wirkung
Anzahl der Neuronen und Synapsen	Energieverbrauch der benötigten Hardware
Anzahl der Schichten im Netz	Systemlatenz
Lernrate und Algorithmus	Anpassung an neuen Umgebungen, Dauer der Einrichtung, Erfolgsrate
Netzwerkarchitektur	Systemlatenz, Hardwarearchitektur
Verzögerung an den Synapsen	Systemlatenz
Größe des Trainings und Validierungsdatensatzes	Dauer der Einrichtung

Abhängig von der Anwendung des jeweiligen Algorithmus, gibt es unterschiedliche Mindestanforderungen an die Systemparameter und somit die Parameter der Netzwerke. Diese werden für die Algorithmen im Einzelnen definiert.

Darüber hinaus ist das Ziel des Projektes schrittweise die einzelnen Verarbeitungsschritte durch neuromorphe Algorithmen zu ersetzen. Daher werden die endgültigen Verarbeitungsketten aus klassischen und neuromorphen Algorithmen bestehen. Die Schnittstelle zwischen klassischen und neuromorphen Algorithmen spielt eine entscheidende Rolle und wurde detailliert untersucht.

Während die herkömmliche Radarverarbeitung über digitalisierte Daten erfolgt, verwenden neuromorphe Algorithmen eine Pulsreihe, bei welchen die Informationen in den Pulszeitpunkten kodiert sind. Daher ist es notwendig, eine Datenkonvertierung durchzuführen, die auf zwei Hauptansätzen basieren kann:

- Datenkonvertierung vom analogen, rohen Radarsignal zu Pulse,
- Datenkonvertierung von digitalisierten Daten zu Pulse und deren Rückkonvertierung.

Die Auswahl des Datenkonvertierungsparadigmas hängt vollständig von der Stufe der Verarbeitungskette ab, die durch die neuromorphen Algorithmen ersetzt wird. Wenn die Algorithmen mit den Rohdaten des Sensors versorgt werden, ist eine Analog-zu-Spike-Konvertierung geeignet. Andererseits, wenn die neuromorphen Algorithmen auf vorverarbeiteten Daten angewendet werden, um eine hochstufige Datenanalyse durchzuführen, ist eine Digital-zu-Spike-Konvertierung geeignet und ggf. ein Spike-zu-Digital Konvertierung, um die Ergebnisse auszulesen oder weiter zu verarbeiten.

Im zweiten Unterpunkt des AP1 (T1.2) befasste sich die TUM mit den Datenschnittstellen und Datenspeicherformaten. Die Schnittstelle des neuronalen Netzwerks hängt vollständig von dem Verarbeitungsschritt ab, auf den das SNN angewandt wird.

Die Hauptmerkmale der Verarbeitungsschnittstelle sind in der folgenden Liste für jeden Schritt in der Verarbeitungspipeline zusammengefasst:

- Analoges Signal: Kontinuierliche, unstrukturierte Daten, viel Rauschen.
- Datenwürfel: Die Daten sind bereits im digitalen Bereich, aber immer noch in Form von Spannungswerten formatiert.
- Entfernung-Geschwindigkeit-Winkel Karte: Hier ist die Dichte der Datenproben deutlich geringer, das Rauschen wurde reduziert, und die Daten sind im Frequenzbereich in 2D, 3D oder 4D-Karten strukturiert.
- Punktwolke: Dies ist der letzte Schritt der Radar-Signalverarbeitungskette, mit der geringsten Datendichte und dem wenigsten Rauschen.

Die Tabelle 2 fasst die Hauptmerkmale der Schnittstellen zum SNN für jeden Schritt der Radar-Pipeline zusammen, wobei die Eingangspulsrate entscheidend ist und von den Eigenschaften des Radarsensors und der Verarbeitungspipeline abhängt. Es ist entscheidend, dass die Eingangspulsrate mit der verwendeten neuromorphen Hardware kompatibel ist.

Für die effiziente Verwertung der Daten und der Evaluierung der Algorithmen spielt das Datenspeicherformat eine entscheidende Rolle. Die TUM hat hierfür verschiedene Speicherformate getestet. Für das

Tabelle 2 Zusammenfassung der Schnittstelleneigenschaften für jede Stufe der Radarverarbeitungspipeline.

	Eingangsspike	Ausgangsspike	Kodierung
Analoges Signal	$10^4 - 10^6$ Hz	$10^1 - 10^2$ Hz	Analog-zu-Spike
Datenwürfel	$10^4 - 10^6$ Hz	$10^1 - 10^2$ Hz	Analog-zu-Spike
Entfernung-Geschwindigkeit-Winkel	$10^1 - 10^2$ Hz	10 Hz	Rate / Zeit
Punktwolke	10 Hz	10 Hz	Rate / Zeit

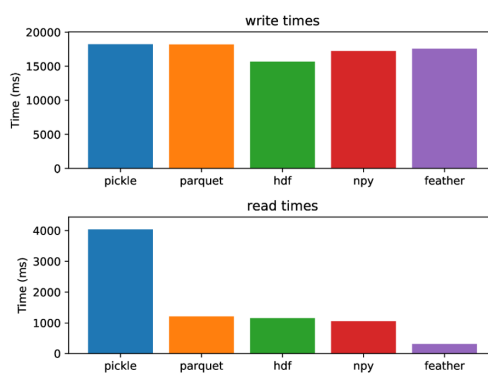


Abbildung 2 Lese- und Schreibzeiten für eine Datengröße von 5 GB für verschiedene Datenspeicherformate.

Projekt wurde das Speicherdatenformat HDF5 ausgewählt. Dieses Format ist besonders stabil, leistungsfähig und flexibel. Es kann plattformübergreifend verwendet werden und ist mit verschiedenen Programmiersprachen kompatibel. HDF5 ermöglicht die Speicherung von Daten in einem hierarchischen Format mit Gruppen und kleinen Datensätzen. Es gibt offizielle Bibliotheken für die wichtigsten Programmiersprachen, es ist langfristig stabil, hat eine hohe Leistung in Bezug auf Lese-/Schreibgeschwindigkeit sowie Speicher und ist zudem quelloffen (Open Source). In einem Experiment mit 5 GB Daten zeigt sich, dass HDF5 die beste Schreibgeschwindigkeit bietet, während das Feather-Format die besten Lesegeschwindigkeiten aufweist, jedoch für das Projekt ungeeignet ist, da es derzeit nicht in einer stabilen Version vorliegt.

Zusammen mit der OTH wurde die Datenstruktur und die gespeicherten Metainformationen festgelegt. Für einen strukturierten und einfachen Umgang mit den Daten hat die TUM darüber hinaus eine Software entwickelt, die es ermöglicht die Daten mit wenigen Befehlen von den Servern zu laden und zu verarbeiten, sodass die Partner mit den gleichen und aktuellen Testdaten arbeiten und ihre Ergebnisse miteinander vergleichen konnten.

2.2 Arbeitspaket 2

Die Basis für die (neuromorphe) System- bzw. Softwarearchitektur, welche innerhalb dieses Arbeitspakets entwickelt wurde, bildet die konventionelle Radarverarbeitungskette, wie sie aktuell in den meisten Systemen eingesetzt wird (siehe Abbildung 3). Diese Kette besteht zunächst aus den Rohdaten, welche von einem frequenzmodulierten Dauerstichradar (FMCW Radar) erzeugt werden. Dem folgen drei Algorithmen bzw. Komponenten (grün), welche typischerweise zur Signaldatenverarbeitung verwendet werden:

ADC Analog-Digital-Konverter - Konvertiert das analoge Radarsignal in einen digitalen Datenstrom

FFT Schnelle Fourier-Transformation - Berechnet Frequenzanteile in den digitalen Daten

CFAR Konstante Falschalarmrate - Erkennung von Objekten und Entfernung von Hintergrundrauschen

Zusätzlich kann nach der Fourier-Transformation noch eine Einfallswinkelberechnung (AoA) der reflektierten Radardaten durchgeführt werden. Mittels einer Kombination mehrerer Empfangsantennen wird bei diesem Algorithmus mit Hilfe der Triangulation der Einfallswinkel des reflektierten Signals berechnet. Für die anschließende Objektdatenverarbeitung sind nachfolgend beispielhaft zwei Algorithmen dargestellt, welche üblicherweise die Daten nach der CFAR verwenden:

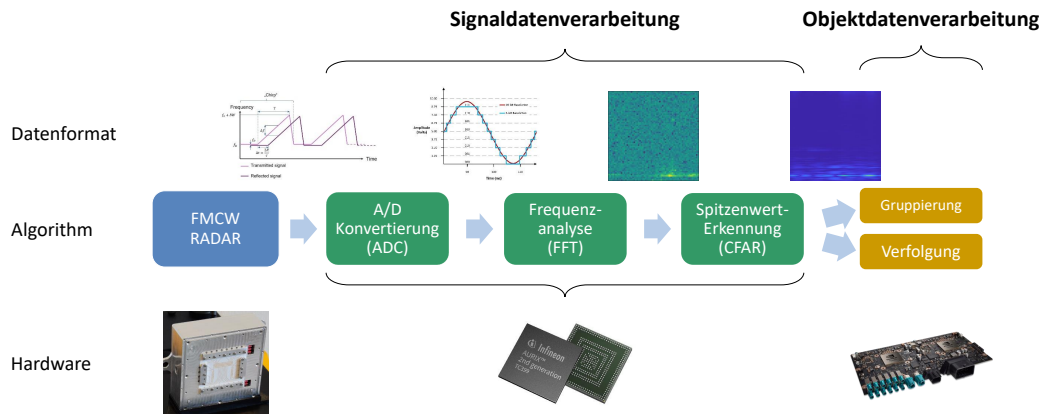


Abbildung 3 Die Algorithmen der konventionellen Radarverarbeitungskette (mittig) mit den dazugehörigen Ausgangsdaten je Schritt (oben) sowie der Hardware auf der dieser ausgeführt wird (unten).

Clustering Gruppierung von Objektdatenpunkten in räumlicher Nähe

Tracking Verfolgung sowie möglicherweise Prädiktion von Objekten bzw. Objektdatenpunkten

Diese Informationen über den Aufbau der konventionellen Radarverarbeitungskette wurden gemeinsam mit den Partnern der Infineon AG sowie der BMW AG im Rahmen der Aufgabenpakete T2.1, T2.2 sowie T2.3 zusammengetragen.

