BACHELORARBEIT

Methoden zur Erkennung von Geräuschen und Sprache im Kontext altersgerechter Assistenzsysteme

Deep Learning zur Klassifizierung von Audiosignalen im Wohnumfeld älterer Menschen

vorgelegt am 15. April 2024

Kai-Michael Wolters

Erstprüferin: Prof. Dr. Larissa Putzar
Zweitprüfer: Prof. Dr. Roland Greule

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG
Department Medientechnik
Finkenau 35
22081 Hamburg

Zusammenfassung

Diese Bachelorarbeit befasst sich mit Methoden und Deep-Learning-Techniken zur Erkennung von Geräuschen und Sprache im Rahmen von altersgerechten Assistenzsystemen in Wohnbereichen. Mit einer zunehmenden alternden Bevölkerung steigt die Nachfrage nach innovativen Lösungen, die die Lebensqualität und Sicherheit von Senioren in ihren Wohnräumen verbessern. Assistenzsysteme können dabei helfen festzustellen, ob ältere Menschen in ihrem Wohnumfeld in einer Gefahrensituation oder Notlage sind und Hilfe benötigen.

Die Arbeit beginnt mit der Untersuchung wissenschaftlicher und technischer Grundlagen, die im Zusammenhang mit der Audiosignalerfassung und -verarbeitung im Kontext der Anwendung von Deep-Learning-Techniken Anwendung finden. Es werden verschiedene Mikrofontechniken verglichen als auch die digitale Signalverarbeitung von Audioaufnahmen beschrieben. Anschließend werden die theoretischen Grundlagen des Deep Learnings untersucht und die Prinzipien hinter neuronalen Netzen erläutert. Besonderes Augenmerk liegt auf Neuronalen Faltungsnetzen (Convolutional Neuronal Networks CNNs), die sich gut für die Verarbeitung von Audiodaten eignen. Im weiteren Verlauf der Arbeit erfolgt die Konzeptionierung und Implementierung verschiedener Methoden zur Geräusch- und Spracherkennung. Eine Studie mit der Gegenüberstellung unterschiedlicher Parametrierungen und Deep-Learning-Architekturen vergleicht und bewertet die Leistungsfähigkeit bei der Klassifizierung aufgenommener Geräusche und Sprache.

Die Ergebnisse dieser Studie tragen zum Verständnis der Machbarkeit und Wirksamkeit von Deep-Learning-Ansätzen in Anwendungen von altersgerechten Assistenzsystemen bei. Die Auswirkungen dieser Ergebnisse werden im Kontext der Entwicklung praktischer Lösungen diskutiert. Insgesamt liefert diese Arbeit wertvolle Einblicke in die Integration von Deep-Learning-Methoden zur Geräusch- und Spracherkennung in altersgerechte Assistenzsysteme und ebnet den Weg für zukünftige Anwendungen, die darauf abzielen, die Autonomie und das Wohlbefinden älterer Menschen im täglichen Leben zu verbessern.

Abstract

This bachelor's thesis deals with methods and deep learning techniques for recognizing sounds and speech as part of Ambient Assisted Living systems (AAL) in living areas. With an increasing aging population, the demand for innovative solutions that improve seniors' quality of life and safety in their living spaces is increasing. Assistance systems can help determine whether older people are in a dangerous situation or emergency in their living environment and need help.

The work begins by examining scientific and technical principles applicable to audio signal capture and processing in the context of applying Deep Learning techniques. Various microphone technologies are compared, and the digital signal processing of audio recordings is described. The theoretical foundations of Deep Learning are then examined and the principles behind neural networks are explained. Particular attention is paid to Convolutional Neural Networks (CNNs), which are well suited for processing audio data. As the work progresses, various methods for noise and speech recognition will be designed and implemented. A study comparing different parameters and Deep Learning architectures compares and evaluates the performance in classifying recorded noise and speech utterances.

The results of this study contribute to the understanding of the feasibility and effectiveness of Deep Learning approaches in AAL system applications. The implications of these results are discussed in the context of developing practical solutions. Overall, this work provides valuable insights into the integration of Deep Learning methods for sound and speech recognition into AAL systems and paves the way for future applications aimed at improving the autonomy and well-being of older people in daily life.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Formelverzeichnis

Liste der Codeblöcke

1 Einleitung

1.1 Motivation

Methoden zur Erkennung und Unterscheidung von Geräuschen und Sprache haben seit Jahren Einzug in vielfältige Lebensbereiche genommen. Automatische Spracherkennungssysteme in Fahrzeugen, im Haushalt oder im Bereich von Telefonhotlines unterstützen den Menschen im Alltag und Optimieren und Beschleunigen Abläufe.

Die in diesem Zusammenhang eingesetzten Deep-Learning-Verfahren haben in den letzten Jahren die Qualität von Bild- und Spracherkennung signifikant verbessert und neue Anwendungsgebiete erschlossen. Das Interesse am Thema Deep Learning ist unter anderem durch Entwicklungen von Produkten gesteigert worden, die nur mit Hilfe von Deep-Learning-Technologien zur realisieren waren. Dem Aufkommen von Suchmaschinen und öffentlich zugänglichen Bild, Ton- und Videoarchiven ist es zu verdanken, dass eine ausreichende Datenbasis für das Training von Modellen zur Erkennung von Schriften, Sprache und Bildern geschaffen wurde. Parallel hat sich die Leistungsfähigkeit moderner Computerhardware stark gesteigert [1].

Diese Arbeit ist motiviert durch das Interesse, im Kontext altersgerechter Assistenzsysteme Methoden zu beschreiben und zu bewerten, die mittels Geräusch- und Spracherkennung im Wohnbereich älterer Menschen in der Lage sind, Gefahrensituationen und Bedarf zur Hilfe vom normalen Lebensalltag abzugrenzen und im Notfall oder auf Anforderung der älteren Person fremde Hilfe anfordert. Diese Methoden betrachten das Gebiet der Geräusch- und Sprachaufnahme mit verschiedenen Arten von Aufnahmequellen sowie den programmtechnischen Bereich der Signalverarbeitung, Extraktion und Klassifizierung von Audioaufnahmen mit Hilfe von Deep Learning.

1.2 Hintergrund

Die meisten Menschen in Deutschland leben im Alter allein im eigenen Zuhause. Im vergangenen Jahr lebten circa 4% der über 65-Jährigen in einer Pflegeeinrichtung, einem Altersheim oder einer ähnlichen Gemeinschaftsunterkunft. Pflegebedürftigkeit bedeutet keineswegs den

Verlust ihres eigenen Zuhauses: Fast 75% der Pflegebedürftigen ab 80 Jahren werden zu Hause versorgt, davon die Hälfte von ihnen überwiegend durch Angehörige [2].

In der häuslichen Versorgung ist die Zahl an professionellen Pflege- und Betreuungskräften zu niedrig, um den vorhandenen Bedarf decken zu können. Unterstützende Technologien, die unter dem Begriff Ambient Assisted Living (AAL) oder altersgerechte Assistenzsysteme zusammengefasst werden, haben das Potenzial, Lebenssituationen älterer Personen, die auf Pflege- und Betreuung angewiesen sind, sowie deren Angehörigen zu verbessern [3].

Altersgerechte Assistenzsysteme unterstützen die Bereiche Gesundheit (Telemedizin), häusliche Pflege (intelligentes Wohnen), Versorgung und Haushalt (automatische Bestellsysteme) und Sicherheit (Monitoringsysteme) [4].

1.3 Zielsetzung

Das grundlegende Ziel dieser Arbeit ist, Methoden zu finden und ihre Praxistauglichkeit zu bewerten, die auf Basis einer Audiosignalerfassung und nachfolgender Bearbeitung mit Hilfe von KI-gestützten Methoden wie Deep Learning Geräusche und Sprache im Wohnbereich älterer Menschen klassifizieren und entscheiden, ob eine Gefahrenlage für die ältere Person vorliegt. Angehörige oder bevollmächtigte Personen wie ambulante Pflegeeinrichtungen sollen aufgrund der Klassifizierung der Audiosignale als Assistenzsystem zum Beispiel in Form einer App-Anwendung eine Statusmeldung erhalten, die anzeigt, ob die ältere Person im Wohnbereich Hilfe benötigt oder ob von einer normalen Alltagssituation auszugehen ist. In diesem Zusammenhang werden am Ende dieser Arbeit als Ausblick weitere unterstützende Assistenzsysteme wie Vibrationssensoren, Bluetooth-Präsenzerkennung und Vitalüberwachung betrachtet, die die Zuverlässigkeit der KI-gestützten Entscheidung auf Basis der Audiosignalbearbeitung substanziieren.

Ein Schwerpunkt dieser Arbeit liegt in der Ermittlung geeigneter Hardwarelösungen sowie die Erfassung, Verarbeitung und Datenextraktion von Audioaufnahmen. Der zweite Schwerpunkt betrachtet Deep Learning mit Convolutional Neuronal Networks (CNN) und analysiert die Qualität der Klassifizierungen. Ein Ergebnis ist die Gegenüberstellung verschiedener CNN-Modelle und deren Parametrierung im Kontext der Geräusch- und Spracherkennung als ein Fundament, eine praxisreife Softwarelösung in Python zu programmieren.

Unternehmen, die altersgerechte Assistenzsysteme in Form einer Geräusch- und Spracherkennung im Wohnbereich älterer Menschen oder in vergleichbaren Bereichen wie Seniorenheimen, Einrichtungen für pflegebedürftige Personen oder Kliniken entwickeln wollen oder existierende Systemlösungen um eine Erfassung und Klassifizierung von Audiosignalen ergänzen möchten, erhalten mit dieser Arbeit substanzielle Informationen und Ergebnisse für eine Anwendung und Realisierung.

1.4 Aufbau der Arbeit

Das nachfolgende Kapitel 2 befasst sich ausführlich mit dem wissenschaftlichen Stand der Technik im Kontext der Audiosignalerfassung und -datenverarbeitung. Ein weiterer Schwerpunkt behandelt verschiedene Analysewerkzeuge von Audiosignalen mit dem Ziel, die Eigenschaften unterschiedlicher Signale zu unterscheiden und Merkmale zu extrahieren.

Um eine Klassifizierung von Audiosignalen vorzunehmen, bieten CNN-Modelle die Möglichkeit, diese extrahierten Audiodaten zu interpretieren und auf Basis von Trainings- und Testdaten zu erkennen. Im Kapitel 2.4 werden die Grundlagen und Bestandteile von KI-Methoden wie Deep Learning und CNN im Allgemeinen als auch in Hinblick auf die Erkennung und Klassifizierung von Audiosignalen dargestellt.

Im letzten Teil des zweiten Kapitels werden Konzepte und Lösungen vorgestellt, die sich mit einer Überwachung (Monitoring) von Wohnbereichen älterer Menschen mit Hilfe von Kameras, Mikrofonen oder anderen Sensoriken befassen.

Kapitel 3 beschreibt die Konzeptionierung und die Implementierung geeigneter Komponenten und Methoden zur Geräusch- und Spracherkennung in Wohnräumen. Dazu zählen einerseits verschiedene Hardwarekomponenten zur Signalaufnahme und deren Anforderungen, Eigenschaften und Leistungsfähigkeiten und andererseits die Datenverarbeitung der aufgenommenen Audiosignale und die Trainingsdatengenerierung zur Anwendung verschiedener CNN-Modelle.

Im Kapitel 4 werden die im Rahmen der Arbeit durchgeführten experimentellen Ergebnisse beschrieben und bewertet. Ein Schwerpunkt dieses Kapitels behandelt die Gegenüberstellung eingesetzter Hardwarekomponenten für die Audioaufnahmen und den Vergleich von CNN-Modellen und ihrer Parametrierung.

Eine Zusammenfassung der Arbeit sowie eine Betrachtung einer Integration in bestehende Systeme zur Raum- und Vitalüberwachung mit Hilfe von Kameras finden im letzten Kapitel statt. Zusätzlich wird ein technischer Ausblick gegeben, wie die ermittelten Methoden der Geräusch- und Spracherkennung als altersgerechtes Assistenzsystem im Wohnumfeld älterer Menschen weiter vertieft und entwickelt werden können.

2 Wissenschaftliche und technische Grundlagen

In diesem Kapitel werden wissenschaftliche Veröffentlichungen analysiert, der aktuelle Status der Technik im Kontext der Audiosignalerzeugung und -verarbeitung beschrieben, als auch Methoden dargestellt, Geräusche und Sprache zu erkennen und zu unterscheiden.

Diese Informationen sind das Fundament für die in dieser Arbeit beschriebenen Modelle einer CNN-gestützten Geräusch- und Spracherkennung im Wohnumfeld älterer Menschen und tragen insbesondere bei der Entscheidung zur Auswahl von Hardwarekomponenten, Verfahren zur Audiosignalverarbeitung und -extraktion sowie der Gestaltung eines CNN-Modells bei.

Im ersten Teil dieses Kapitels werden verschiedene technische Möglichkeiten einer Audioaufnahme beschrieben, die für eine Geräusch- und Spracherkennung geeignet sind. Danach werden Prozesse der Verarbeitung des Audiosignals insbesondere die Digitalisierung und die Möglichkeiten der Extraktion der aufgenommene Audiorohdaten behandelt.