Basierend auf diesen Grundlagen wurde anschließend eine flexible, modulare System- bzw. Softwarearchitektur entwickelt. Die Systemarchitektur wurde gemeinsam vom gesamten Konsortium entworfen (T2.1) und ist in Abbildung 4 dargestellt.

Das hier dargestellte KI-ASIC-System für die Radarsignal- und Datenverarbeitung besteht aus vier Hauptkomponenten: einem hochauflösenden Radar, einem Ethernet-Switch, dem SpiNNaker Board und einem PC. Der Datenfluss des Systems beginnt bei den Radar-mmICs des hochauflösenden Radars. Sie sammeln die Radarsignale und leiten sie an die AURIX-Mikrocontroller auf der Radarplatine weiter. Der AURIX übernimmt die eingehenden Daten und führt entweder die ersten Schritte der Vorverarbeitung durch (Abbildung 3, grün) oder leitet die Rohdaten weiter. Die (vorverarbeiteten) Daten werden dann über einen Ethernet-Switch an das SpiNNaker Board übertragen. Das Board besteht aus zwei Unterkomponenten – den ARM-Prozessoren und dem SpiNNaker 2 FPGA. Die Arm-Prozessoren können die Daten vorverarbeiten, bspw. in gepulste Daten konvertieren. Anschließend können die Daten im FPGA des SpiNNaker 2 Boards verarbeitet, mit Hilfe der ARM Prozessoren wieder konvertiert und dann weiter, bspw. an den PC übertragen werden. Auf dem PC wird dann auch das User Interface ausgeführt, um die finalen Ergebnisse anzuzeigen. Die Verarbeitungsschritte im PC und auf dem SpiNNaker Board sind hier völlig flexibel und können sich auch mehrfach abwechseln innerhalb einer Verarbeitungskette. Demnach könnten auch bspw. die Rohdaten vom Hi-Res Radar an das SpiNNaker Board übertragen werden um eine gepulste Fourier-Transformation durchzuführen. Anschließend können die Daten dann an den PC für die weitere Verarbeitung übermittelt werden. Diese Architektur ermöglicht somit den flexiblen Austausch von Algorithmen bzw. deren Plattformen in der herkömmlichen Verarbeitungskette.

Auf dieser Systemarchitektur aufbauend hat die TUM eine Softwarearchitektur entwickelt, welche nicht nur die Erprobung, sondern vor allem auch die Entwicklung von neuromorphen Komponenten erleichtert. Der Hauptaspekt bei der Entwicklung der Architektur war die Modularität. Diese ermöglicht es in der entwickelten Architektur Komponenten (*Algorithmen*) flexibel durch andere, bspw. neuromorphe Komponenten, zu ersetzen und somit flexible *Pipelines* zu konstruieren. Die verschiedenen Klassen der in Python implementierten Architektur sowie deren Abhängigkeiten sind in Abbildung 5b dargestellt. Eine schematische Darstellung des funktionalen Prinzips ist in Abbildung 5a zu sehen. Die Algorithmen (schwarze Umrandung) können beliebig ausgetauscht werden. Die Pfeile zwischen den Algorithmen stellen den Verlauf der Radarkette dar. Werden zwei neuromorphe Blöcke miteinander verbunden, so kann die Kommunikation direkt stattfinden. Folgen jedoch ein neuromorpher und ein klassischer Algorithmus aufeinander, so wird ein

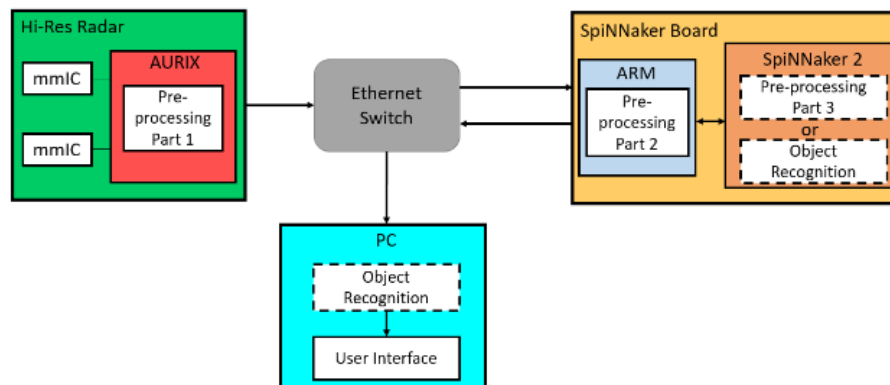


Abbildung 4 Die Systemarchitektur des (neuromorphen) Radarverarbeitungssystems.

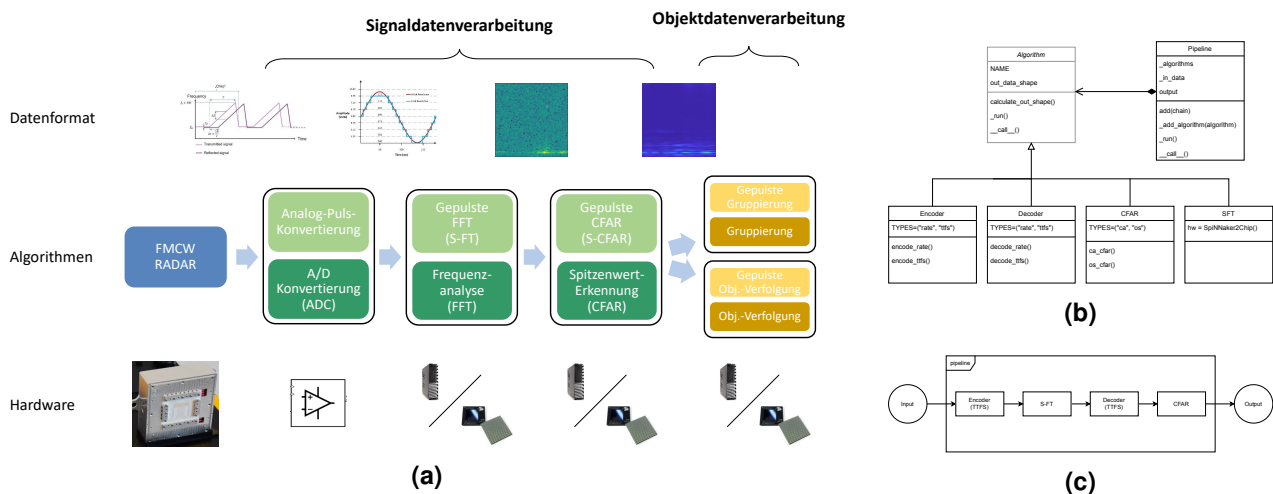


Abbildung 5 Die Softwarearchitektur des (neuromorphen) Radarverarbeitungssystems. **(a)** Schematischer Ablauf der Radarverarbeitungskette mit Hilfe der Softwarearchitektur. Schwarz umrandete Blöcke sind austauschbare Algorithmen, welche auf beliebiger Hardware (PC/neuromorph) ausgeführt werden können. **(b)** Die Softwarearchitektur als Klassendiagramm mit der zentralen Struktur der *Pipeline*, welche ausgeführt werden kann und selbst aus anderen Pipelines bzw. *Algorithmen* besteht. Spezifische Algorithmen (bspw. die CFAR) können von der Basisklasse abgeleitet werden. **(c)** Eine Beispielanwendung mit verschiedenen Algorithmen (eckige Blöcke) innerhalb einer Pipeline, welche Eingangsdaten (Kreis) verarbeiten und Ausgangsdaten (Kreis) erzeugen.

zusätzlicher Algorithmus Block dazwischen benötigt, welcher die Werte (de-)kodiert. Ein Beispiel für solch ein Szenario ist in Abbildung 5c dargestellt. Hier folgt ein klassischer Algorithmus (CFAR) auf einen neuromorphen Algorithmus (S-FT). Dafür wird in diesem Fall ein Kodierer (TTFS - Time-to-first-spike) verwendet um die Fließkommazahlen des A/D Wandlers in gepulste Werte für SNNs umzuwandeln. Nach der S-FT folgt dann ein Dekodierer (TTFS) um aus den Pulsen wieder Fließkommazahlen für die Weiterverarbeitung in der CFAR zu erzeugen.

Die vorgestellte Bibliothek bietet somit eine flexible Möglichkeit um neue Algorithmen (klassisch, künstliche neuronale Netze oder gepulste Netze) zu implementieren und gemeinsam zu evaluieren. Sie soll außerdem über das Projekt hinaus einen einfachen Einstieg in die Radarverarbeitung für neue Forscher und Entwickler in diesem Themengebiet bilden. Die TUM plant daher die Veröffentlichung des Quellcodes für die Pipeline Bibliothek im Jahr 2024.

2.3 Arbeitspaket 3

Im Rahmen des Arbeitspakets 3 hat die TUM daran gearbeitet, ein für dieses Projekt geeignetes, lokales Testsystem zu identifizieren, die nötige Hardware hierfür zu beschaffen und das System aufzubauen bzw. in Betrieb zu nehmen. Nachdem zunächst ein Versuchsaufbau ähnlich zu einem "Core XY"3D-Drucker



Abbildung 6 Die mobile Roboterplattform der TUM, ausgestattet mit Stereokamera & Radarsensor (Infineon AG).

diskutiert wurde, bei dem der Extruder durch einen Radarsensor ersetzt wird, ist die Entscheidung dann doch auf eine mobile Roboterplattform gefallen. Der Hauptgrund hierfür ist, dass solch ein Versuchsaufbau starke Reflexionen des Radarsignals verursacht hätte. Diese können durch das Anbringen von Absorptionsmaterial reduziert bzw. durch Filter teilweise entfernt werden, jedoch ist ein automobiler Radarsensor nur sehr bedingt für Innenbereiche geeignet. Eine mobile Roboterplattform hingegen bietet eine enorme Flexibilität, denn hier können die Algorithmen, welche in AP4 entwickelt wurden, auf ähnliche Weise getestet werden, wie es auch im Versuchsträger möglich ist. Zudem bietet der Roboter noch die Möglichkeit zusätzlich Daten aufnehmen zu können und diesen auch direkt steuern zu können, bspw. durch ein gepulstes neuronales Netz.

Die TUM hat daher in T3.1 für die Anschaffung eines Clearpath Jackal Außenroboters entschieden (siehe Abbildung 6). Dieser Roboter ist vielseitig einsetzbar, da er IP-zertifiziert und somit wasserdicht ist. Des weiteren erlauben die großen Räder einen Einsatz in verschiedenstem Terrain. Speziell aufgrund der Tatsache, dass Radarsensoren deutlich besser in schlechteren Wetterbedingungen funktionieren als bspw. ein Lidar oder eine normale Kamera, ist diese Robustheit des Roboters relevant gewesen. Der Roboter wurde von der TUM mit einem aktuellen NUC PC-System von Intel, einem WLAN-Router mit Netzwerkschicht sowie der in Abbildung 6 dargestellten Sensorik nachgerüstet. Somit ist der Roboter in der Lage, eine ähnliche Systemarchitektur bereitzustellen wie die in AP2 beschriebene. Dieser Aufbau ermöglicht der TUM somit ein Demonstrator-ähnliches Testumfeld.

Nachdem die TUM den Roboter vollständig aufgesetzt und in Betrieb genommen hatte (T3.1) wurde dieser zunächst verwendet um den Radarsensor zu testen und das Gesamtsystem hiermit zu prüfen. Im Besonderen wurde die Aufnahme von Szenarien für eine Auswertung der Angle-of-Arrival (AoA) Algorithmen fokussiert, um die hohe Winkelauflösung des (HiRes-) Radarsensors hervorzuheben. Hierbei ist die Kalibrierung der einzelnen Antennen entscheidend. Auch nach intensiver Zusammenarbeit mit der Infineon AG konnte das Basismodell des Sensors, welcher der TUM zur Verfügung stand, nicht ausreichend kalibriert werden und somit die Daten des Sensors nicht für die Auswertung der AoA Algorithmen verwendet werden. Die aufgenommen Rohdaten wurden allerdings anschließend für die Entfernungsbestimmung verwendet.

Darüber hinaus war es innerhalb der Projektlaufzeit aufgrund verschiedener Verzögerungen leider nicht möglich für die TUM ein SpiNNaker 2 Board für die Integration in den Roboter zu erhalten. Die TUM plant dies jedoch im Jahr 2024 nachzuholen, sobald die Hardware von der TUD bereitgestellt werden kann. Anschließend werden noch Experimente mit den in AP4 beschriebenen Algorithmen auf der Roboterplattform durchgeführt und ihre Effektivität somit in einem Echtzeit-Experiment demonstriert.

2.4 Arbeitspaket 4

Die Hauptaufgabe der TUM im Projekt KI-ASIC war die Erforschung, Implementierung und Evaluierung von Neuronen- bzw. Netzmodellen sowie passenden Lernverfahren für die Signal- bzw. Objektdatenverarbeitung von Radardaten. Aufgrund von Verzögerungen bei der Erstellung des Datensatzes sowie der

generell geringen Verfügbarkeit von Radardatensätzen im Automobilbereich hat die TUM einen Fokus auf unüberwachte Lernverfahren sowie fest definierte mathematisch abgeleitete Netzwerkstrukturen gelegt. Somit wird nicht nur die Abhängigkeit von großen, gekennzeichneten Datensätzen verringert, sondern auch die Möglichkeit für effiziente Strukturen (mathematikbasiert) bzw. biologisch inspirierte Lernverfahren ermöglicht. Nachfolgend werden die einzelnen Ansätze der TUM in der Reihenfolge ihres Vorkommens in der Radarverarbeitungskette, beginnend mit den Ansätzen zur Signaldatenverarbeitung, vorgestellt und die erzielten Ergebnisse aufgezeigt.

2.4.1 Signaldatenverarbeitung

Für die Signaldatenverarbeitung hat die TUM schrittweise die Verarbeitungsblocks, die in Abbildung in grün dargestellt sind, mit gepulsten neuronalen Netzen ersetzt.

Analog-zu-Puls Kodierer Die Verarbeitung von Sensordaten mit gepulsten neuronalen Netzwerken auf digitalen neuromorphen Chips erfordert die Umwandlung von kontinuierlichen analogen Signalen in Pulse. Hierzu existieren unter anderem zwei Optimierungsmöglichkeiten, welche einen geringen Energieverbrauch sowie schnelle Verarbeitungsgeschwindigkeiten in neuromorphen Ende-zu-Ende Anwendungen gewährleisten. Die erste Optimierung besteht darin, analoge Signale direkt in Pulse zu kodieren, um den Bedarf an einem Analog-Digital-Umsetzer zu umgehen. Als zweite Verbesserung wird eine zeitliche Kodierungstechnik verwendet, um die Puls-Sparsamkeit zu maximieren, was ein entscheidender Parameter für schnelle und effiziente neuromorphe Verarbeitung ist. Der von der TUM entwickelte Analog-zu-Puls Kodierer (Engl. analog-to-spike encoder, ASE) vereint beide Strategien und konvertiert kontinuierliche analoge Signale in zeitkodierte Pulse mit Hilfe eines optimierten "Leaky Integrate-and-Fire" (LIF) Neuronenmodells. Dieses Modell steuert aktiv die Refraktärzeiten der Neuronen, um einen Datenpunkt mit einem einzigen Puls zu kodieren. Die Information ist hierbei im Zeitpunkt des Pulses gespeichert, ein früher Puls spiegelt einen hohen Datenwert wider und ein später Puls einen geringen Wert (Time-to-first-spike Kodierung). Diese Kodierungsmethode wird als Phasenkodierung und im speziellen als TTFS-Kodierung bezeichnet, da die Information in der Phase eines Referenzsignals enthalten ist. In unserem Modell ist das Referenzsignal durch ein fallendes lineare Signal im vordefinierte Zeitfenster $[t_m, t_{m+1}]$ gegeben und entspricht damit der TTFS Kodierung. In Abbildung 7 ist beispielhaft die Generation eines Pulses dargestellt. Das Neuronenmodell wurde auf einer physischen Schaltung implementiert und mit verschiedenen elektrischen Signalen getestet. Die genaue Funktionsweise kann in der dazugehörigen Publikation nachgelesen werden [1].

Gepulste Fouriertransformation Die TUM hat die Entwicklung einer gepulsten Fouriertransformation mit einer raten-basierten Variante begonnen. Hierbei werden die Werte durch die Rate der Pulse kodiert und die Gewichte des Netzwerks durch die komplexen Gewichte der klassischen diskreten Fouriertransformation (DFT) ersetzt [2]. Sofern das Zeitfenster der Ratenbestimmung ausreichend lang ist liefert die ratenbasierte Fouriertransformation identische Ergebnisse zur klassischen DFT. Allerdings ist anzunehmen, dass der Energieverbrauch aufgrund der hohen Pulszahl ungeeignet ist für Automobil-Anwendungen. Daher erarbeitete die TUM eine optimierte Version der gepulsten Fouriertransformation, welche auf einer zeitbasierten Kodierung beruht und kompatibel mit dem ASE ist. Dazu wurde eine neues Neuronenmodell entwickelt, das mit zwei Stufen arbeitet [3]. Die erste Stufe dekodiert die zeitbasierten Pulse und führt eine Vektor-Vektor-Multiplikation durch. Auch hier werden die Gewichte der DFT verwendet und ein Lernen ist nicht notwendig. Die zweite Stufe kodiert den Wert des Neurons zu einem zeitbasierten Puls, sodass ein nachfolgendes Neuron die Daten weiterverarbeiten kann. Das Stufenmodell ist in Abbildung 8 beispielhaft dargestellt. Die Aneinanderreihung der Neuronen ermöglicht die Berechnung von mehrdimensionalen Fouriertransformationen, und somit die Radardatenverarbeitung zu Entfernung-Geschwindigkeit, Entfernung-Winkel Karten (2D) oder Entfernung-Geschwindigkeit-Winkel Karten (3D). Mit diesem Modell war es der TUM möglich die Anzahl der Pulse stark zu reduzieren und einen für neuromorphe Hardware optimierten Algorithmus zu bilden.

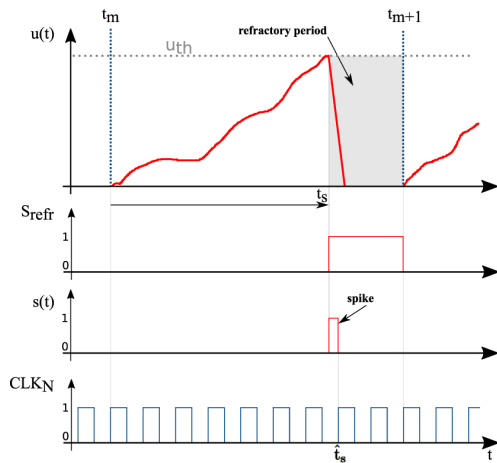


Abbildung 7 Signale des Analog-zu-Puls Kodierers (ASE) für die Erzeugung eines Pulses von oben nach unten: Membranspannung, Refraktärzustand, Ausgangspuls, Abtastrate. [1]

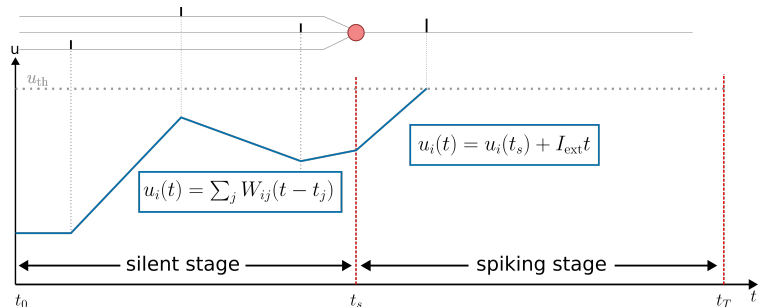


Abbildung 8 Beispielhafte Darstellung des Zweistufen-Neuronenmodells. Die "silent stage" führt eine Vektormultiplikation durch, die "spiking stage" wandelt den Membranwert in einen phasenkodierten Puls. [4]