Im weiteren Verlauf werden KI-gestützte Verfahren insbesondere Deep Learning und CNN beschrieben und ihre Eignung im Kontext mit der Geräusch- und Spracherkennung behandelt.

Abschließend werden in diesem Kapitel Systeme und Anwendungen vorgestellt, die sich mit verschiedenen Methoden einer Raumüberwachung, KI-Modellen oder Technologien im Bereich der Audiosignalverarbeitung und Extraktion beschäftigen.

2.1 Audioaufnahme und -verarbeitung

Im Folgenden werden die für die Audioaufnahme relevanten Teile Mikrofontechnik und digitale Signalverarbeitung beschrieben.

2.1.1 Mikrofontechnik

Die Auswahl geeigneter Mikrofone und deren Positionierung sind in der professionellen Umgebung als auch in alltäglichen Situationen eine wichtige Voraussetzung für eine hochwertige Aufnahme.

Im professionellen Bereich sind im Allgemeinen dynamische Mikrofone mit elektrodynamischer Wandlung und Kondensatormikrofone mit elektrostatischer Wandlung im Einsatz, um Geräusche und Sprache in einer hohen Qualität aufzunehmen [5]. Diese Art von Mikrofonen generieren mittels Membran analoge Audiosignale, die nachgelagert elektrisch verstärkt und in der Regel nach einer Analog-Digital-Wandlung digitaltechnisch weiterverarbeitet werden (Abb.2.1).

Im Bereich der Unterhaltungselektroniken wie Smartphones, Laptops und Tablets werden Miniaturmikrofone eingesetzt, die im Gegensatz zu den oben genannten professionellen Mikrofonen ein elektrostatisches Wandlerprinzip aufweisen. Dabei handelt es sich um Mikrofone mit mikro-elektro-mechanischen Systemen, fachsprachlich MEMS-Mikrofone genannt.

Während Mikrofone mit elektrodynamischer oder elektrostatischer Wandlung an Mischpulten oder digitalen Audiointerfaces angeschlossen werden, bietet sich bei MEMS-Mikrofonen die Integration auf den Boards der Unterhaltungselektroniken oder eine Kopplung mit einem separaten Mikrocomputer an.

MEMS-Mikrofone zeichnen sich durch ihre kleine Baugröße, die Eigenschaft direkt auf Platinen gelötet werden zu können als auch ihren niedrigen Preis aus [6]. Weiterhin haben Ausführungen von MEMS-Mikrofonen einen integrierten AD-Wandler und häufig eine digitale Schnittstelle zur seriellen Übertragung der Audiosignale. Bei MEMS-Mikrofonen ohne integriertem AD-Wandler spezifiziert der AD-Wandler des Mikrocomputers oder nachgeschalteter Elektroniken die Güte der Wandlung.

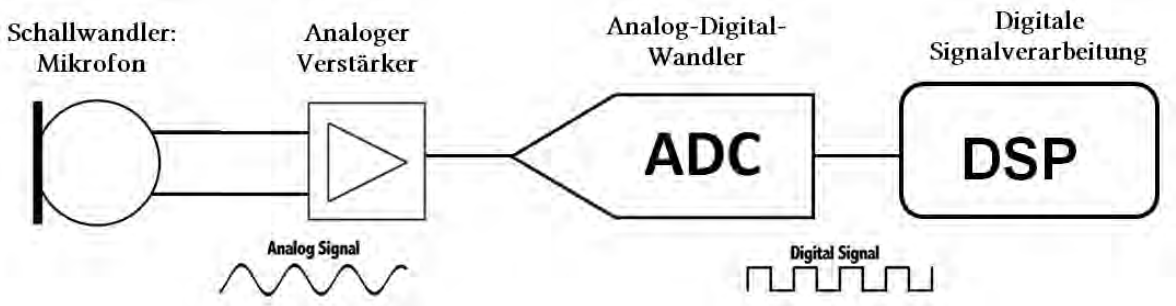


Abbildung 2.1: Blockschaltbild Mikrofon - Verstärker - AD-Wandlung
Quelle: Eigene Darstellung

Die nachfolgende Übersicht (Tabelle 2.1) zeigt eine Gegenüberstellung von Mikrofonen und deren wesentlichen Eigenschaften, die für eine Geräusch- und Spracherkennung in Frage kommen und im Kapitel 3.2.2 näher betrachtet werden.

Tabelle 2.1: Mikrofone und ihre mögliche Eignung zur Geräusch- und Spracherkennung im Wohnumfeld älterer Menschen

Mikrofontechnik	**Eigenschaften positiv**	**negativ**
Dynamisches Mikrofon	Frequenzbereich, geringes SNR	Baugröße, Kosten Dynamik
Kondensatormikrofon	Frequenzbereich, geringes SNR	Baugröße, Kosten
Elektrolytmikrofon	Kosten	Frequenzbereich, Baugröße
MEMS-Mikrofon	Frequenzbereich, Kosten, Größe	mittleres SNR benötigt Spannung
Smartphone o. Tablet-Mikrofon	im Gerät integriert, Software im Gerät	mittleres SNR, Frequenzbereich

2.1.2 Digitale Signalverarbeitung

Grundlage für eine digitale Signalverarbeitung ist die Wandlung des analog aufgenommenen Audiosignals in ein digitales Signal.

Seit Ende der 1970er Jahre findet im Audiobereich ein Systemwandel mit der Ablösung analoger Systeme durch digitale Technologien statt. Die Hauptgründe sind unter anderem [7]

- bessere technische Übertragungseigenschaften
- verlustloses Kopieren und Archivieren von Tonaufnahmen
- zusätzliche Möglichkeiten der Signalverarbeitung
- niedrige Kosten digitaler Hardware und Software

Abtastung

Bei der Abtastung wird ein zeitkontinuierliches Signal in eine Folge von Abtastwerten (Samples) gewandelt. Danach werden diskrete Amplitudenwerte zum Abtastzeitpunkt verarbeitet.

Die Abtastrate, auch Samplingfrequenz genannt, ist die Anzahl der Abtastungen pro Sekunde. Die Abtastrate ist ein Qualitätsmerkmal und bestimmt die Bandbreite des abgetasteten Signals. Ein abgetastetes Signal lässt sich ohne Informationsverlust rekonstruieren, wenn die Abtastfrequenz mehr als doppelt so hoch ist wie die höchste im Signal vorkommende Frequenz (Shannon Abtasttheorem) [8].

Das bedeutet, dass ein für den Menschen hörbares Frequenzspektrum von 20 Hz bis 20 kHz Abtastraten von 40 kHz und höher benötigt. Für eine Erkennung von Geräuschen und Sprache hat dies eine entscheidende Bedeutung. Eine Abtastrate von 16 kHz deckt im Frequenzbereich eine Bandbreite von 8 kHz ab. Damit wirkt die Abtastrate wie ein Tiefpassfilter (High-Cut). Die Auswirkungen werden in den folgenden Kapiteln im Detail beschrieben.

Quantisierung

Ebenso wie ein digitales Signal keinen kontinuierlichen Zeitverlauf hat, kann es auch keinen kontinuierlichen Amplitudenverlauf besitzen, wodurch nur diskrete Werte abgespeichert werden können.

Der Amplitudenwert wird in diesem Fall durch die Wortbreite, d.h. die Zahl der Bits pro Zahlenwert bestimmt [9]. Eine Wortbreite von 10 bit entspricht 2^{10} = 1024 Quantisierungsstufen. Bei einer maximalen Amplitudenspannung von 1 V würde in diesem Fall die Auflösung einer Quantisierungsstufe 1 V / 1024 ≅ 1 mV entsprechen.

Eine niedrige Wortbreite führt zu Quantisierungsfehlern (Abbildung 2.2).

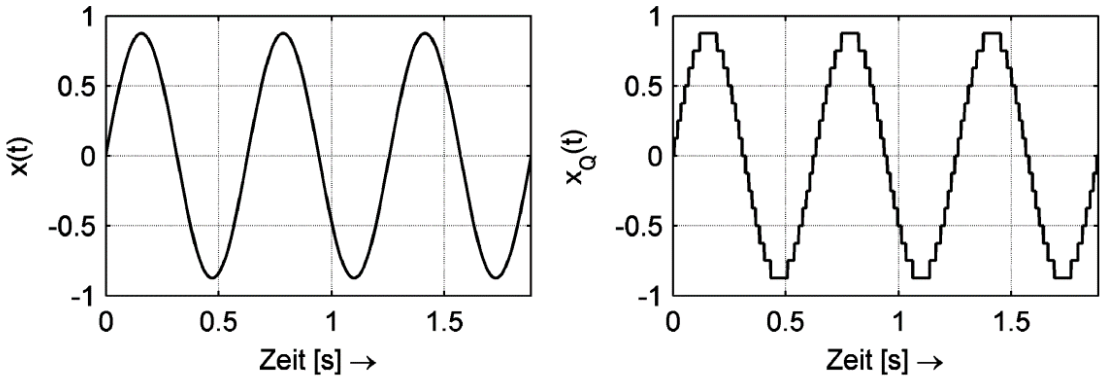


Abbildung 2.2: Quantisierungsfehler: kontinuierliches Originalsignal x(t) (links) und mit einer Auflösung von 4 bit quantisiertes Signal xQ(t) (rechts) Quelle: Stefan Weinzierl, Handbuch der Audiotechnik [9]

Die Größe des Quantisierungsfehlers wird durch den Signal-Rauschabstand (Signal-to-Noise-Ratio SNR) angegeben und als logarithmisches Verhältnis (Pegelverhältnis) von Signalleistung W_s zu Fehlerleistung W_q berechnet (2.1):

$$SNR = 10log\left(\frac{W_s}{W_q}\right) \tag{2.1}$$

Da es sich um deterministische Werte handelt, wird die Leistung über den Mittelwert der Quadrate aller Amplitudenwerte berechnet.

Wie bereits oben bei der Abtastung beschrieben, führt die Wandlung eines analogen Signals, das einen Frequenzbereich bis 20 kHz abdeckt, bei einer Abtastrate unterhalb des Zweifachen dieser Frequenz zu einem Informationsverlust des Signals. Ein Vergleich des Frequenzspektrums eines analogen und eines digital-gewandelten Audiosignals zeigt diesen Verlust im hohen Frequenzbereich. Vertiefende Möglichkeiten einer bildlichen Darstellung von Audiosignalen wie Frequenzspektren oder Spektrogramme werden im nächsten Abschnitt näher beschrieben.

Abbildung 2.3 zeigt Frequenzspektren eines Testsignals mit Frequenzen bis zu 16 kHz mit einer Abtastrate von 48 kHz (verlustfrei) sowie einer Abtastrate von 16 kHz, bei der zu erkennen ist, dass Frequenzen ab 8 kHz nicht mehr dargestellt werden und Informationen über die Bandbreite des Audiosignals verloren gehen.

Ein Nachteil der AD-Wandlung mit dem Anspruch einer verlustlosen Weiterverarbeitung des Signals ist die große Datenmenge der Signalinformationen pro Zeiteinheit. Eine Abtastrate von 16 kHz bei einer Bitrate von 10 bit erzeugt 160 kbit/s für ein Monosignal und eine Abtastrate von 48 kHz bei einer Bitrate von 24 bit bereits 1.152 kbit/s.

Im Folgenden werden verschiedene Kompressionsverfahren beschrieben, die die Datengröße der Signalinformationen reduzieren mit dem Vorteil einer schnelleren Datenverarbeitung zum Beispiel bei der Signalverarbeitung mit CNN-Modellen. Entscheidend hierbei ist ein akzeptabler Kompromiss aus Datenreduktion gegenüber dem Verlust von Informationen des Signals. Konkrete Auswirkungen einer erhöhten Extraktion mit Verlust von Informationen werden im Kapitel 4.1.4 dargestellt.

2.1.3 Datenkompression

Datenkompressionsverfahren unterscheiden sich durch verlustbehaftete und verlustlose Codierungen. Der erreichbare Kompressionsgrad hängt von dem jeweiligen Kompressionsverfahren ab.