Strahlformung Resonate-and-Fire Netzwerk Neben der zeitbasierten Fouriertransformation untersuchte die TUM auch alternative Neuronenmodelle, die sich für eine Frequenzanalyse eignen. Für diese Zwecke entwickelte die TUM ein Netzwerk, das aus "Resonate-and-Fire" (RF) Neuronen besteht. Diese Neuronen oszillieren mit einer vordefinierten Eigenfrequenz und addieren zu ihrem aktuellen Zustand eintreffende Signale. Wenn die Frequenz der Neuronenoszillation mit der Frequenz des Eingangssignals übereinstimmt, kommt es zu einer Resonanz und die Amplitude des Neuronenzustands steigt exponentiell an. Dieses Verhalten kann dazu genutzt werden, Frequenzen aus einem existierenden Signal zu filtern. Diese Methode bietet den Vorteil, dass eintreffende Daten direkt verarbeitet werden und diese nicht erst gesammelt werden müssen. Das Radarsignal kann somit kontinuierlich verarbeitet werden. In diesem Zusammenhang setzte sich die TUM mit der Bestimmung der Ankunftsrichtung auseinander. Dieser Winkel wird typischerweise mit den Daten von mehreren Empfängerantennen berechnet. Zum einen ermöglicht dieses Datenformat die Vorteile von SNNs zu nutzen, nämlich die parallele Verarbeitung der Daten, zum anderen können keine RF Neuronen für diesen Zweck verwendet werden, da die Daten nicht konsekutiv eintreffen, sondern gleichzeitig von jeder Antenne. Aus diesem Grund verwendet die TUM Strahlformungsmethoden, um den Ankunftsrichtung zu bestimmen. Strahlformungsmethoden richten künstlich den Blick der Antennen in eine bestimmte Richtung, und überprüfen die Existenz eines Objektes bzw. eines erhöhten Eingangssignals aus dieser Richtung. Mathematisch wird diese Operation mit einer Matrixmultiplikation dargestellt. Demnach kann eine Entfernung-Winkel Karte bestimmt werden, indem eintreffende Signale von verschiedenen Antennen komplex gewichtet zu RF Neuronen geführt werden. Diese Architektur ist in Abbildung 9 dargestellt. Bisherige Ergebnisse zeigen, dass Objekte schon frühzeitig erkannt werden können im Vergleich zu einer blockbasierten FFT. Aktuelle Auswertungen, welche die Vorteile dieser Methode aufzeigen, werden noch in diesem Jahr veröffentlicht.

Deep Learning Ansatz Neben der Entwicklung von neuromorphen Algorithmen, die ohne Lernen bzw. Trainieren auskommen, hat sich die TUM auch mit dem Ansatz beschäftigt, klassische neuronale Netzwerke (Engl. artificial neural networks, ANN) zu trainieren und diese in gepulste neuronale Netzwerke (SNNs) zu konvertieren. Zuerst untersuchte die TUM die Anwendung von komplexwertigen Gewichten in ANNs, da die Vorverarbeitungsschritte der Radardatenverarbeitung in der Regel auf komplexe Matrizen basiert (siehe vorherige Paragraphen). Insbesondere wurde die Anwendung komplexwertiger ANNs für die Bestimmung des Ankunftsrichtung (Engl. angle of arrival, AoA) untersucht. Hierbei wurden Netzwerke trainiert, die den Objektwinkel aus der komplexen Entfernung-Geschwindigkeit Karte bestimmen. Durch die Untersuchung wurde festgestellt, dass komplexwertige Netzwerke eine geringere Fehleranfälligkeit aufweisen, als realwertige Netzwerke.

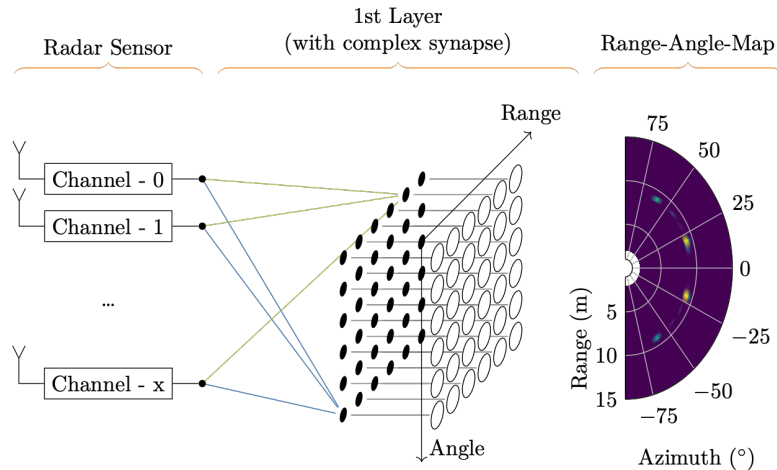


Abbildung 9 Architektur des Strahlformung Resonate-and-Fire Netzwerks. Beispielhaft sind die Empfangsantennen (Channel) mit komplexen Synapsen dargestellt, sowie die Resonate-and-Fire Neuronen als Pixel der Entfernung-Winkel Karte.

Da in diesem Projekt die Effizienz der Algorithmen eine entscheidende Rolle spielt, hat sich die TUM für eine zeitbasierte Konvertierung von ANNs zu SNNs entschieden. Hierbei konnten die Stärken des in Kapitel 2.4.1 vorgestellten Neuronenmodells genutzt werden. Das Neuronenmodell kann mit einem einzigen Puls einen Wert kodieren und somit die Anzahl der Pulse minimieren. Die Konvertierung wurde mit dem MNIST Datensatz ausgewertet und auf der Cognitive and Computational Neuroscience Konferenz in San Francisco vorgestellt. Die TUM erreichte mit dem Neuronenmodell eine durchschnittliche Pulszahl pro Neuron von 0.142 [4].

Gepulste OS-CFAR Der letzte Schritt der Signaldatenverarbeitung ist die Spitzenwerterkennung, die Aufschluss darüber gibt, ob ein Objekt vorhanden ist oder nicht. Daher wird das Ergebnis oft als Maske bestehend aus 0 und 1 dargestellt. In Radaranwendungen wird hierfür die konstante Falschalarmrate (constant false alarm rate, CFAR) Erkennung verwendet. Der ordered-statistic (OS) CFAR Algorithmus ist eine vielversprechende Variante der Spitzenwerterkennung, da hierbei erneut die effiziente Phasenkodierung bzw. TTFS Kodierung aus Kapitel 2.4.1 verwendet werden kann. Der OS-CFAR Algorithmus deklariert einen Wert als Objekt wenn dieser Wert größer ist, also der k -te Wert aus einem Vergleichsvektor bestehend aus benachbarten Werten. Die gepulste Version des Algorithmus trifft diese Entscheidung, wenn k Pulse vor dem Puls des zu klassifizierenden Wertes auftreten. Der Algorithmus wurde in [2] veröffentlicht und in [5] mit der gepulsten Variante des cell-averaging (CA) CFAR Algorithmus verglichen. Der gepulste OS-CFAR Algorithmus benötigt weniger Zeitschritte bei einer höheren Präzision als die CA-CFAR Variante [5]. Darüber hinaus ermöglicht die Kodierung des OS-CFAR Algorithmus, die direkte Anbindung an die gepulste Fouriertransformation.

2.4.2 Objektdatenverarbeitung

Die Aufgaben, welche nach der Signaldatenverarbeitung ausgeführt werden können sind vielfältig und reichen von einer bloßen Objekterkennung durch Gruppierung von einzelnen Datenpunkten über die Klassifizierung dieser Punkte als Objekt bis hin zu der Verfolgung des Objekts über mehrere Zeitschritte hinweg. Innerhalb des KI-ASIC Projekts hat die TUM sich auf die beiden Aufgaben der Gruppierung von Datenpunkten sowie der Erkennung und Verfolgung von Objekten fokussiert. Ein Hauptgrund für diese Entscheidung war die größere Flexibilität in der Entwicklung aufgrund der nicht vorhandenen Abhängigkeit von großen, gekennzeichneten Datensets, wie es bspw. bei einer Klassifizierung der Fall wäre.

Gepulste Gruppierung von Datenpunkten Für das Gruppieren von Radardaten wurde ein Netzwerk basierend auf der Idee von Radial Basis Function (RBF) Neuronen [6–8] entwickelt. Aufgrund der zeitbasierten Kodierung innerhalb des Netzwerks hat dieser Ansatz das Potenzial äußerst effizient auf neuromor-

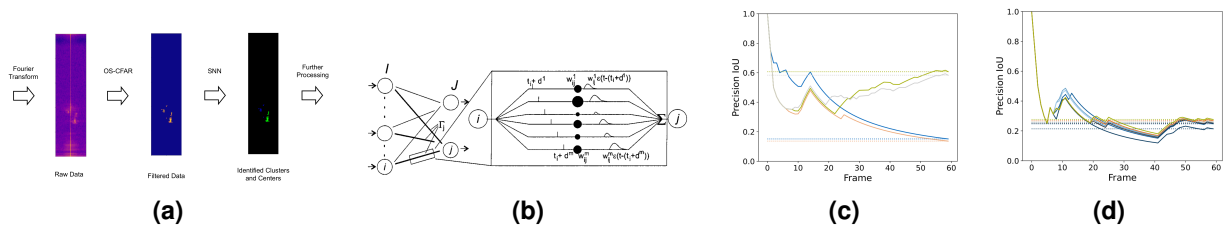


Abbildung 10 Eine Übersicht der Verarbeitungskette des RBF Netzwerks (a), der Aufbau des Netzwerks selbst (b) sowie die Evaluierungsergebnisse für die eigenständige (c) bzw. sequenzielle (d) Verarbeitung einer Karte.

pher Hardware zu laufen, denn n Pulse reichen hier aus um einen n -dimensionalen Datenpunkt zu kodieren. Das RBF Netzwerk lernt unüberwacht, d.h. ohne einen gekennzeichneten Datensatz. Als Eingangsdaten werden hier die nach der Fourier-Transformation und CFAR erhaltenen Entfernung-Geschwindigkeits Karten verwendet (siehe Abbildung 10a). Jeder 2-dimensionale Datenpunkt aus dieser Karte wird einzeln in das Netzwerk gegeben und mit zwei Pulsen kodiert (zeitkodiert).