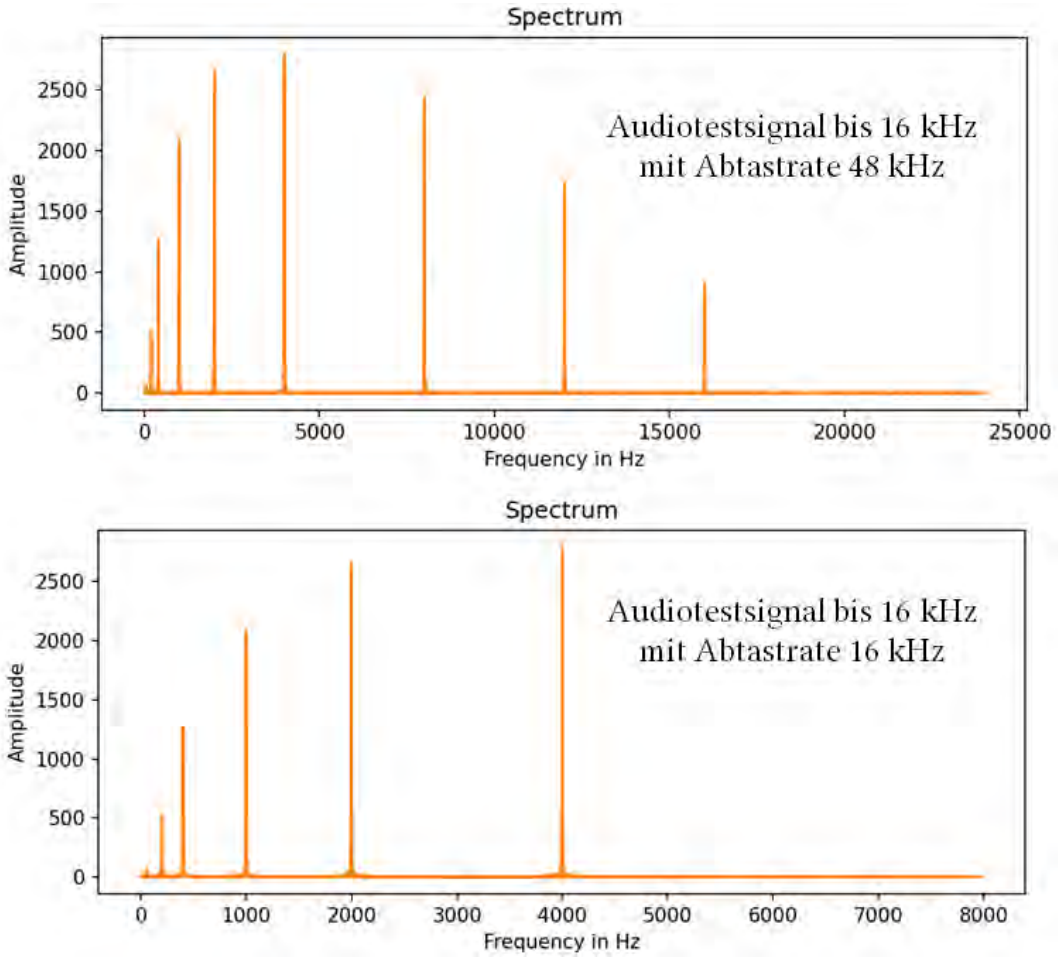


Abbildung 2.3: Spektrum nach Wandlung eines Testsignals mit einer Abtastrate von 48 kHz und 16 kHz
Quelle: Eigene Darstellung

WAW-Format

Audiodaten im WAV-Format sind unkomprimiert und damit verlustfrei. Der verwendete Speicherplatz ist gegenüber komprimierten Audioformaten hoch. Bei einer ausreichenden Abtast- und Bitrate ist die digitale Information des Signals nicht vom analogen Ursprungssignal zu unterscheiden. Dies hat den Vorteil, dass eine spektrale Darstellung des Signals dem Spektrum der Aufnahmequelle entspricht. Im Gegensatz dazu wirkt sich die Größe der Daten negativ auf die Datenübertragungs- und Verarbeitungsgeschwindigkeit aus.

MP3-Format

Das MP3-Format ist eines der bekanntesten Audioformate. Es handelt sich hierbei um ein Format mit verlustbehafteter Codierung. Der Grad der Komprimierung und damit die Güte des Audiosignals kann bei MP3-Formaten eingestellt werden. Bei kleineren Bitraten kommt es zu einem hörbaren Qualitätsverlust. Das Kompressionsverfahren nutzt den Umstand, dass der Mensch einen Großteil der akustischen Informationen mit dem Ohr nicht wahrnehmen kann. Insbesondere bewirkt der Verdeckungseffekt im menschlichen Ohr, dass Töne aufgrund des Vorhandenseins anderer Töne nicht wahrgenommen werden. Bei einer spektralen Auswertung zum Beispiel im Rahmen einer Geräusch- und Spracherkennung kann eine MP3-Codierung zu Informationsverlusten des Originalsignals führen.

WMA-Format

Bei dem WMA-Format handelt es sich um ein Windows-Audio-Format, das wie das MP3-Format eine verlustbehaftete Komprimierung durchführt. Das WMA-Format bietet eine hohe Klangqualität trotz guter Kompression. Es besteht ebenfalls bei dieser Komprimierung das Risiko eines Informationsverlustes.

2.2 Bildliche Darstellungsformen von digitalen Audiosignalen

Im folgenden Abschnitt werden bildliche Darstellungen von Audiosignalen in Form von Kartesischen Koordinatensystemen mit verschiedenen Achsenparametern vorgestellt.

2.2.1 Amplitude-Zeit-Diagramm, Frequenzspektrum und Spektrogramm

Im Rahmen einer bildlichen Darstellung von Audiosignalen bieten sich verschiedene Darstellungsformen an. Wesentliche Parameter sind die drei Dimensionen Zeit, Amplitude als auch die Frequenz. Das Amplituden-Zeitdiagramm stellt den Verlauf eines Signals als Amplitude einer Spannung über einen zeitlichen Verlauf dar und liefert keine Information über den Frequenzbereich des Signals.

Im Gegensatz dazu liefert ein Frequenzspektrum die Amplituden von Frequenzen entlang eines Frequenzbereichs, ohne Information über den zeitlichen Verlauf und damit die Änderung

des Signals über einen Zeitraum zu geben. Das Frequenzspektrum wird durch eine Diskrete Fourier-Transformation aus einem endlichen, zeitdiskreten Audiosignal generiert.

Spektrogramme sind eine kombinierte Darstellung aus einem Amplituden-Zeit-Diagramm und einem Frequenzspektrum. Sie zeigen die spektrale Veränderung über einen zeitlichen Verlauf. Veränderungen der Amplitude werden durch Farbunterschiede gekennzeichnet. Eine Interpretation und Auswertung eines Spektrogramms ist bei einer Geräusch- und Spracherkennung hilfreich, da in einem Bild sowohl die Frequenzen des Signals als auch der zeitliche Verlauf erkennbar sind. Spektrogramme sind eine Folge von Short-Time-FFT erstellten, überlappenden Zeitfenstern von Signalen. In einem Spektrogramm sind die farblich dargestellten Frequenzverteilungen im zeitlichen Verlauf zu erkennen.

Im Rahmen dieser Arbeit wurde zur bildlichen Darstellung von Audiosignalen ein Testsignal mit den folgenden Eigenschaften generiert:

- Abtastrate: 48 kHz
- Bitrate: 32 bit
- Amplitude: 2^{16} (Maximum)
- Signaldauer pro Freq.: 1 Sekunde pro Frequenz mit 90% Signal, danach 10% Pause
- Frequenzverlauf: Tabelle 2.2
- Weißes-Rauschen: nach 9 Sekunden für 0,9 s
- Abschluss-Signal: Pulsfolge 5x 1 kHz 50 ms nach 10 Sekunden

Tabelle 2.2: Frequenzverlauf Testsignal

Zeitperiode	0-1 s	1-2 s	2-3 s	3-4 s	4-5 s	5-6 s	6-7 s	7-8 s	8-9 s
Frequenz	50 Hz	200 Hz	400 Hz	1 kHz	2 kHz	4 kHz	8 kHz	12 kHz	16 kHz

Auf Basis dieses Testsignals werden zur Veranschaulichung mit einem Pythonprogramm generierte Diagramme erzeugt.

Das Amplituden-Zeit-Diagramm (Abbildung 2.4) zeigt keine Frequenzen. Die Amplituden werden aufgrund der zu geringen Abtastrate der Auswertungssoftware bei höheren Frequenzen nicht korrekt abgebildet.

Im Diagramm des Frequenzspektrums (Abbildung 3.2) sind die spektralen Inhalte deutlich zu erkennen, wobei das Weiße Rauschen als Amplitude nicht sichtbar ist. Aus der Darstellung kann kein zeitlicher Verlauf abgeleitet werden. Somit ist eine Unterscheidung einer Tonfolge wie bei Klingeltönen eines Telefons nicht erkennbar.

Um einen zeitlichen Verlauf des Frequenzspektrums darzustellen, werden Spektrogramme (Abbildung 2.6) verwendet. Das Weiße Rauschen wird in diesem Fall als eine gleichmäßige Farbe über den Frequenzbereich angezeigt.

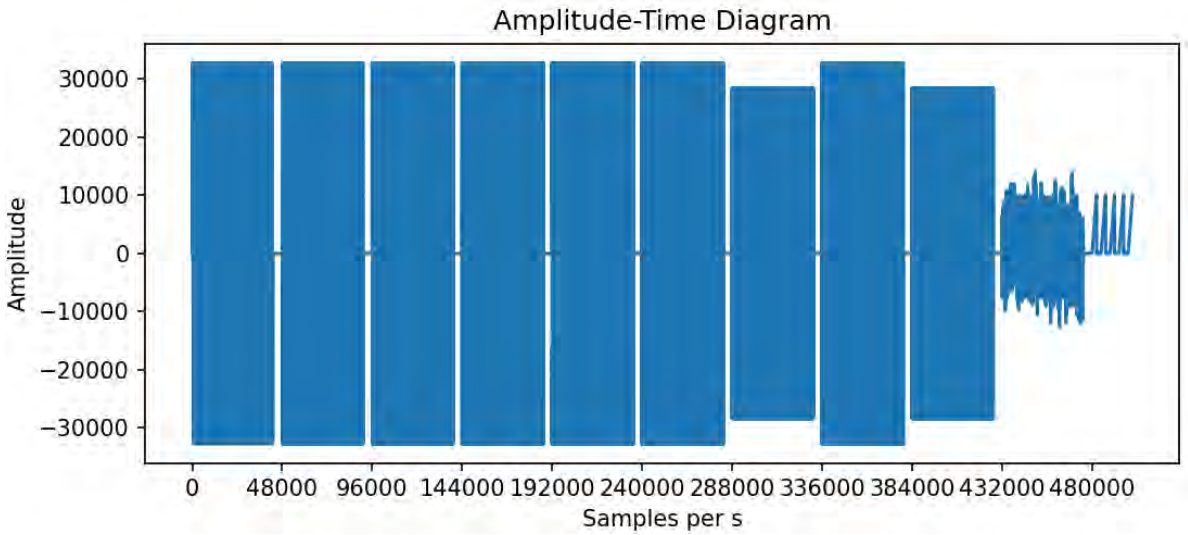


Abbildung 2.4: Amplitude-Zeit-Diagramm mit Abtastrate 48 kHz
Quelle: Eigene Darstellung

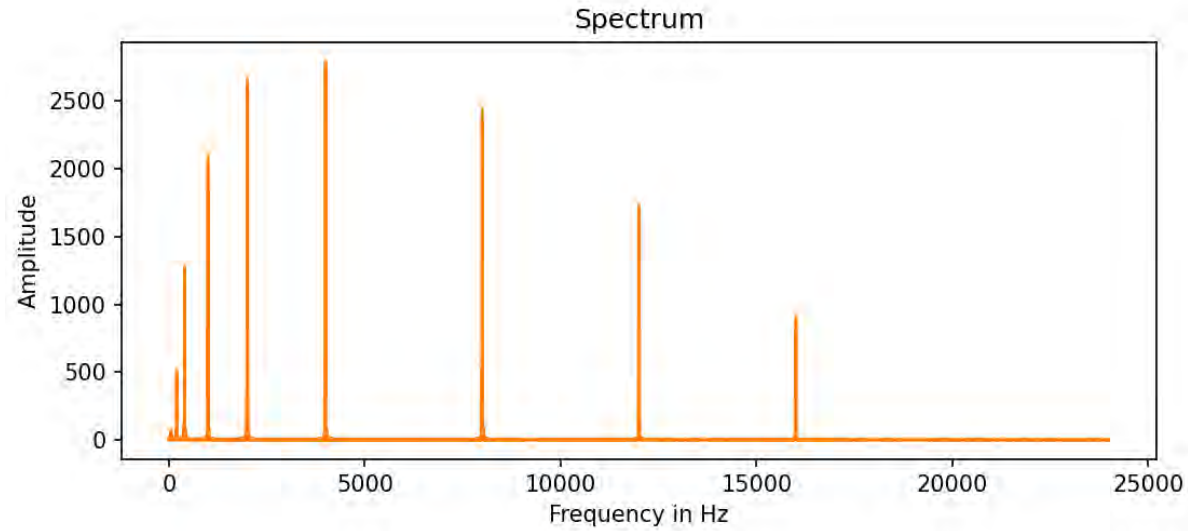


Abbildung 2.5: Spektrum-Diagramm
Quelle: Eigene Darstellung

2.2.2 Mel-Spektrum und Mel-Frequenz Cepstrum Koeffizienten

Das Mel (vom englischen Wort melody) ist die Maßeinheit für die psychoakustische Größe Tonheit mit dem Formelzeichen Z (oder z) und beschreibt die wahrgenommene Tonhöhe von Sinustönen, also die Tonhöhenwahrnehmung.

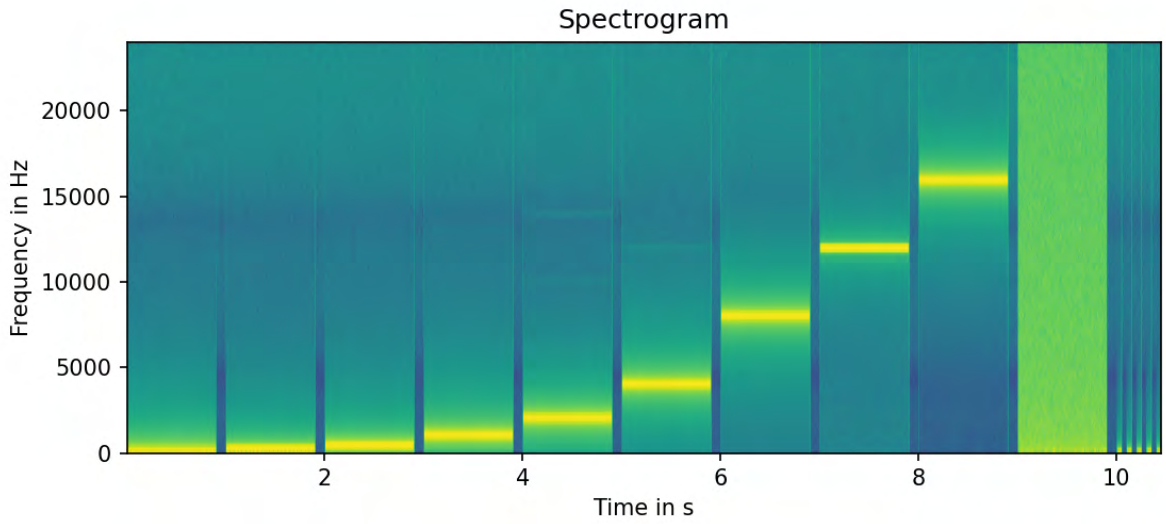


Abbildung 2.6: Spektrogramm
Quelle: Eigene Darstellung

Mel-Spektrogramm

Wie oben dargestellt, werden Frequenzspektren durch eine Fourier-Transformation von Audiosignalen im Zeitbereich generiert. Sie bieten hilfreiche Information bezüglich der Verteilung der Frequenzen im hörbaren Bereich. Damit eignet sich das Spektrum zur Unterscheidung von Audiosignalen mit unterschiedlichen Frequenzen.

Menschen nehmen Frequenzen entlang ihres Hörspektrums nicht linear wahr. Unterschiede in niedrigen Frequenzen können besser wahrgenommen werden als Unterschiede hoher Frequenzen, wenn der absolute Abstand zwischen den Paaren zum Beispiel mit 500 Hz gleich ist.

Stevens, Volkmann und Newman stellten 1937 die Tonhöheneinheit Mel vor, so dass Tonhöhenabstände für den Zuhörer gleich weit entfernt klingen. Diese Skalierung wird Mel-Skala bezeichnet [10].

Das Mel Spektrogramm ist ein Frequenzspektrogramm, indem die Mel-Skala Berücksichtigung findet.

Mel-Frequenz Cepstral Koeffizient MFCC

Mel-Frequenz Cepstral Koeffizienten sind Extrakte aus Frequenzspektren zur Reduzierung der Datenmenge und bieten eine gute Möglichkeit, Merkmale von Audiosignalen zu beschreiben, ohne das komplette Frequenzspektrum zu betrachten.

Die Bestimmung der MFCC ist ein Verfahren, die wichtigsten Merkmale eines Mel Spektrums in Form von Koeffizienten abzubilden. Der Begriff “Cepstral” ist hierbei eine Phantasieableitung des Wortes Spectrum und wurde 1963 in einem Artikel von Bogert, Healy und Tukey eingeführt [11].

MFCC wurden in zahlreichen experimentellen Ergebnissen ermittelt und bilden einen guten Datensatz an Merkmalen für die Spracherkennung.

Die Berechnung der Koeffizienten erfolgt in den Schritten [12]:

1. Berechnung der Koeffizienten $c_{\tau k}^{(ls)}$ des Leistungsspektrums für ein Datenfenster mit N Abtastwerten am diskreten Zeitpunkt τ
2. Transformation in die Koeffizienten $c_{\tau j}^{(mf)}$ der mel-Frequenzskala, wobei N_d Filter d(j,k) mit Mittenfrequenz j verwendet werden
3. Berechnung von $N_{m\text{-}c}$ mel-Cepstrum Koeffizienten $c_{\tau k}^{(mc)}$ durch Logarithmierung und Diskrete-Cosinus-Transformation (DCT)
4. Berechnung der ersten und zweiten zeitlichen Ableitungen $\Delta_{\tau k}^{(mc)}$ und $\Delta\Delta_{\tau k}^{(mc)}$
5. Bildung eines Merkmalsvektors c_τ aus $c_{\tau k}^{(mc)}$, $\Delta_{\tau k}^{(mc)}$ und $\Delta\Delta_{\tau k}^{(mc)}$
6. Gegebenenfalls Kompression des Merkmalsvektors

Bei der Berechnung gibt es Freiheitsgrade wie die zeitliche Positionierung der Fensterfunktion, der Wahl von Zahl, Mittenfrequenz und Breite der Dreiecksfenster, auf die an dieser Stelle nicht eingegangen wird.

Zur Spracherkennung werden in der Regel dreizehn Koeffizienten gewählt, bei Musik reichen die ersten fünf Koeffizienten [13].

Testsignal mit Mel-Spektrum und MFCC

Ergänzend zu den oben gezeigten Diagrammen Amplitude-Zeit, Frequenzspektrum und Spektrogramm wird nachfolgend das Testsignal als Mel-Spektrum (Abbildung 2.7) und als MFCC inklusive der Differenzen der Koeffizienten - als delta und delta^2 bezeichnet - (Abbildung 2.8) dargestellt.

Die Darstellung des Mel-Spektrums ist vergleichbar mit dem Spektrogramm (Abbildung 2.6).

Mel-Spektrogramme berücksichtigen das menschliche Hörverhalten, indem die Abstände zwischen den Frequenzen gegenüber einer linearen Darstellung angepasst werden. Es ist zu prüfen, ob diese Berücksichtigung einen Vorteil in der Klassifizierung von Geräusch- und Sprachsignalen mit CNN-Modellen bietet.

Ergänzend zum Mel-Spektrogramm existiert die Bark-Skala nach Heinrich Barkhausen, die in dieser Arbeit nicht verfolgt wird.

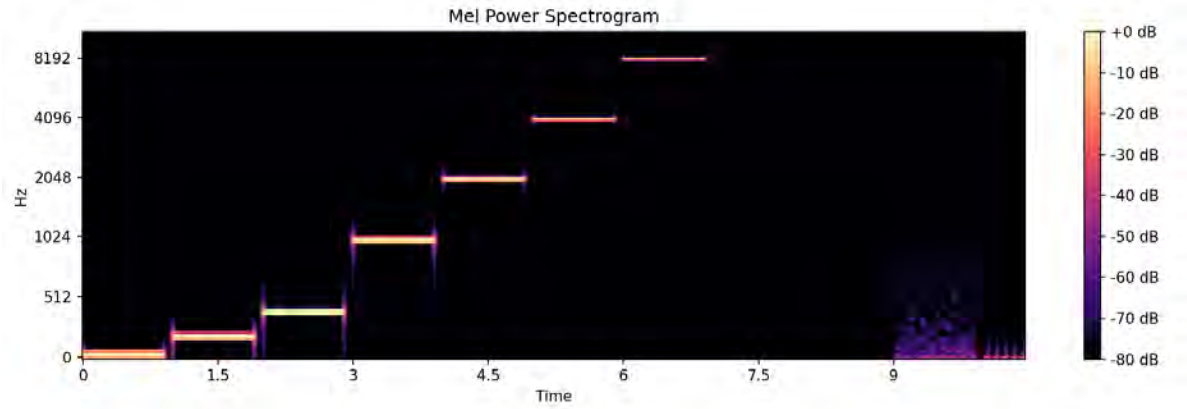


Abbildung 2.7: Mel-Spektrogramm
Quelle: Eigene Darstellung

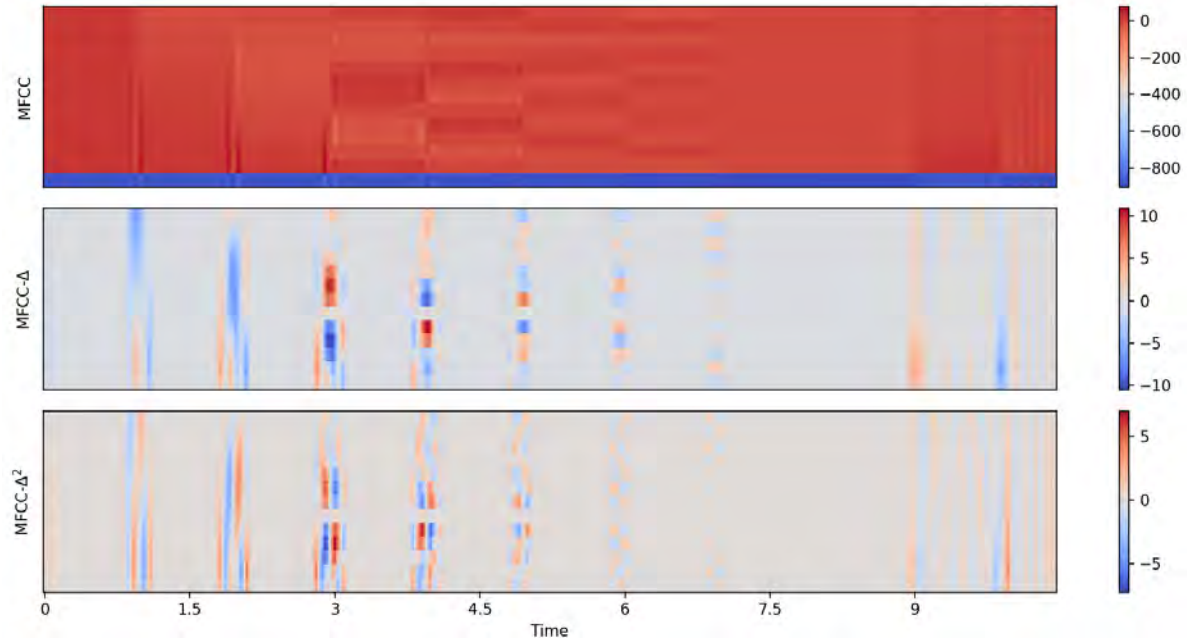


Abbildung 2.8: MFCC, delta und delta^2
Quelle: Eigene Darstellung

MFCC komprimieren die Information eines Audiosignals und eignen sich somit für eine Erkennung und Unterscheidung von Audiosignalen. MFCC-Algorithmen sind in Programmiersprachen wie Python und MatLab als Bibliotheken vielfältig vorhanden. Ein Nachteil von MFCC sind schlechtere Ergebnisse von Merkmalen bei Hintergrundgeräuschen [14].

2.3 Merkmale Frequenzbereich basierender Audioeigenschaften

Neben einer bildlichen Darstellung von Audiosignalen inklusive der Darstellung von speziellen Merkmalen wie die MFCC existieren in der Audiotechnik weitere Messgrößen, um Merkmale von Signalen zu bestimmen mit dem Ziel, Ähnlichkeiten oder Unterschiede zwischen den Signalen festzustellen.

2.3.1 Band Energy Ratio BER

Das Band Energy Ratio (BER) gibt das Verhältnis von tiefen und hohen Frequenzbändern an und wird häufig für die Unterteilung eines Audiosignals in Musik- und Sprachteile genutzt.

2.3.2 Spectral Centroid SC

Das Spectral Centroid (Schwerpunktwellenlänge) gibt den Schwerpunkt des Spektrums als Frequenzband an und wird als Mittel der Wellenlängen, gewichtet mit ihren Amplituden, berechnet. Es dient als Indikator für die Klangfarbe und wird unter anderem in der Audio- und Musikklassifikation verwendet. Weitere mit diesem Merkmal verwandte Eigenschaften sind unter anderem Spectral Bandwidth, Spectral Spread und Spectral Flux.

2.3.3 Korrelation

Die Korrelation beschreibt die Ähnlichkeit von Signalen. Ein normierter Korrelationsfaktor oder Korrelationskoeffizient ist ein Ähnlichkeitsmaß zweier Signale und berechnet sich vereinfacht aus dem möglichst großen Zeitintegral der Amplitudendifferenz dieser beiden Signale. Programmiersprachen wie Python bieten die Möglichkeit, aus zwei digitalen Audiosignalen einen Grad der Ähnlichkeit zu berechnen und daraus Musik, Geräusche und Sprache zu unterscheiden.

2.3.4 Zero Cross Rating ZCR

Zero Cross Rating ist ein Parameter für die Nulldurchgangsrate. Wenn das Signal vom Positiven zum Negativen beziehungsweise vom Negativen zum Positiven wechselt, wird der Nulldurchgang diskret gezählt. Der Wert wird zur Erkennung von Sprache und Musik genutzt [15].