Auch wenn das eigentliche Ziel das Gruppieren von Radardaten ist, so sollte im Zuge dieser Forschung auch die generelle Anwendbarkeit von RBF Netzen auf sequenzielle Daten untersucht werden. Normalerweise muss ein Datenpunkt in dieser Art Netzwerk mehrfach verarbeitet werden, damit die Gewichte "Zeit haben" sich anzupassen. Da die Frequenz eines Radargeräts im Automotivbereich allerdings sehr hoch ist können sich die Objekte zwischen zwei aufeinander folgenden Karten jedoch nicht weit bewegen und somit verändern sich diese Daten von einem Zeitschritt zum nächsten auch nur geringfügig. In den durchgeführten Experimenten hat sich dieser Umstand als nützlich erwiesen, um die Netze effizienter zu trainieren. Durch die geringe Veränderung zwischen zwei Zeitpunkten konnte, unter Verwendung der sequenziellen Daten, ein deutlicher Anstieg der Genauigkeit nach bereits wenigen Trainingsdurchläufen (< 4) ausgemacht werden. Dies ist in Abbildung 10 durch einen Vergleich des Lernerfolges über Zeit für das normale Lernverfahren (10c) sowie das sequenzielle Lernverfahren (10d) dargestellt. Das normale Lernverfahren, bei dem die Gewichte nach jedem Eingangsbild wieder zurückgesetzt werden, erzielt durchweg eine niedrigere Genauigkeit. Zusätzlich zu dieser niedrigen Genauigkeit in der Gruppierungsberechnung benötigt dieser Algorithmus zudem ca. 10 Sekunden für die Verarbeitung einer Karte in der aktuellen Computersimulation. Zwar ließe sich diese Zeit durch eine neuromorphe Implementierung um den Faktor 5-10 reduzieren, die geringe Genauigkeit bliebe jedoch bestehen. Die TUM hat sich daher entschieden die Forschung an dieser Methode nicht weiter zu führen.

Gepulste Objekterkennung/-verfolgung Die Grundlage der Objektverfolgung bildet ein kontinuierliches Attraktor Netzwerk (Engl. continuous attractor network, kurz CANs). Diese Netzwerke sind dynamische Systeme, welche den Zustand eines Systems diskret abbilden und verfolgen, auch wenn sie zu einem Zeitpunkt keine Informationen über das System erhalten. Die Übergänge von einem Zustand zum nächsten sind kontinuierlich. Die Neuronen eines solchen Netzwerks können bspw. in einer Linie (eindimensional), einem Ring (eindimensional) oder einer Ebene (zweidimensional) angeordnet sein und sind üblicherweise mittels eines "mexican-hat" Verbindungsschemas miteinander verknüpft. Das bedeutet, dass Neuronen, welche sehr nah beieinander sind mit positiven Gewichten (excitatorisch) verbunden sind und Neuronen, welche weit voneinander entfernt sind, mit negativen Gewichten (inhibitorisch). Diese Netzwerke werden häufig in der theoretischen Neurowissenschaft eingesetzt um verschiedene Aufgaben bzw. Neuronengruppen des Gehirns zu modellieren, speziell für sogenannte Orts-, oder auch Gitterzellen in der Hippocampusformation [9, 10]. Da vermutet wird, dass diese Zellen die Aufgabe haben die eigene Po-

Tabelle 3 Übersicht der Leistungsmetriken zur Objektdatenverarbeitung

Netzwerk	Aufgabe	Eingangsdaten	# Neurone	# Synapsen	Durchlaufzeit
RBF	Gruppierung	range-doppler	30	1600	10 s/Bild
CAN	Verfolgung	range-angle	2025	4.100.625	10 ms/Bild

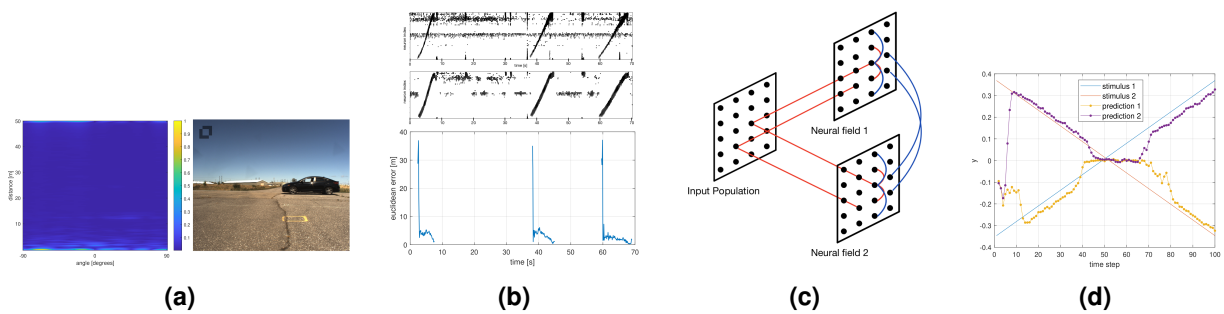


Abbildung 11 Die Eingangsdaten (Entfernung-Winkel Karte) des Objektverfolgung CANs (a) sowie der Verfolgungsfehler eines einzelnen Objekts mit diesen Daten (b). Die schematische Darstellung eines Netzwerks mit zwei eigenständigen 2D Attraktoren zur Verfolgung von mehreren Objekten (c), zusammen mit dem Resultat einer Objektverfolgung von zwei sich kreuzenden Objekten (d).

sition fortlaufend abzuschätzen, sehen wir diese Methodik als vielversprechend an um die Position eines anderen Objektes abzuschätzen.

Das von der TUM zur Objektverfolgung entwickelte CAN verwendet als Eingangsdaten die Entfernung-Winkel Karte, welche direkt von der Fourier-Transformation erzeugt wird - ohne vorherige Rauschminderung durch CFAR (siehe Abbildung 11a). Dies ist möglich, da das CAN intrinsisch eine Rauschminderung durchführt. Das Netzwerk ist darüber hinaus fähig ein Objekt zuverlässig zu erkennen und akkurat zu verfolgen. In einem Experiment mit 1985 Eingangsbildern konnte in 79 % aller Fälle das Objekt korrekt erkannt und mit einem durchschnittlichen Fehler von 3,22 m lokalisiert bzw. verfolgt werden (siehe Abbildung 11c). Im Vergleich zu dem zuvor vorgestellten Ansatz zur Objektdatengruppierung benötigt dieses Netzwerk jedoch deutlich mehr Neurone und Synapsen (siehe Tabelle 3). Der Ansatz ist hingegen deutlich effektiver und auch schneller, da hier eine gesamte Entfernung-Winkel Karte auf einmal verarbeitet wird. Die Implementierung des Netzwerks auf neuromorpher Hardware wird im nachfolgenden Kapitel (AP5) besprochen.

Da aufgrund der inherenten Gewichtsstruktur innerhalb eines CANs meist jedoch nicht mehr als ein Objekt verfolgt werden kann, haben wir noch Möglichkeiten zur Verfolgung von mehreren Objekten untersucht. Eine Implementierung mit zwei CANs, welche sich gegenseitig inhibieren, schafft es auch zwei Objekte gleichzeitig zu verfolgen und diese sogar weiter zu verfolgen, nachdem sie sich gekreuzt haben. Eine schematische Darstellung dieser Netzwerke ist in Abbildung 11d zu sehen. In einem Experiment mit 6765 Entfernung-Winkel Bildern auf denen sich teilweise zwei Objekte befinden war es möglich in ungefähr 75 % aller Bilder die Objekte korrekt zu erkennen und zu verfolgen. Eine Schwierigkeit, welche bei diesem Ansatz auftritt ist das Assoziationsproblem. Nachdem zwei Objekte sich kreuzen oder nah beieinander sind können die Netzwerke ihre Zuordnung verlieren. Dieses Prinzip ist in Abbildung 11b dargestellt. Hier kreuzen sich zwei Objekte und danach werden sie vom jeweils anderen Netzwerk verfolgt.

Aufgrund der dennoch vielversprechenden Ergebnisse mit diesem Netzwerk plant die TUM auch weiterhin an diesem Ansatz zu arbeiten. Zur Erforschung weiterführender Ansätze im Bereich der CANs hat die TUM darüber hinaus daran gearbeitet neuartige Attraktor Netzwerke mit mehreren Skalen, basierend auf neusten neurobiologischen Erkenntnissen, zu evaluieren und weiter zu entwickeln [11]. Zudem wurde bereits ein Reinforcement Learning Ansatz entwickelt, um ein Objekt (eine Person) mit einem Roboter zu verfolgen, basierend auf den ausgegebenen Daten des hier vorgestellten CANs. Sobald die TUM ein vollwertiges SpiNNaker 2 Board zur Verfügung gestellt bekommt, werden diese Ansätze kombiniert und in einem ersten Echtweltexperiment getestet.

2.4.3 Neuromorphe Radarverarbeitungsketten

Basierend auf den bisher vorgestellten Ansätzen zur Signal- bzw. Objektdatenverarbeitung ist es nun möglich zwei eigenständige neuromorphe Radarverarbeitungsketten aufzubauen - eine Zeitkodierte und eine Ratenkodierte. Beide Radarverarbeitungsketten sind mit den dazugehörigen Algorithmen an der entsprechenden Stelle der Kette in Abbildung 12 dargestellt. Die zeitkodierte Kette besteht ausschließlich aus

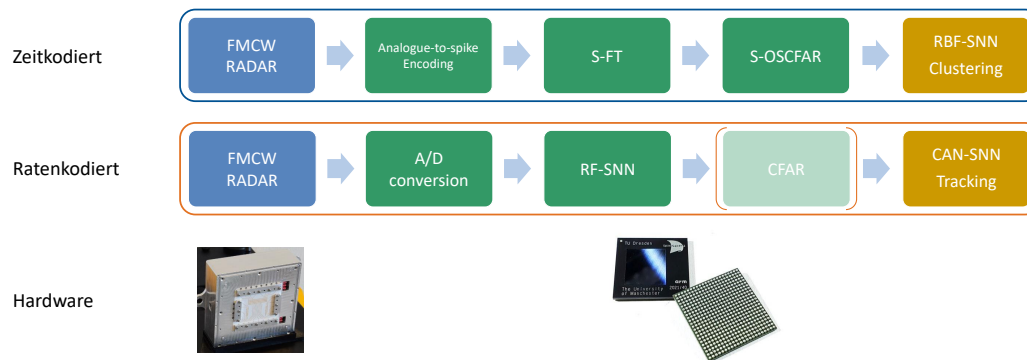


Abbildung 12 Die entwickelten neuromorphen Ansätze zur Radarverarbeitung, aufgeteilt in zwei eigenständige Verarbeitungsketten – zeitkodierte (blau) und ratenkodierte (orange).