2.4 Deep-Learning Methoden zur Klassifizierung von Audiosignalen

2.4.1 Begrifflichkeiten

Der Begriff "Künstliche Intelligenz (KI)" entstand 1956 auf einer Konferenz von Wissenschaftlern am Dartmouth College im US-Bundesstaat New Hampshire. John Mc Carthy schlägt diesen Begriff vor für die Ansicht der Wissenschaftler, dass Aspekte des Lernens sowie andere Merkmale der menschlichen Intelligenz von Maschinen simuliert werden können.

Maschinelles Lernen (Machine Learning ML) ist ein Teil der KI. ML analysiert große Datenmengen in Form von statistischen Modellen mit dem Ziel, Muster zu erkennen und das Ergebnis des Erfolges unter anderem mit einer Genauigkeit bzw. Wahrscheinlichkeit zu quantifizieren.

ML unterscheidet zwei Methoden, das überwachte und das unüberwachte Lernen. Im Rahmen dieser Arbeit wird das überwachte Lernen betrachtet. Diese Methode benötigt als Eingabe eine große Menge an Daten mit bekannten Zusammenhängen, die als Trainings-, Test- und Validierungsdaten genutzt werden, um neue Eingaben festgelegten Mustern zuzuordnen.

Bestandteile einer Implementierung des ML sind die Auswahl und Aufarbeitung des Trainingsdatensatzes sowie die Auswahl von einem Modell und seinem Algorithmus mit dem Ziel, eine hohe Prognose für die Zuordnung neuer Eingaben zu einer Prognose oder einem Muster, auch Klassifikation genannt, zu finden (Lernprozess).

Deep Learning (DL) ist ein spezieller Bereich des ML, bei dem künstliche Neuronen erzeugt werden, um Muster zu erkennen. Ziel ist es, die Struktur eines menschlichen Gehirns (neuronales Netzwerk) nachzuempfinden (Abbildung 2.9).

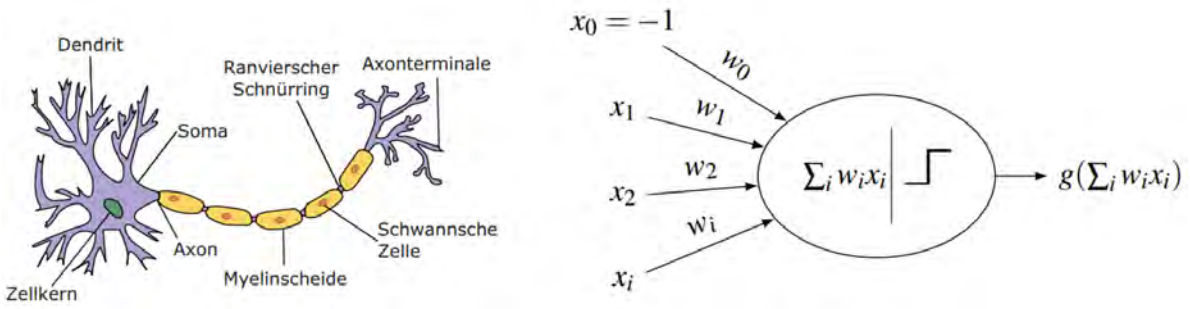


Abbildung 2.9: Darstellung eines realen und künstlichen Neurons
Quelle: Hartmut Ernst et al., Grundkurs Informatik [16]

2.4.2 Deep Learning

Ein Charakteristikum des Deep Learning und damit künstlicher, neuronaler Netzwerke ist die Schichtenstruktur:

- Die **Eingangsschicht** dient zur Informationsaufnahme und verarbeitet die Rohdateneingabe zum Beispiel das Audiosignal im WAV-Format oder als Spektrum in Form eines Bildes.
- Dahinter liegen mehrere **verdeckte Schichten** und Ebenen, wo die Informationen weiterverarbeitet und reduziert werden.
- Die letzte Schicht ist die **Ausgabeschicht**, die das Ergebnis als Informationsausgabe darstellt.

Eingangsschicht - Input Layer

Wie in Abbildung 2.9 dargestellt, werden die Eingangsinformationen mit den individuellen Gewichten w_i, wobei i für das individuelle künstliche Neuron steht, zu einer gewichteten Summe verrechnet. Überschreitet die Summe einen Schwellwert, entsteht am Ausgang ein Signal. Das Lernen bedeutet, die Gewichte aus einer Stichprobe zu bestimmen. Der Schwellwert zur Entscheidung, ob ein Signal am Ausgang entsteht, nennt man Bias. Die Schwelle wird durch Aktivierungsfunktionen beschrieben. Bei der Entwicklung eines Modells bietet es sich an, verschiedene Aktivierungsfunktionen zu erproben. Am weitesten verbreitet sind neben der Sprungfunktion die Sigmoid und die ReLu-Funktion.

Verdeckte Schicht - Hidden Layer

Verdeckte Schichten sind in mehreren Schichten aufgebaut, wobei jedes Neuron mit dem Neuron der nachfolgenden Schicht verbunden ist.

Deep Learning bedeutet, dass es viele verdeckte Schichten gibt und es sich um ein tiefes Netz handelt. Aus Vereinfachungsgründen ist in der Abbildung 2.10 ein System mit nur zwei verdeckten Schichten dargestellt. Da der Informationsfluss nur in eine Richtung geht, handelt es sich um ein Feed-Forward-Netz.

Für das Training des Modells ist eine Stichprobe in Form von Trainingsdaten erforderlich, bei denen bekannt ist, welcher Klasse (Muster) sie zugehören. Im Falle einer Geräusch- und Spracherkennung entspricht dies zum Beispiel dem Telefonklingeln oder einem Hilferuf einer Person.

Am Anfang werden die Gewichte auf Basis zufälliger Zahlen vorgegeben und die Trainingsdaten durch das Modell geschickt. Die tatsächliche Ausgabe an der Ausgangsschicht wird

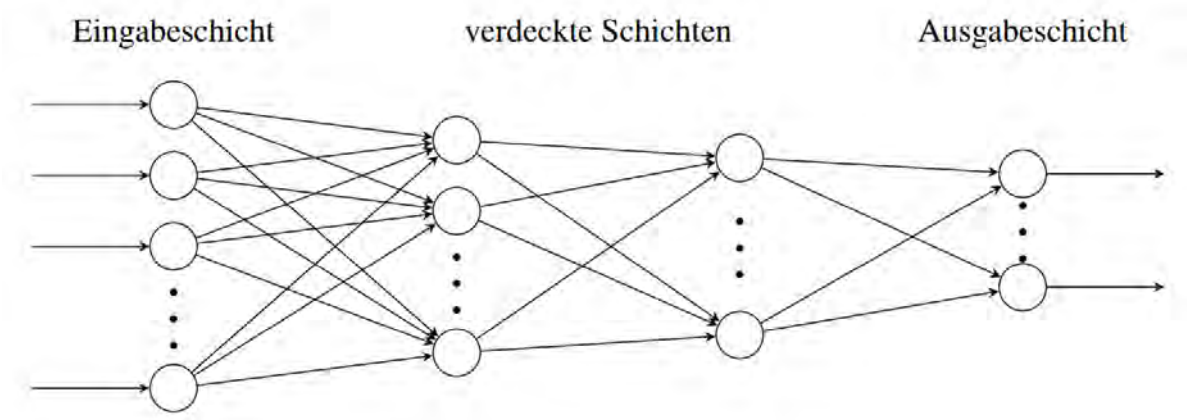


Abbildung 2.10: Struktur eines Mehrschichtperzeptrons
Quelle: Hartmut Ernst et al., Grundkurs Informatik [16]

danach mit dem gewünschten Ergebnis verglichen. Die Ergebnisse werden verwendet, um die Gewichtung von hinten nach vorne Schicht für Schicht zu ändern, damit das Netz besser wird. Diesen Prozess nennt man Fehlerrückführung bzw. Error Backpropagation. Die Lernrate bestimmt die Dynamik des Trainings. Eine geringe Lernrate führt zu einem lange dauernden Training, während eine hohe Lernrate zum Überschwingen des Systems führen kann. Dies hat ein Overfitting zur Folge.

Bei einem neuen Model wie bei einer Entwicklung eines Modells für eine Geräusch- und Spracherkennung ist die Lernrate ein wichtiges Kriterium. Dies gilt insbesondere für den Fall, dass nur eine eingeschränkte Menge an Trainings- und Testdaten zur Verfügung stehen, die Qualität dieser Daten beschränkt ist und das Modell viele Schichten besitzt.

Mittlerweile gibt es Modelloptimierer wie Adam, die unter anderem eine adaptive Lernrate verarbeiten.

Ausgangsschicht - Output Layer

Während des Lernens berechnet das Modell bei jedem Durchlauf (Epoche) die Vorhersagegenauigkeit (Accuracy). Für den Entwickler stehen Werkzeuge wie das Tensorboard und Hyperparametertuning zur Verfügung, die Ergebnisse zu überwachen und durch Veränderung von Parametern wie Anzahl Epochen und Lernrate das Modell zu optimieren.

Die oben ausgeführte Beschreibung basiert im Wesentlichen auf “Kapitel 18 Maschinelles Lernen” aus dem Buch “Grundkurs Informatik” von Hartmut Ernst, Jochen Schmidt und Gerd Beneken [16].

Transfer Learning

Transfer Learning ist eine Methode aus dem Deep Learning, mit der ein vortrainiertes, künstliches neuronales Netz für die Lösung neuer Problemstellungen genutzt wird. Dazu wird der Lernfortschritt des bestehenden Modells transferiert. Dadurch ergeben sich Vorteile, wie schnellere Erstellung, bessere Modellqualität und weniger Ressourceneinsatz [17].

Bei einem Transfer Learning entfernt man zumindest die Ausgabeschicht des Netzes, womit auch die Gewichte der letzten Schicht wegfallen. Im nächsten Schritt ersetzt man die Ausgabeschicht mit eine für das Problem passende Schicht, initialisiert die Gewichte neu und startet das Training erneut.

Das Netz bleibt damit größtenteils unverändert. Diese Vorgehensweise bietet sich besonders bei einer geringen Anzahl von Daten wie bei der Geräusch- und Spracherkennung an, wenn eigene Trainingsdaten aus Audiorohdaten im Wohnumfeld älterer Menschen generiert werden. Transfer Learning ist heute der Normalfall und keine Ausnahme [16].

2.4.3 Convolutional Neuronal Networks (CNN)

Bei einer KI-gestützten Erkennung von Bildern werden in der Regel Neuronale Faltungsnetze (Convolutional Neuronal Network CNN) verwendet. Dies hängt unter anderem damit zusammen, dass Bilder zweidimensional sind und jeder Punkt (Pixel) einen dreidimensionalen Vektor für die Farbe (RGB) hat. Für die Verarbeitung in der Eingangsschicht müssen die Informationen eindimensional aufbereitet werden. Da zur Geräusch- und Spracherkennung Bilder wie Spektrogramme oder MFCC ausgewertet werden, sind Machine-Learning-Modelle mit CNN-Architektur zu verwenden.

Abbildung 2.11 zeigt ein Beispiel einer CNN-Architektur. Die Eingabe besteht aus dem Bild mit allen Pixeln und Farbinformationen. Anschließend kommen mehrere Faltungsschichten hintereinander gefolgt von einem Pooling zur Verdichtung auf die jeweils stärksten Merkmale einer Matrix. Diese Abfolge wird mehrfach wiederholt.

Am Ende folgt in diesem Beispiel die Aktivierungsfunktion ReLu. Für die Ausgabefunktion wird eine passende Aktivierung wie in diesem Beispiel Softmax gewählt [16].

Faltungsschichten machen sich den Effekt zu Nutze, kleine Umgebungen in einem Bild zu betrachten und dieses Musterstück mit ähnlichen Musterstücken anderer Bilder zu vergleichen. Vergleichbare Faltungsoperationen werden auch bei digitalen Filtern in der Bildverarbeitung zum Beispiel Kontrast- und Kantenzeichner eingesetzt.

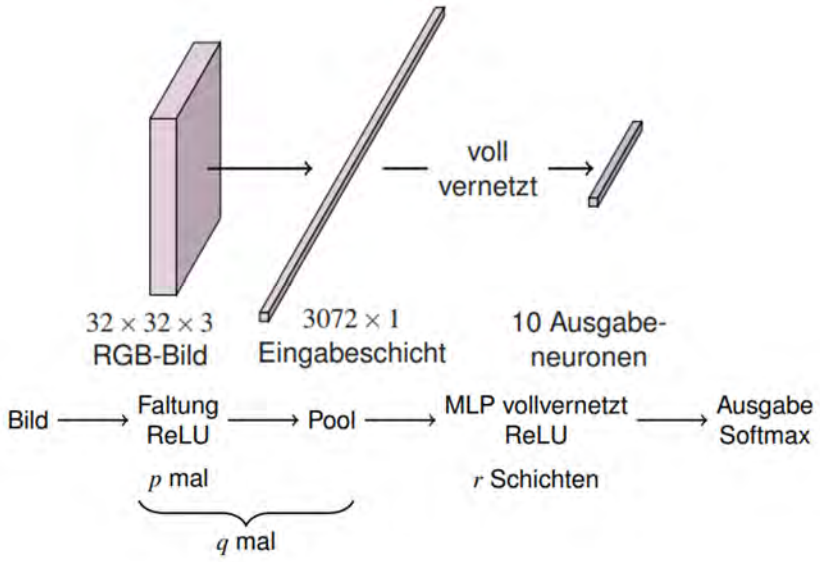


Abbildung 2.11: Architektur eines CNN
Quelle: Hartmut Ernst et al., Grundkurs Informatik [16]

Die oben beschriebenen Verfahren und Modelle sind geeignet, eine KI-gestützte Geräusch- und Spracherkennung zu ermöglichen. Ein wesentlicher Baustein ist dabei die Anzahl und die Qualität der Trainingsdaten.

2.5 Systembeispiele von Raumüberwachung und Geräuscherkennung im Wohnumfeld älterer Menschen

2.5.1 Intelligenter Bilderrahmen mit Stimmungsanzeige

Die Beyond Emotion GmbH [18] ist ein aus einer Ausgründung aus dem Forschungs- und Transferzentrum (FTZ) Smart Systems der Fakultät TI der Hochschule für Angewandte Wissenschaften Hamburg entstandenes Unternehmen, das den Intelligenten Bilderrahmen BEJOY entwickelt hat, der 17 verschiedene Emotionen und Gesichtsausdrücke analysiert. Der Bilderrahmen lässt (Groß-)Eltern Fotos anzeigen und informiert autorisierte Personen wie deren Kinder mit einer App für Smartphones in 5 verschiedenen Stufen über ihr Wohlbefinden. BEJOY besitzt eine individuell anpassbare Analysesensibilität und kann Gesichtsausdrücke auf die Eigenheiten einer Person abstimmen.

Das System verwendet ein Samsung Tablet zur optischen Aufnahme und Erkennung des Gesichtsausdrucks und wertet die Stimmung der Person im Raum mit Hilfe eines KI-gestützten Algorithmus aus. Zum Schutz der Privatsphäre erfolgt keine Übertragung von Videosignalen. Akustische Signale werden mit dem Produkt ebenfalls nicht erfasst und ausgewertet. Aus Sicht des Autors bietet sich an dieser Stelle die Option einer Integration mit einer Geräusch- und Spracherkennung an, wie sie im Kapitel 5.2 beschrieben wird.

2.5.2 Wohnbereich Telemonitoring auf Basis von Geräuschüberwachung

Titel: **Habitat Telemonitoring System based on the Sound Surveillance** [19]
Autoren: **Eric Castelli et al.**

Eric Castelli et al. beschreiben in ihrer Studie das Telemonitoring im häuslichen Bereich mit physiologischen Sensoren, Positionserkennung von Personen in Form von Infrarotsensoren sowie dem Einsatz von Mikrofonen. Der Ansatz ist es, Videokameraüberwachung zu ersetzen, da sie bei Patienten zum Schutz der Privatsphäre keine Akzeptanz findet.

Castelli et al. präsentieren ein System, das Daten kombiniert, die aus medizinischen Informationen und Audiosignalen bestehen, wobei die Auswertung der Audiosignale durch einen Algorithmus Stresssituationen erkennen soll. Die erfassten Audiosignale nehmen die gängigen Alltagsgeräusche auf und werden nicht gespeichert. Der Auswertungsalgorithmus klassifiziert die Kategorien. Im Rahmen der Entwicklung des Telemonitoringsystems wurden verschiedene Algorithmen erprobt. Castelli et al. bewerten das Ergebnis mit einer Fehlerrate von 10% als erfolgversprechend.

Es ist zu betonen, dass das System im medizinischen Bereich Anwendung finden soll. Das Einsatzgebiet beschränkt sich ausschließlich auf Innenbereiche, wobei in der Studie insgesamt fünf Räume mit jeweils einem Mikrofon inklusive Räume wie Toilette und Bad bestückt wurden.

Aus technischer Sicht wurde eine Abtastrate von 16 kHz gewählt. Trainingsdaten wurden sowohl selbst produziert als auch aus Datenbanken wie Geräusche von Haartrocknern verwendet. Es existierten insgesamt 20 verschiedene Geräuschkategorien mit Minimum 10 Beispielen. Damit ist von einer geringen Anzahl an Trainingsdaten auszugehen.

Als Modell wurde das Gauss Mixture Model verwendet, das im Kontext der Extraktion unter anderem MFCC verwendet. Mit dem MFCC-Verfahren wurden die besten Ergebnisse erzielt. Aus dem Bericht geht nicht hervor, inwieweit die IR- und physiologischen Sensoren die Vorhersagegenauigkeit beeinflusst haben.

2.5.3 Erkennung des Fallens von älteren Personen mit Hilfe von Geräuscherkennung und Vibrationssensor

Titel: **A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound** [20]
Autoren: **Yaniv Zigel et al.**

In ihrer Studie beschreiben Zigel et al. eine Methode zum Erkennen des Fallens älterer Menschen mit einem Vibrationssensor und zusätzlicher Geräuschüberwachung.

Im Rahmen der Extraktion der Merkmale wurden die Energiesignale des Vibrationssensors als auch eine Spektralanalyse mit MFCC verwendet. Es wurden 13 MFCC-Koeffizienten zur Erkennung verwendet. Die größte Herausforderung ist die Abgrenzung eines Fallereignisses einer Person gegenüber anderen Ereignissen. Die praktischen Versuche wurden mit einer Puppe ("Rescue Randy") durchgeführt, wobei in der Trainingsphase nur 40 Fallereignisse durchgeführt wurden. Die Testphase umfasste 20 Fall-Ereignisse. Die positive Erkennung eines Fall-Ereignisses wird in der Studie mit über 97% angegeben.

Die Autoren schlagen für weitere Versuche eine höhere Trainings- und Testanzahl der Fallereignisse vor. In Anbetracht der oben genannten Erfolgsquote ist die Validität der Ergebnisse aus Sicht des Autors dieser Arbeit zu verifizieren.

Aus der Studie geht ebenfalls nicht hervor, welchen quantitativen Beitrag die Geräuschüberwachung und Extraktion mit MFCC zum Gesamtergebnis beigetragen hat.

2.5.4 Fallereigniserkennung mit Microsoft Kinect in Wohnbereichen älterer Menschen

Titel: **Fall Detection in Homes of Older Adults Using the Microsoft Kinect** [21]
Autoren: **Erik Stone et al.**

Die Autoren Stone et al. setzen im Gegensatz zur oben genannten Methode von Zigel et al. einen Microsoft-Kinect-Sensor zur Detektion von Fallereignissen ein. Dabei handelt es sich um eine Hardware zur Steuerung von Videospielkonsolen. Aus technischer Sicht ist dies eine Kombination aus Farbkamera, Tiefensensor, 3D-Mikrofon und Software [22].

Die Evaluation wurde über 9 Jahre in 13 verschiedenen Apartments mit über 450 Fallereignissen durchgeführt. Dabei ist festzustellen, dass 445 Fallereignisse durch trainierte Stuntpersonen absolviert wurden.

Die Studie beschreibt eine Methode, in der neben der Microsoft Kinect-Sensorik keine weiteren unterstützenden Verfahren wie zum Beispiel eine Geräuscherkennung eingesetzt werden.

Es wurden verschiedene Fallereignisse behandelt. Dabei ist das Fallen aus einer aufrechten Position (standing) leichter zu detektieren, als wenn die Person aus der Sitzhaltung (sitting) zu Boden fällt. Eine weitere Herausforderung beschreiben die Autoren in der Anzahl falscher Alarme, die zu einer geringen Akzeptanz des Systems führen kann.

Die beiden oben genannten Studien machen deutlich, dass eine Erkennung eines Fallereignisses durch Sensoren und Geräuscherkennung eine technische Herausforderung darstellt. Sensoriken wie Vibrationssensor und Microsoft-Kinect müssen aus Sicht des Autors zusätzlich zur Geräuscherkennung eingesetzt werden, um ein Fallereignisses mit einer hohen Wahrscheinlichkeit und geringer Fehlalarmquote zu detektieren.

2.5.5 MFCC-CNN Stimmerkennung mit ESP32 und Web-Applikation

Titel: **On-Device MFCC-CNN Voice Recognition System with ESP-32 and Web-Based Application** [23]
Autoren: **Muhammad Ichsan Ramadani et al.**

Die Autoren Ramadani et al. stellen in ihrer Studie eine Low-Cost-Lösung einer Spracherkennung mit einem ESP32-Mikrocomputer, einer Auswertung der Audiosignale mittels MFCC-Extraktion und nachfolgender Klassifizierung mit einem CNN-Modell vor.

Die Simulationsergebnisse liefern eine Genauigkeit von 93% bei einer Reaktionszeit von unter 30 ms. Es handelt sich bei dem System ausschließlich um eine Spracherkennung von 6 definierten Befehlen. Die Sprachbefehle wurden von 15 Personen gesammelt. Die Abtastrate der Signale liegt bei 48 kHz. Jeder Sprachbefehl wurde mit einer Länge von einer Minute zehnmal aufgenommen. Als Extraktionsverfahren wurde MFCC gewählt. Für das CNN-Modell wurden 80% der Datensätze für Training und jeweils 10% für Test und Validierung verwendet. Insgesamt trainierte das Modell 40 Epochen.

Es ist hervorzuheben, dass der ESP32-Mikrocomputer in der Studie von Ramadani et al. nicht zur Erfassung der Audiosignale, sondern als Back-End zur Steuerung von Hardware wie einer Lampe auf Basis der Ausgabe des CNN-Modells eingesetzt wird.

Ramadani et al. verwenden in ihrer Studie eine geringe Anzahl von Sprachbefehlen. Die Erkennung der Sprachbefehle mit dem CNN-Modell liegt bei über 90%. Die Trainingsdatensätze wurden durch ausgesuchte Personen erstellt. Es wurde nicht auf bestehende Datensätze von Sprachbefehlen zurückgegriffen.

In dem folgenden Kapitel werden die Konzeptionierung und die Implementierung von Methoden zur Erkennung von Geräuschen und Sprache vorgestellt.

3 Konzeptionierung und Implementierung

In diesem Kapitel werden für die Konzeptionierung Rahmenbedingungen definiert, die die Methoden zur Geräusch- und Spracherkennung im Wohnumfeld beeinflussen. Dies umfasst die Festlegung der Räume und deren Größe, die Positionierung von Aufnahme- zu Schallquellen als auch die Betrachtung der akustischen Eigenschaften in einem Raum.

Weiterhin werden Hardwarekonfigurationen vorgestellt, die für eine Audiosignalerfassung und -verarbeitung mit dem Ziel der Geräusch- und Spracherkennung geeignet sind. Mit der Auswahl leiten sich verschiedene Softwareoptionen der eingesetzten Hardware ab, die im Anschluss der Hardwareauswahl vorgestellt werden. Der letzte Teil der Konzeptionierung ist die Festlegung der Audioklassen mit der Beschreibung der Signalcharakteristika (Abb. 3.1).

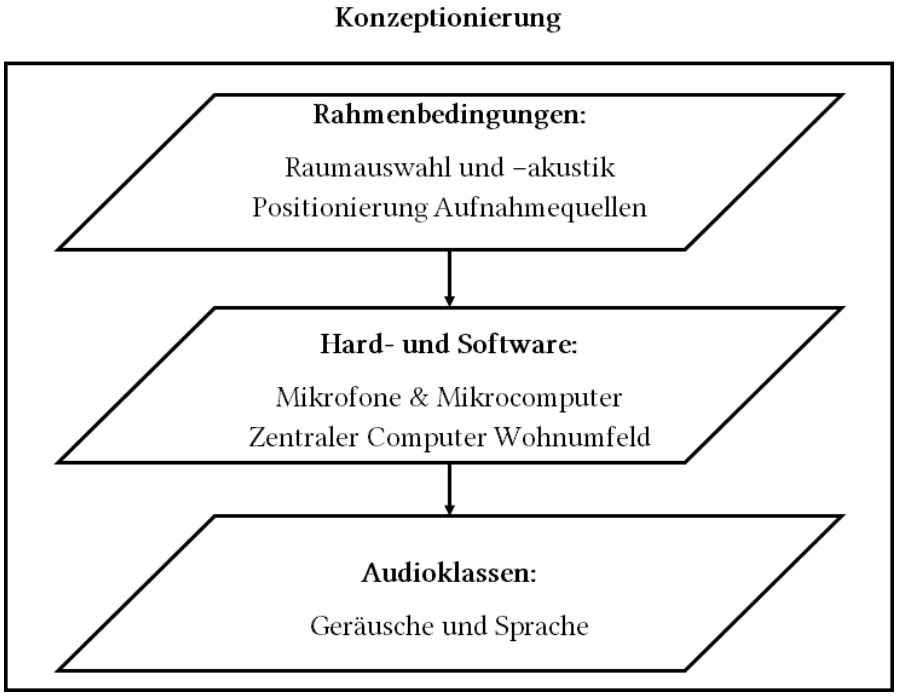


Abbildung 3.1: Konzeptionierung der Methoden zur Geräusch- und Spracherkennung
Quelle: Eigene Darstellung

Die Implementierung teilt sich auf in die Datenvorverarbeitung mit der Aufnahme von Audiorohdaten und der Generierung von Trainingsdaten sowie in die drei Hauptbereiche Audiodatenerfassung, Extraktion der wesentlichen Signalmerkmale und Deep Learning mit verschiedenen CNN-Modellen.