Netzwerken, welche sowohl als Eingangs- als auch als Ausgangswerte zeitkodierte Pulse verwenden. Somit sind alle Ansätze kompatibel. Gleiches gilt für die Ansätze in der ratenkodierten Kette, welche alle nur ratenkodierte Werte verwenden. Für beide Verarbeitungsketten gilt, dass jeweils zwischen den einzelnen Netzwerken gegebenenfalls eine Transformation der Werte in einen anderen Raum stattfinden muss. Die Verarbeitungsketten wurden bislang in dieser Form nicht evaluiert, da die Implementierung der Netzwerke teils in unterschiedlichen Systemen bzw. Sprachen stattgefunden hat. Einzelne Teile der Kette sind jedoch gemeinsam getestet worden. Die Ergebnisse dieser Integration bzw. Evaluation werden im nächsten Kapitel (AP5) beschrieben.

2.5 Arbeitspaket 5

Das AP5 umfasste die Prüfung und Bewertung der in AP4 eingeführten gepulsten Netzwerke. Diese Netze wurden separat evaluiert und mit herkömmlichen Algorithmen verglichen. In AP5 wurden dann in einem letzten Schritt ein paar ausgewählte Netzwerke auf neuromorpher Hardware implementiert und evaluiert. Ein Teilabschnitt der zeitbasierten Radarverarbeitungskette (siehe Abbildung 12) wurde zudem gemeinsam auf dem Demonstrationsfahrzeug integriert und evaluiert.

Der zeitbasierte S-FT-Algorithmus wurde im Laufe des Projekts zunächst auf dem neuromorphen Chip Loihi 1 der Intel GmbH implementiert. Für die Evaluation des Netzwerkes wurde ein Datensatz der TUD verwendet, welcher mit einem FMCW-Radar von Texas Instruments (AWR1642Boost-ODS) aufgenommen wurde. Er beinhaltet 5 verschiedene Straßenszenarien die unterschiedliche Herausforderungen darstellen: zwei nahe beieinander liegende Ziele, weit entfernte Ziele, eine Kombination aus einem nahen und einem weit entfernten Ziel, mehrere Ziele vor dem Sensor und ein bewegliches Ziel. Die ersten vier Szenarien wurden in einer Dimension (Entfernung) evaluiert, das letzte Szenario wiederum in zwei Dimensionen (Entfernung und Geschwindigkeit). Die Implementierung erfolgte auf einem Loihi 1 Chip und die Eingabedaten wurden im Stapelverfahren verarbeitet, d.h. vor der Ausführung des Algorithmus wurde ein Stapel von Daten in Puls-Zeiten konvertiert und anschließend von der S-FT in Loihi verarbeitet. Für die Evaluation wurde die Wurzel der mittleren Fehlerquadratsumme (Engl. root mean square error, RMSE) für die verschiedenen Szenarien gemessen. Anschließend wurde die Leistung des Algorithmus anhand von zwei verschiedenen Parametern gemessen: der Simulationslänge und der FFT-Mode-Größe. Die Ergebnisse in Abbildung 14a zeigen, dass die Leistung der S-FT sich exponentiell mit der Simulationslänge verbessert, bei einer gleichbleibenden Distanzauflösung.

Als Teil der Evaluierung wurde außerdem der Fehler des Frequenzspektrums mit dem eines FFT-Beschleunigers verglichen. Der Vergleich, der in der Tabelle 4 dargestellt ist, ist für jedes Szenario individuell, unter Verwendung einer Konfiguration mit 256 Simulationsschritten und 256 Moden. Die Tabelle zeigt die Leistung für eine DFT- und eine FFT-gepulste-Architektur, sowie die Ausgabe des FFT-Beschleunigers. Als Referenz ist zudem die Ausgabe einer softwarebasierten FFT unter Verwendung der Python-Bibliothek NumPy angegeben.

Tabelle 4 Der RMSE der DFT und FFT Architekturen der S-FT Netzwerke zusammen mit den Ergebnissen eines FFT Beschleunigers als Vergleich.

Architecture	S1	S2	S3	S4
S-DFT	0.004	0.041	0.009	0.030
S-FFT	0.006	0.026	0.007	0.028
Accelerator	0.0005	0.0005	0.0005	0.0005

Darüber hinaus wurde für dieses Experiment der Energieverbrauch sowie die Latenzzeit der S-FT auf der Grundlage der von Intel für das Loihi-Board bereitgestellten Leistungsparameter berechnet. Diese Werte hängen von der Anzahl der Simulationsschritte ab, der Anzahl der Neuronen in der S-FT (n_{neurons}) sowie der Anzahl der Puls-Operationen ($n_{\text{spike ops.}}$). Die Leistungsergebnisse der DFT- und FFT-Architekturen sind in der Tabelle 5) zusammengefasst. Das Modell, seine Implementierung in Loihi sowie die entsprechende Evaluation in realen Radar-Szenarien wurde 2022 in einer Fachzeitschrift veröffentlicht [3].

Tabelle 5 Die theoretische Effizienz der S-DFT und S-FFT auf dem Loihi Board basierend auf den von Intel bereitgestellten Leistungsparametern der Hardware. Die Reihen stellen die Anzahl der Neuronen, die Anzahl der Puls-Operationen, die benötigte Energie zum Verarbeiten eines Chirps, die gesamte Zeit zum verarbeiten eines Chirps, die Latenzzeit zwischen zwei aufeinanderfolgenden Chirps und die verbrauchte Energie pro Chirp dar.

Parameter	S-DFT	S-FFT
n_{neurons}	2048	10240
$n_{\text{spike ops. (thsnd.)}}$	2100	84
$E(\mu\text{J})$	65.5	49.9
$T_f(\mu\text{s})$	77.6	105
$\tau_f(\mu\text{s})$	77.6	315
$P(\text{mW})$	844	158

Um den S-OSCFAR-Algorithmus zu bewerten hat die TUM zunächst qualitative Experimente mit simulierten Daten aus dem von der Infineon AG zur Verfügung gestellten Simulator - dem Black-Box-Modell (BBM) - durchgeführt. Die Ergebnisse zeigen, dass der S-OSCFAR-Algorithmus das Verhalten des herkömmlichen Algorithmus für ein-, bzw. zweidimensionale FFT-Spektren replizieren kann. Diese Ergebnisse wurden im Jahr 2021 veröffentlicht [2]. Im Anschluss an diese Publikation wurden umfangreiche Experimente mit Daten aus dem CARRADA-Datensatz durchgeführt und die Ergebnisse des S-OSCFAR Algorithmus mit denen des von der TUD entwickelten gepulsten CA-CFAR verglichen. Die Ergebnisse dieser Evaluation, veröffentlicht in [12], zeigen die Leistung beider SNNs für eine unterschiedliche Anzahl von Zeitschritten. In Abbildung 13 ist zu sehen, dass die S-OSCFAR in einer Simulation mit mehr als 100 Zeitschritten eine Sensitivität und Genauigkeit >99 % aufweisen kann.

Als Teil des finalen Evaluierungsprozesses, hat die TUM abschließend die S-FT und die S-OSCFAR in den SpiNNaker-Chip der TUD integriert. Dieser Teil der Signalverarbeitungskette wurde dann mit Radardaten aus dem von der OTH bereitgestellten Datensatz evaluiert. Die Ergebnisse zeigen, dass sowohl die S-FT als auch die S-OSCFAR eine ähnliche Präzision wie in den vorherigen Experimenten erreichen können. Dies zeigt, dass der SpiNNaker 2 Chip keine zusätzlichen Fehler oder ein Rauschen bei der Ausführung der vorgestellten Netzwerke verursacht. Aufgrund der eingeschränkten Größe des SpiNNaker 2 FPGA und der damit einhergehenden Grenzen bezüglich der Netzwerkgröße wurden die Algorithmen in zwei separaten Experimenten ausgeführt. Die S-FT wurde mit 1D-Daten (256 Abtastwerte) getestet und erzeugte ein Frequenzspektrum mit 128 Moden, welches in Abbildung 14b dargestellt ist. Die S-OSCFAR wiederum wurde mit 2D-Daten, welche aus Entfernung-Geschwindigkeits Karten mit einer Größe von 64x16 Abtastwerten bestanden, getestet. Diese Tests führten zu Pulsmatrizen derselben Größe als Ausgangswerte des Netzwerks.

Um aus den Radardaten des mmIC unmittelbar Pulse für SNNs generieren zu können hat die TUM abschließend noch einen Prototypen des Analog-Puls-Kodierers (Engl. analog-spike-encoder, ASE) als

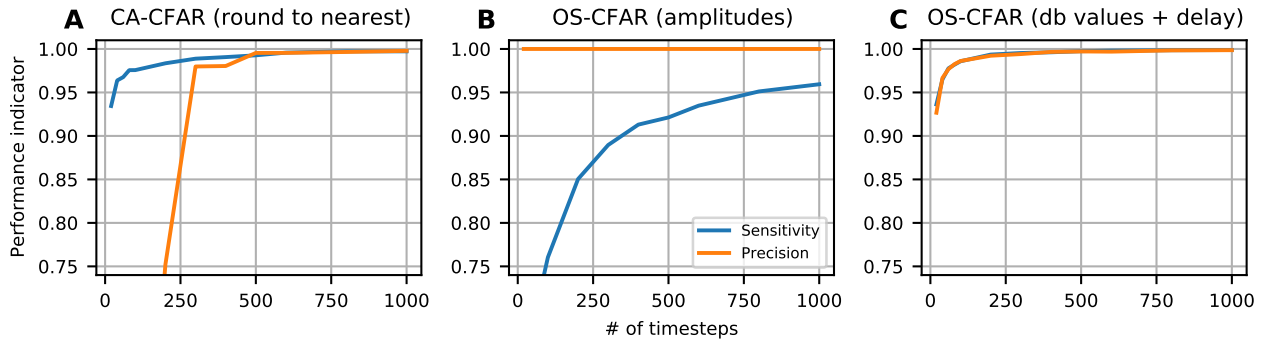


Abbildung 13 Die Sensitivität (blau) und Genauigkeit (orange) von mehreren gepulsten CFAR Ansätzen evaluiert auf 1000 Beispielen aus dem CARRADA Datensatz. Links ist das Ergebnis der S-CACFAR der TUD dargestellt, mittig das Ergebnis der Standardimplementierung der S-OSCFAR der TUM und rechts das Ergebnis der S-OSCFAR angewandt auf logarithmische Eingangsdaten kombiniert mit einer Verzögerung zu benachbarten Zellen von einem halben Zeitschritt.