Abschließend werden Einflussgrößen entlang der Prozesskette analysiert, die eine Auswirkung auf die ausgewählten Methoden haben können.

3.1 Festlegung der Rahmenbedingungen

3.1.1 Auswahl der Wohnräume und Bestimmung der Raumgröße

Diese Arbeit geht von den folgenden Wohnbereichen aus, die von älteren Menschen bewohnt werden:

- Wohnzimmer
- Küche
- Flure und Eingangsbereiche
- Wirtschaftsräume
- Schlafzimmer
- Badezimmer

Die Raumgröße sollte auf maximal 20 m² begrenzt werden. Größere Räume können bei längeren Distanzen zwischen Schall- und Aufnahmequelle Auswirkungen auf die Höhe des Signalpegels und einen negativen Einfluss von Schallreflektionen haben. Für Schallquellen in Räumen überwiegt in größerer Entfernung von der Quelle der Anteil von gebeugtem und an den Wänden reflektiertem Schall, dem sogenannten Diffusschall.

Bei einem Raum mit den Abmessungen von 4 m x 5 m und einer Raumhöhe von 2,40 m würde der Hallradius etwa 1,30 m betragen (3.1).

$$r_{\mathrm{H}} = \sqrt{\frac{A}{16\pi}} \tag{3.1}$$

mit r_{H} = Hallradius, A = Raumflächen Wände, Decke, Boden in m²

Der Hallradius definiert die Entfernung, wo die Schallenergiedichte von Direkt- und Diffusschall gleich groß sind (3.2). Bei größeren Entfernungen überwiegt die Schallenergiedichte des Diffusschalls und damit der Schall, der beim Eintreffen auf den Hörort bzw. der Aufnahmequelle bereits mehrere Reflektionen erfahren hat [24].

$$w_{\mathrm{d}} = \sqrt{\frac{P}{c4\pi r^2}} = w_{\mathrm{r}} = \sqrt{\frac{4P}{cA}} \tag{3.2}$$

mit w_d = Schallenergiedichte Direktschall, w_r = Schallenergiedichte Diffusschall,
P = Schallleistung, A = Gesamtfläche des Raumes, r = Entfernung, c = Schallgeschwindigkeit

Um zu vermeiden, dass im oben genannten Beispiel der Hallradius unwesentlich überschritten wird, sollte die Entfernung zwischen Geräusch- und Aufnahmequelle eine Entfernung von 1,50 m nicht überschreiten. Mit zunehmender Entfernung dominieren Reflektionen und beeinflussen die Reinheit des originalen Audiosignals der Schallquelle.

Im Rahmen der Konzeptionierung wird eine Aufnahmequelle pro Raum verwendet. Dies hat die Vorteile, dass die Komplexität in der Auswertung des Audiosignals gering ist und es zu keinen Interferenzen durch Reflektionen und Signalverschiebung zwischen verschiedenen Aufnahmequellen in einem Raum kommt.

Demgegenüber steht der Nachteil, dass die Direktschallenergie mit dem Quadrat der Entfernung von der Quelle abnimmt und damit das Signal über die Entfernung immer schwächer und gegebenenfalls von Nebengeräuschen überschattet wird.

3.1.2 Berücksichtigung der Raumakustik

Die Raumakustik spielt bei den Methoden zur Erkennung von Geräuschen und Sprache insbesondere mit Deep Learning zur Klassifizierung eine wesentliche Rolle. Wie im Kapitel 2.4.1 auf Seite 17 beschrieben, wird im Rahmen dieser Arbeit das überwachte Lernen betrachtet. Diese Methode benötigt eine große Menge an Trainingsdaten, um neue Eingaben wie die Aufnahme des Audiosignals einer Klasse zuzuordnen.

Nachfolgend werden zwei verschiedene Ansätze beschrieben, wie die Merkmale der Raumakustik in den Trainingsdaten der Deep-Learning-Modelle Berücksichtigung finden:

1. Messtechnische Ermittlung der Raumakustik und Verwendung von Trainingsdaten aus vorhandenen Datenbanken oder
2. Generierung eigener Trainingsdaten im Wohnumfeld ohne Ermittlung der Raumakustik

Ansatz 1:
Messtechnische Ermittlung der Raumakustik und Verwendung von Trainingsdaten aus vorhandenen Datenbanken

Der Unterschied zwischen dem von der Schallquelle ausgesendeten und dem aufgenommenen Audiosignal hängt neben der Raumgröße von weiteren raumakustischen Merkmalen wie Reflektionen der Wände und der Decke sowie der Positionierung der Aufnahmequelle (Mikrofon) zur Schallquelle ab. Jeder Raum beeinflusst mit seinen geometrischen Eigenschaften, der Ausstattung und der Struktur der Reflexionsflächen das Signal der Quelle bis zum Auftreffen auf die Aufnahmequelle. Die Raumakustik kann im Wesentlichen mit der Raumimpulsantwort beschrieben werden und definiert sich als Verhältnis zwischen dem Aufnahmesignal und dem ausgesendeten Signal der Schallquelle.

Mathematisch ist das mit einem Mikrofon aufgenommene Audiosignal y(t) eine Komposition aus dem Signal der Schallquelle x(t) und der Raumimpulsantwort h(t). Die Berechnung erfolgt im Zeitbereich in Form einer Faltung und im Frequenzbereich in Form einer Multiplikation der Spektren der Signale (3.3).

Die folgenden Formeln zeigen die Berechnung in den oben genannten Bereichen Zeit und Frequenz:

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau) \cdot h(t-\tau) d\tau) \tag{3.3}$$

mit
Signal der Schallquelle x(t)
Aufgenommenen Audiosignal y(t)
Raumimpulsantwort h(t)

Durch eine Fouriertransformation in den Frequenzbereich lässt sich die Raumimpulsantwort mathematisch leichter ermitteln, da eine Faltung im Zeitbereich einer Multiplikation im Frequenzbereich (3.4) entspricht:

Ausgangsspektrum = Eingangsspektrum · Frequenzgang des Raumes

$$Y(f) = X(f) \cdot H(f) \tag{3.4}$$

Daraus resultieren die folgenden Rechenoperationen:

Fouriertransformationen des Eingangssignals und des gemessenen Ausgangssignals:

X(f) ●—○ x(t)
Y(f) ●—○ y(t)

Division im Frequenzbereich (3.5):

$$H(f) = \frac{Y(f)}{X(f)} \tag{3.5}$$

Rücktransformation in den Zeitbereich:

h(t) ○—● H(f)

Eine Methode zur Ermittlung einer Impulsantwort x(t) als Ausgangssignal des Systems ist das Aussenden eines Stoßsignals (Dirac) in einem Raum. Eine weitere Möglichkeit ist das Aussenden einer Sprungfunktion oder das Ausschalten Weißen Rauschens mit einer begleitenden Messung der Sprungantwort (Abb. 3.2).

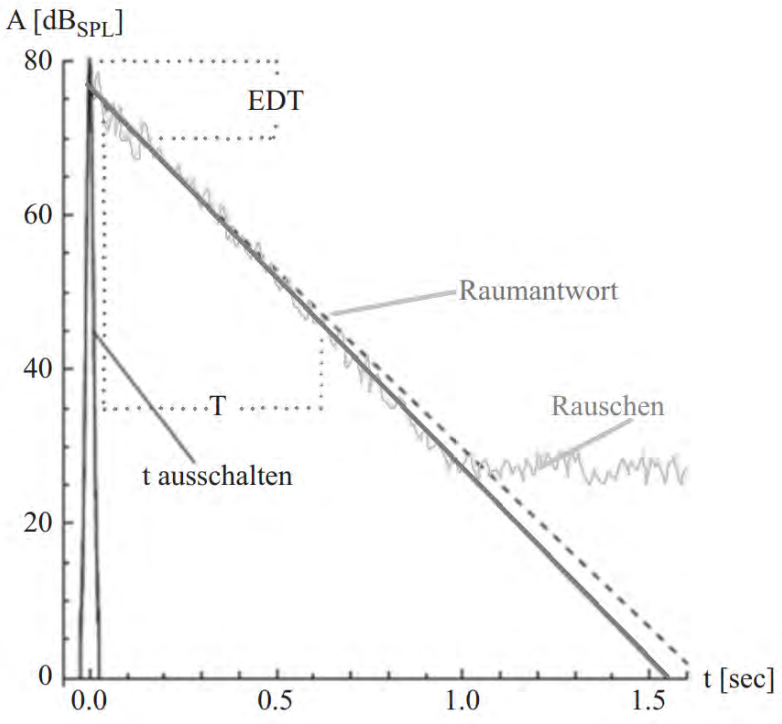


Abbildung 3.2: Raumantwort nach Ausschalten Weißen Rauschens
Quelle: Tim Ziemer Psychoakustische Schallfeldsynthese für Musik [25]

Diese Methoden sind geeignet, die Raumimpulsantwort von jedem Raum messtechnisch zu beschreiben.

Die Ermittlung der Raumimpulsantwort als mathematische Funktion bietet damit die Möglichkeit, auf vorhandene Datenbanken mit umfangreichen Geräusch- und Sprachsignalen zuzugreifen und diese Signale auf die individuelle Raumakustik des Wohnumfelds anzupassen. Dies erspart den Aufwand einer Generierung von individuellen Trainingsdaten. Die Datensätze der Trainingsdatenbank können mit der Raumimpulsantwort verrechnet werden, um Deep Learning Modelle zur Audiosignalerkennung unter Berücksichtigung der Raumakustik zu trainieren.

Ein großer Nachteil ist der messtechnische Aufwand zur Ermittlung der Raumimpulsantwort in einem Wohnbereich älterer Menschen unter praktikablen Aspekten. Jeder Raum muss zur Einmessung mit einer speziellen Ausrüstung bestehend aus Sender (Lautsprecher) und Aufnahmequelle (Mikrofon) ausgestattet werden. Danach erfolgt mittels einer speziellen Software die Bestimmung der Raumimpulsantwort. Dieses Verfahren erfordert Fachkenntnisse und eine besondere technische Ausstattung.

Ansatz 2:
Generierung eigener Trainingsdaten im Wohnumfeld ohne Ermittlung der Raumakustik

Ein alternativer Ansatz ist, die oben beschriebene mathematische Ermittlung der Raumimpulsantwort unberücksichtigt zu lassen und an der Stelle von vorhandenen Trainingsdatenbanken eine individuelle, Wohnumfeld bezogene Datenbank aufzubauen. Diese Methode berücksichtigt sehr spezifisch Geräusche und Sprache, die im Wohnumfeld der betroffenen Menschen auftauchen. Die raumakustischen Eigenschaften sind dabei Bestandteil des Audiosignals, das die Aufnahmequelle empfängt. Ein Nachteil dieser Methode ist, dass die Trainingsdaten im Wohnumfeld mit Hilfe von Originalaufnahmen (Audiorohdaten) für jede Klasse und in einer großen Menge erzeugt werden muss.

Für beide Ansätze sind somit vorbereitende Verfahren im Wohnumfeld der betroffenen Personen notwendig, um eine Geräusch- und Spracherkennung zu implementieren.

Zusätzlich zu beiden Methoden ist die Positionierung der Aufnahmequellen als auch die Veränderung der Entfernung durch Bewegung der Schall- und/oder der Aufnahmequelle und deren Einfluss auf das aufgenommene Signal zu bewerten:

3.1.3 Positionierung der Aufnahmequellen

Bei einer Aufnahmequelle im Raum, die fest positioniert ist, werden Aufnahmen von unbeweglichen Geräuschquellen immer mit einem konstanten Anteil der raumakustischen Eigenschaften verändert. Tabelle 3.1 stellt die vier Kombinationen gegenüber und bewertet, ob sich das Audiosignal verändert, wenn es auf die Aufnahmequelle trifft.

Tabelle 3.1: Positionierung Schall- und Aufnahmequelle (Mikrofon) und Veränderung des Signals

Schallquelle	**Beispiel**	**Mikrofon**	**Beispiel**	**Aufnahmesignal**
fest	Radio	fest	stationäres Mikrofon	konstant
beweglich	Mensch	fest	stationäres Mikrofon	veränderlich
fest	Radio	beweglich	Mikrofon am Körper	veränderlich
beweglich	Mensch	fest	Mikrofon am Körper	konstant

Aus der Tabelle ist zu ersehen, dass sich empfangene Audiosignale gegenüber ausgesendeten Signalen durch die Raumakustik nicht verändern, wenn der Abstand zwischen der Schallquelle und der Aufnahmequelle konstant bleibt. Dies ist der Fall für Geräusche aus fest installierten Quellen zu einem fest installierten Mikrofon oder für Sprachsignale, wenn das Mikrofon am Körper der sprechenden Person befestigt ist.