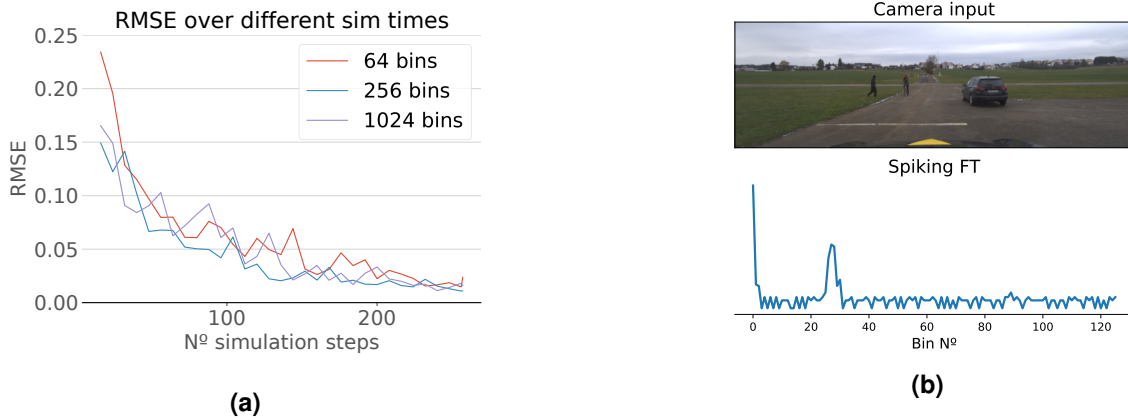
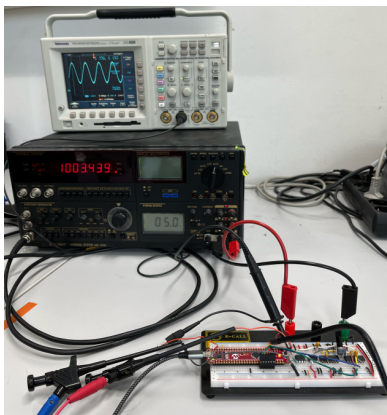


Abbildung 14 Der RMSE der S-FT als eine Funktion der Simulationsschritte für verschiedene Distanzauflösungen (a) und ein Beispielresultat der S-FT auf dem SpiNNaker 2 FPGA (unten) mit einer Verkehrsszene (oben) deren zugehörige Radardaten als Eingang verwendet wurden (b).

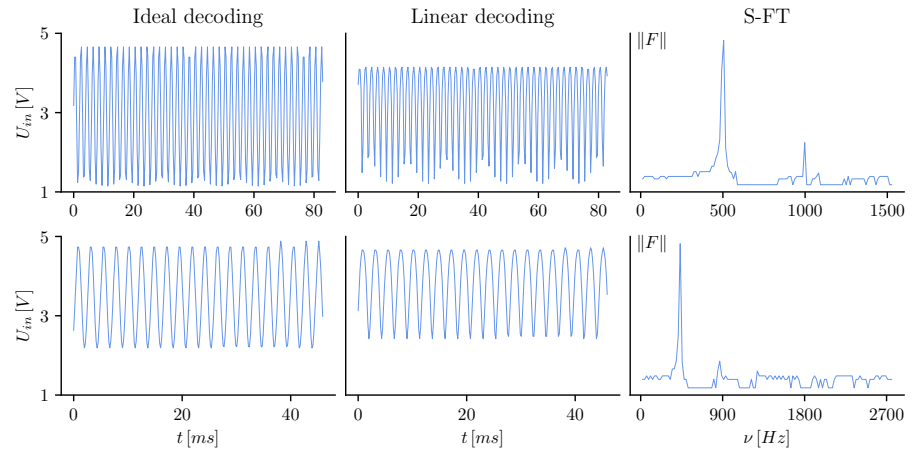
reale Schaltung entwickelt und diesen in einem Elektroniklabor der TUM getestet (siehe Abbildung 15a). Die Schaltung wurde mit Signalen von einem Funktionsgenerator getestet. Für die Steuerung der Schaltung und das Sammeln der Pulse wurde darüber hinaus ein dsPIC-Mikrocontroller programmiert.

Für die erzeugten Pulse wurde dann im Rahmen des ASE Evaluation der Dekodierungsfehler in Form des mittleren absoluten Fehlers (MAE) gemessen. Der MAE blieb unter 0,1 für Eingangssignale im Bereich von 1 bis 5 V und für Schwellenspannungen $u_{th} \in [0.1, 0.5, 0.75, 0.9]$. Darüber hinaus wurden die Ergebnisse des ASE mit der S-FT-Implementierung auf dem SpiNNaker 2 FPGA getestet. Abbildung 15b zeigt das Ergebnis der S-FT für einen Eingang aus einer einzelnen Welle von 500 Hz mit zwei verschiedenen Werten für die minimale Eingangsspannung $U_{in} \in [1, 2]$. Zum Zeitpunkt der Experimente war es noch nicht möglich den ASE an die SpiNNaker 2 Platine anzuschließen. Der ASE benötigt eine Low-Level-Schnittstelle (z.B. I2C, SPI), welche aktuell noch nicht in der SpiNNaker 2 Platine integriert ist. Dies betrifft auch weitere neuromorphe Platinen zu denen die TUM zum Zeitpunkt der Experimente Zugang hatte (SpiNNaker 1, Loihi 1 & 2).

Der biologisch inspirierte Ansatz zur Objektverfolgung mittels eines kontinuierlichen Attraktornetzwerks, welcher in AP4 entwickelt wurde, ist im Rahmen des AP5 auch auf einem SpiNNaker 1 Board evaluiert worden. Dieses Board stand der TUM aus einem vorangegangenen Projekt noch zur Verfügung und verfügt außerdem über die nötige Größe, um das CAN vollständig darauf zu simulieren. Das Netzwerk besteht, wie zuvor beschrieben, aus 2.025 Neuronen, welche vollständig mit Hilfe von mehr als 4 Millionen Syn-



(a)



(b)

Abbildung 15 Der Aufbau des Analog-Puls-Kodierers im Labor (a) und das Resultat des ASE für ein Eingangssignal mit 500 Hz (b).

apsen verbunden sind. Auf dem von der TUD für den Demonstrator bereitgestellten SpiNNaker 2 FPGA konnten zum Zeitpunkt der Evaluationsphase weniger als 1.000 Neurone und 100.000 Synapsen simuliert werden. Eine Evaluation auf diesem System war daher innerhalb des Projektrahmens ausgeschlossen. Durch die Evaluation auf dem SpiNNaker 1 Board konnten wir jedoch zeigen, dass der hier vorgestellte Ansatz das Objekt in 79 % aller Fälle (1565/2087 Bildern) korrekt erkennt und hierbei einen Durchschnittsfehler von 3,22m erreicht. Das Netzwerk kann darüber hinaus die Daten bereits vor der Rauschminderung durch den CFAR Algorithmus verarbeiten und reduziert somit die Komplexität der Radarverarbeitungskette um einen Schritt. Durch die zeitgleiche Verarbeitung einer gesamten Entfernung-Winkel Karte erreicht das Netzwerk zudem eine Verarbeitungsrate von 100 Bildern pro Sekunde und ist somit echtzeitfähig. Wir planen diese Ergebnisse im Jahr 2024 noch auf der mobilen Roboterplattform der TUM zu evaluieren, sobald eine ausreichend große SpiNNaker 2 Platine verfügbar ist.

3 Weiterführende Informationen

3.1 Vorhaben, Ergebnisse & Zuwendungsverwendung

Tabelle 6 Übersicht der gesteckten Ziele, der erreichten Ergebnisse sowie den zugehörigen Verwendungen.

AP	Geplantes Ergebnis	Art der Zuwendung	Erzieltes Ergebnis
AP1	Definition von Anforderungen an Signalverarbeitung und Spezifikation von Schnittstellen/Datenformaten	Personalkosten	Ausgearbeitete Spezifikations- und Anforderungslisten der neuromorphen Algorithmen und Datenverarbeitungsschritte. Evaluation von Datenformate mit Festlegung auf das HDF Format. Software zur effizienten Handhabung der verschiedenen Datensätze, bspw. Download, Datenformatierung, etc.
AP2	Architekturentwicklung für Gesamtsystem, Objekt- und Signaldatenverarbeitung	Personalkosten	Definition einer Architektur für das Gesamtsystem im Konsortium. Entwicklung einer Softwarearchitektur (Pipeline) für die gesamte Radarverarbeitungskette (Signal- & Objektdaten). Veröffentlichung der Pipeline inkl. Standardalgorithmen zur Radarverarbeitung (FT, CFAR, etc.) und gepulstem Beispiel (S-FT).

AP3	Ausbau und Betrieb eines lokalen Testsystems	Personalkosten, Gegenstände	Beschaffung, Aufbau und Inbetriebnahme einer mobilen Roboterplattform für die Evaluation von gepulsten Netzen in einer Echtzeitumgebung. Integration des FMCW Radarsystems von Infineon.
AP4	Neuromorphe Algorithmen für Signal- und Objektdatenverarbeitung	Personalkosten, Dienstreisen	Erfolgreiche Entwicklung von zwei vollständig funktionalen, gepulsten Radarverarbeitungsketten (zeitbasiert & ratenbasiert). Veröffentlichung der Ergebnisse auf Konferenzen und in Fachzeitschriften.
AP5	Evaluation und Demonstration der entwickelten Algorithmen	Personalkosten, Gegenstände	Integration/Evaluation ausgewählter gepulster Netzwerke in Demonstrator (SpiNNaker 2 FPGA) sowie finale Demonstration am Projektende. Veröffentlichungen zu den durchgeführten Experimenten auf Konferenzen und in Fachzeitschriften.

3.2 Wichtige Positionen des zahlenmäßigen Nachweises

Tabelle 7 Übersicht der einzelnen Budgetpositionen, ihrer Beschreibungen und Verwendungen im Projekt.

Position	Benennung im Antrag (AZK/AZA)	Verwendung
0812	Beschäftigte E12-E15	Personalkosten – Wissenschaftliche Mitarbeiter
0822	Beschäftigungsentgelte	Nicht verwendet. Verwendung der Mittel für Server (0850) benötigt für Training & Evaluation.
0846	Dienstreisen	Teilnahmen an Projekttreffen, Veranstaltungen und Konferenzen
0850	Gegenstände über 800 €	Hardware – Großgeräte: <ul style="list-style-type: none"> • 3 x lokaler Server für Training & Evaluation • mobile Roboter Plattform Hardware – Kleingeräte: <ul style="list-style-type: none"> • zusätzliche Hardware für mobilen Roboter • Platine/Hardware für Analog-zu-Puls Konverter

3.3 Notwendigkeit & Angemessenheit der Projektarbeiten

In dem Projekt KI-ASIC wurde durch die TUM, in Zusammenarbeit mit den Partnern aus Industrie und Forschung, ein wichtiger Schritt in Richtung der Erforschung neuer, alternativer Rechnerarchitekturen für die KI der Zukunft erzielt. Eine der Kernaufgaben heutiger künstlicher neuronaler Netzwerke ist die Verarbeitung von Sensordaten, bspw. im automobilen Umfeld. Die Entwicklung von neuromorphen Verarbeitungsketten für Sensordaten ist eine essentielle Herausforderung auf dem Weg zu einer vollständig neuromorphen Sensordatenverarbeitung. Durch die Komplexität der Aufgabenstellung bestanden erhebliche technische und wirtschaftliche Risiken hinsichtlich des Erreichens der Projektziele. Das risikofolle und ambitionierte Projekt hätte die wirtschaftlichen Möglichkeiten der TUM als staatliche Bildungseinrichtung weit überfordert. Speziell aufgrund des fachübergreifenden Charakters dieses Projekts – Entwicklung von neuromorpher Hardware (TUD), Radartechnologie (Infineon AG) und neuromorphen Algorithmen (TUM), sowie Bereitstellung von einem Erprobungsträger (BMW AG) und der Aufnahme von Sensordaten (OTH). Die Realisierbarkeit eines Vorhabens dieser Größenordnung selbständig zu finanzieren, organisieren und durchzuführen wäre somit undenkbar. Nur durch die Förderung der Bundesrepublik konnte eine erfolgreiche Umsetzung des Vorhabens durch das gesamte Konsortium sichergestellt werden.

3.4 Nutzen & Verwertbarkeit des Ergebnisses

Tabelle 8 Übersicht des voraussichtlichen Nutzens sowie der Verwertbarkeit der TUM Ergebnisse im Sinne des fortgeschriebenen Verwertungsplans.

Projektergebnis/Inhalt	Nutzen/Verwendung
Entwicklung einer Softwarearchitektur (Pipeline) zur Verarbeitung von Radardaten, inklusive herkömmlicher Algorithmen zur Radarverarbeitungskette	Es ist eine Veröffentlichung der Softwarearchitektur mit den dazugehörigen Standardalgorithmen zur Radarverarbeitung geplant. Diese dient nicht nur der TUM als Basis für zukünftige Radarprojekte, sondern ermöglicht auch Radar-unerfahrenen Entwicklern einen einfachen Einstieg in das Thema.
Entwicklung einer Schaltung zur Analog-zu-Puls Konvertierung	Basierend auf den vielversprechenden Erkenntnissen und Ergebnissen aus diesem Forschungsblock wird die TUM einen Antrag für ein Folgeprojekt zum Thema neuromorphe Sensorik (Radar) stellen. Das Ziel dieses Projektes wird es sein, für die im Projekt KI-ASIC entwickelten Methoden und Netzwerke für die Signaldatenverarbeitung nun passende Hardware zu designen um einen Vergleich zum Stand der Technik zu ermöglichen. Sämtlicher Code wurde oder wird noch veröffentlicht. Außerdem wird, wie in Kapitel 2.4 angedeutet, noch eine gemeinsame Arbeit veröffentlicht werden, welcher mehrere der vorgestellten Netzwerke zu einer voll neuromorphen Radarverarbeitungskette kombiniert
Entwicklung von SNNs für die Signaldatenverarbeitung von Radardaten	
Entwicklung von SNNs für die Objektdatenverarbeitung von Radardaten	Die Erkenntnisse und Ergebnisse aus dieser Forschung fließen bereits in ein Nachfolgeprojekt (EBRAIN) in Kooperation mit der Universität Linz ein. Sämtlicher Code wurde oder wird noch veröffentlicht. Außerdem wird, wie in Kapitel 2.4 angedeutet, noch eine gemeinsame Arbeit veröffentlicht werden, welcher mehrere der vorgestellten Netzwerke zu einer voll neuromorphen Radarverarbeitungskette kombiniert.
Anschaffung, Aufbau, Inbetriebnahme und Verwendung von mobiler Roboterplattform (inklusive FMCW Radar)	Die mobile Roboterplattform wird bereits in Universitären Projekten mit Studenten verwendet. Sie wird auch die Basis der gemeinsamen Arbeit an einem neuromorphen, personenfolgenden Roboter sein (siehe Kapitel 3.6.2). Des weiteren ist sie auch Teil von Folgeprojekten in der Forschung, bspw. dem Projekt EBRAIN mit der Universität Linz.

3.5 Fortschritt auf dem Gebiet bei anderen Stellen

Während der Projektdurchführung wurden keine relevanten Ergebnisse Dritter bekannt.

3.6 Veröffentlichungen der TUM

3.6.1 Erfolgte Veröffentlichungen

¹J. López-Randulfe et al., "Spiking Neural Network for Fourier Transform and Object Detection for Automotive Radar", *Frontiers in Neurorobotics* 15 (2021).

²J. Lopez-Randulfe et al., "Time-Coded Spiking Fourier Transform in Neuromorphic Hardware", *IEEE Transactions on Computers*, 1 (2022).

³J. López-Randulfe, N. Reeb, and A. Knoll, "Conversion of convnets to spiking neural networks with less than one spike per neuron", in *2022 conference on cognitive computational neuroscience* (2022), pp. 553–555.

⁴R. Dietrich et al., *Grid Codes versus Multi-Scale, Multi-Field Place Codes for Space*, (Aug. 11, 2023) <https://www.biorxiv.org/content/10.1101/2023.06.18.545252v2> (visited on 08/18/2023), preprint.

⁵J. Lopez-Randulfe, N. Reeb, and A. Knoll, "Integrate-and-fire circuit for converting analog signals to spikes using phase encoding*", Neuromorphic Computing and Engineering 3, 044002 (2023).

3.6.2 Geplante Veröffentlichungen

Tabelle 9 Übersicht der potenziellen Verwertung der Ergebnisse der TUM

Titel	Zeitraumen	Inhalt
A Fully Neuromorphic Person Following Robot with FMCW Radar Data	06/2024	Eine gemeinsame Arbeit, welche sowohl die Signaldatenverarbeitung als auch die Objektdatenverarbeitung verknüpft. Das Ziel ist, den Roboter einer Person folgen zu lassen. Die Radardaten werden auf dem neuromorphen Chip mit den in AP4 beschriebenen Netzen vorverarbeitet. Die Person wird anschließend mit dem Objektverfolgungs CAN aus AP4 erkannt und verfolgt. Der Roboter folgt der Person dann durch ein R-STDP Netzwerk, welches kurz in AP4 beschrieben wurde. Dies wird ein erstes Echtzeit Beispiel einer vollständig neuromorphen Radardatenverarbeitung.
Beamforming Resonate-and-Fire Network for Radar Data	12/2023	Der in 2.4 beschriebene Ansatz Resonate-and-Fire Neuronen und eine Beamforming Matrix zu verwenden, wird aktuell evaluiert. Die Veröffentlichung soll die Vorteile einer kontinuierlichen und parallelen Verarbeitung der Radardaten aufzeigen und somit die Stärken von neuromorphen Algorithmen unterstreichen.

Literatur

- ¹J. Lopez-Randulfe, N. Reeb, and A. Knoll, "Integrate-and-fire circuit for converting analog signals to spikes using phase encoding*", Neuromorphic Computing and Engineering 3, 044002 (2023).
- ²J. López-Randulfe et al., "Spiking neural network for fourier transform and object detection for automotive radar", Frontiers in Neurorobotics 15, 10.3389/fnbot.2021.688344 (2021).
- ³J. López-Randulfe et al., "Time-coded spiking fourier transform in neuromorphic hardware", IEEE Transactions on Computers 71, 2792 (2022).
- ⁴J. López-Randulfe, N. Reeb, and A. Knoll, "Conversion of convnets to spiking neural networks with less than one spike per neuron", in 2022 conference on cognitive computational neuroscience (2022), pp. 553–555.
- ⁵B. Vogginger et al., "Automotive radar processing with spiking neural networks: concepts and challenges", Frontiers in Neuroscience 16, 10.3389/fnins.2022.851774 (2022).
- ⁶J. J. Hopfield, "Pattern recognition computation using action potential timing for stimulus representation", Nature 376, 33 (1995).
- ⁷T. Natschläger and B. Ruf, "Spatial and temporal pattern analysis via spiking neurons", Network: Computation in Neural Systems 9, 319 (1998).
- ⁸S. M. Bohte, H. L. Poutre, and J. N. Kok, "Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer RBF networks", IEEE Transactions on Neural Networks 13, 426 (2002).
- ⁹K. J. Jeffery, "Place Cells, Grid Cells, Attractors, and Remapping", Neural Plasticity 2011, e182602 (2011).
- ¹⁰Y. Burak and I. R. Fiete, "Accurate Path Integration in Continuous Attractor Network Models of Grid Cells", PLOS Computational Biology 5, e1000291 (2009).
- ¹¹R. Dietrich et al., *Grid Codes versus Multi-Scale, Multi-Field Place Codes for Space*, (Aug. 11, 2023) <https://www.biorxiv.org/content/10.1101/2023.06.18.545252v2> (visited on 08/18/2023), preprint.
- ¹²B. Vogginger et al., "Automotive Radar Processing With Spiking Neural Networks: Concepts and Challenges", Frontiers in Neuroscience 16, 851774 (2022).