Ein Mikrofon als bewegliche Aufnahmequelle am Körper eines Menschen bietet viele Vorteile bei der Erfassung der Schallquellen. Die Sprachsignale der Person sind aufgrund der kurzen Entfernung zum Mikrofon deutlich und werden nur gering durch die Raumakustik oder andere Schallquellen wie Musik aus dem Radio beeinflusst. Eine korrekte Erkennung von Sprache und Worten hat gegenüber der Erkennung von Geräuschen insbesondere bei Gefahrensituationen oder Bedarf von Hilfe Vorrang. Damit wäre die Befestigung eines Mikrofons am Körper der überwachten Person erstrebenswert.

Dem stehen zwei signifikante Nachteile gegenüber. Bewegliche Mikrofone bestehen aus der Mikrofonkapsel, einer Verstärkereinheit und einem drahtlosen Sender. Damit benötigen sie eine wiederaufladbare Spannungsversorgung (Akku). Ein nicht geladener Akku führt zu einer Fehlfunktion des Gerätes. Ein zweiter Nachteil ist der Umstand, dass das Mikrofon dauerhaft am Körper getragen werden muss. Sollte das Gerät durch die Person nicht angelegt und eingeschaltet sein, erfolgt keine Aufnahme des Audiosignals [26].

3.1.4 Konfiguration auf Basis der beschriebenen Rahmenbedingungen

Eine Abwägung der Vor- und Nachteile der oben genannten Methoden und Ansätze führt zu der folgenden Entscheidung hinsichtlich der Konfiguration im Wohnumfeld älterer Menschen:

- **Raumgröße**
  Die Raumgröße sollte eine Fläche von 20 m² nicht überschreiten.
  Es wird ein Mikrofon pro Raum eingesetzt, um eine wechselseitige Beeinflussung mehrerer Aufnahmequellen auszuschließen.
  Das Mikrofon sollte zentral im Raum positioniert werden, um im Bereich des Hallradius unterschiedlicher Schallquellen zu sein.

- **Raumakustik**
  Raumimpulsantworten werden nicht ermittelt. Die erforderlichen Trainingsdaten werden im Wohnumfeld der älteren Menschen durch die Aufnahme von Audiorohdaten für festgelegte Geräusche und Sprache generiert.

- **Positionierung**
  Aufgrund der geringen Akzeptanz zum Tragen eines Mikrofons am Körper und dem Risiko, dass der Akku nicht geladen ist, wird festgelegt, pro Raum ein fest installiertes Mikrofon vorzusehen.

3.2 Hard- und Softwareauswahl zur Audiosignalaufzeichnung

Die erste Stufe in der Prozesskette der Geräusch- und Spracherkennung ist die Audiosignalaufzeichnung im betroffenen Umfeld. Dieser Abschnitt beschreibt einen Wohnbereich mit seinen Merkmalen, verschiedene Mikrofonsysteme als auch die Software zur Aufnahme und zum Abspeichern der Audiorohdaten. In Abbildung 3.3 ist exemplarisch ein Wohnbereich und die Positionierung von Aufnahmequellen in verschiedenen Räumen dargestellt.

3.2.1 Übersicht

Abbildung 3.4 zeigt ein vereinfachtes Blockschaltbild der Prozesskette zur Darstellung der Hardwarekomponenten. Ein Mikrofon erfasst das Geräusch im Raum und sendet das aufgenommene Signal über WLAN an einen, im Wohnbereich aufgestellten, zentralen Computer,

Abbildung 3.3: Darstellung eines Wohnraumes und Lokalisierung von Aufnahmequellen
Quelle: Eigene Darstellung mit Vorlage [27]

der die Extraktion und Erkennung durchführt. Das daraus resultierende Ergebnis kann über das Internet an Smartphones und andere Ausgabegeräte zum Beispiel in Form einer Statusmeldung übertragen werden.

Abbildung 3.4: Vereinfachtes Blockschaltbild Konfiguration,
Quelle: Eigene Darstellung

Nachfolgend wird die Hardware der Aufnahmequellen und des zentralen Computers zur Auswertung der Audiosignale als auch die verwendete Software beschrieben.

3.2.2 Hardwarekomponenten

Die Audioaufnahme im Wohnumfeld kann mit verschiedenen Aufnahmegeräten durchgeführt werden. Ziel ist es, dass das Mikrofon und die nachfolgende digitale Wandlung in eine

Audiodatei Anforderungen erfüllt, das Geräusch oder die Sprache mit Hilfe eines CNN-Modells zu erkennen. Dies setzt eine ausreichende Quantisierung und Abtastrate voraus. Der Frequenzbereich muss aufgrund der Anforderung, Geräusche zu unterscheiden, ein großes Frequenzspektrum abdecken. Im Gegensatz dazu würde bei der Anforderung einer reinen Spracherkennung aufgrund des kleineren Frequenzspektrums eine geringere Bandbreite und damit korrelierende Abtastrate erforderlich sein.

Im Folgenden werden verschiedene Aufnahmequellen bewertet, die im Rahmen dieser Arbeit eingesetzt werden. Unter Berücksichtigung von Kosten und einem geringen Installationsaufwand eignen sich Miniaturmikrofone mit integriertem Analog-Digital-Wandler und Funksender. Im Kapitel 2.1.1 werden die Mikrofone und ihre mögliche Eignung beschrieben. Im Kapitel 4 werden die Ergebnisse einer Studie dargestellt, inwieweit die gewählten Aufnahmequellen die Anforderungen zur Geräusch- und Spracherkennung erfüllen.

Alle Aufnahmequellen senden kontinuierlich ihre aufgenommenen Signale an einen zentralen, im Wohnumfeld installierten Computer. Das Abspeichern der aufgenommenen Signale erfolgt auf diesem Computer als Empfangsquelle der digitalen Audiodaten. Der Computer beinhaltet ebenfalls die Trainingsdatenbank und das CNN-Modell zur Geräusch- und Spracherkennung. Eine kabelgebundene Verdrahtung von installierten Mikrofonen zu dem zentralen Computer wird in dieser Arbeit nicht betrachtet. Bei der Installation im Wohnumfeld wird vorausgesetzt, dass ein WLAN-Netz zur Verfügung steht oder ein Access-Point zum Computer installiert wird. Um verschiedene Aufnahmequellen in ihrer Eignung zu bewerten, wird zum Vergleich ein Kondensatormikrofon mit einem Audiointerface und einem angeschlossenen Computer als Referenz eingesetzt.

Die Aufnahmequellen werden im Raum fest installiert. Für die Mikrofone ist eine Spannungsversorgung in Form einer 230V-Steckdose erforderlich. Die Entfernung zu den verschiedenen Schallquellen ist abhängig von der Position und Beweglichkeit der jeweiligen Schallquelle und kann veränderlich oder fest sein. Zu beweglichen Schallquellen gehören Personen oder Tiere im Raum als auch bewegliche Geräte wie Staubsauger.

Im Rahmen dieser Arbeit werden folgende Hardwarekomponenten eingesetzt:

MEMS-Mikrofon und ESP32-Mikrocomputer

Die wesentlichen Anforderungen sind ohne Berücksichtigung einer Priorität:

- geringe Baugröße
- geringe Kosten
- AD-Wandler mit einer Abtastrate über 20 kHz und einer Bitrate ≥ 16 bit
- WLAN-fähig

MEMS-Mikrofone wie das INMP441 haben geringe Abmessungen (Abb. 3.5), besitzen einen integrierten AD-Wandler und einen digitalen I²S-Bus und erfüllen die Anforderungen einer Aufnahme zur Geräuscherkennung in Kombination mit einem Mikrocomputer wie den ESP32. Der Mikrocomputer dient dabei zur Speicherung und drahtlosen Übertragung der digitalen Daten. Alternativ kann ein MEMS-Mikrofon ohne AD-Wandler eingesetzt werden und die Wandlung durch den integrierten AD-Wandler des Mikrocomputers erfolgen.

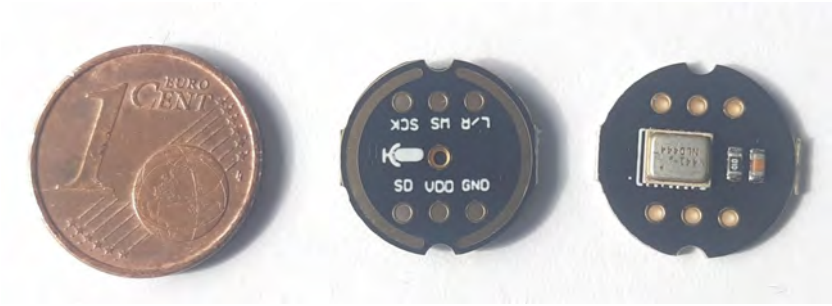


Abbildung 3.5: INMP441 MEMS-Mikrofone im Größenvergleich

Vorversuche, die in diesem Kapitel im Abschnitt Implementierung beschrieben werden, sowie Recherchen im Internet [28] haben ergeben, dass die Signalerfassung mit dem MEMS-Mikrofon, AD-Wandlung, direkte Speicherung in den Mikrocomputer eigenen SPIFFS-Speicher und nachfolgende Datenübertragung bereits zeitkritisch bei Abtastraten von über 24 kHz sind, obwohl die Speicherung im Mikrocomputer mit einem DMA-Controller ohne CPU-Unterstützung erfolgt. Daher wurde in dieser Arbeit eine Auswahl eines analogen MEMS-Mikrofons mit AD-Wandlung im ESP32-Mikrocomputer verworfen.

Der ESP32-Mikrocomputer verfügt neben einem WLAN-Modul über einen internen Speicher von 1,5 MB. Bei einer Abtastrate von 24 kHz kann eine Audiodatei mit einer Länge von 20 Sekunden abgespeichert werden. Der Mikrocomputer verfügt zusätzlich über eine Bluetooth-Schnittstelle, die beispielsweise zur Anwesenheitsüberwachung eingesetzt werden kann. Die Programmierung des Mikrocomputers erfolgt in der Sprache Python in einer Arduino-IDE. Es besteht im Internet eine umfangreiche Sammlung an Bibliotheken zur Einbindung von MEMS-Mikrofonen, WLAN als auch Bluetooth-Anwendungen.

Die technischen Daten des MEMS-Mikrofons INMP441 als auch des ESP32-Mikrocomputers finden sich im Literaturverzeichnis [29] [30]. Der Versuchsaufbau ist in der Abbildung 3.6 dargestellt.

Medion Android Tablet

Wie im Kapitel 2.1.1 beschrieben, beinhalten Smartphones und Tablets ebenfalls Miniaturmikrofone, einen integrierten AD-Wandler als auch die Möglichkeit der digitalen und kabel-

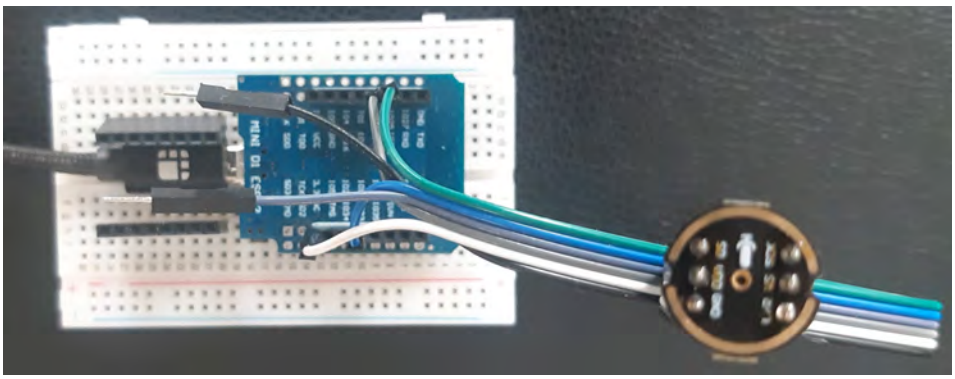


Abbildung 3.6: Versuchsaufbau MEMS-Mikrofon und ESP32-Mikrocomputer

losen Signalübertragung. Nachteilig ist eine aufwendigere Programmierung in der Android-Umgebung als bei Mikrocomputern. Weiterhin sind Parametrierungen wie Abtastraten und Bittiefe nicht möglich.

Ein im Kapitel 2.5.1 vorgestelltes Systembeispiel nutzt ein Samsung Tablet als Intelligenten Bilderrahmen zur Stimmungsanzeige. Die Kombination einer Geräusch- und Sprachüberwachung mit einem digitalen Bilderrahmen bietet, abgesehen von den oben aufgeführten Nachteilen, die Vorteile eines einfachen Systems zur Audiosignalerfassung im Wohnumfeld. Tablets erfüllen weiterhin die Anforderungen einer hohen Abtastrate von 44,1 kHz als auch der Möglichkeit, Daten über WLAN an einen zentralen Computer oder in eine Cloud zu senden.

Ein Tablet im Wohnzimmer in Kombination mit MEMS-Mikrofonen in den anderen betrachteten Räumen im Wohnumfeld älterer Menschen, bietet eine sinnvolle Kombination zur Geräusch- und Spracherkennung. Das System des Intelligenten Bilderrahmens BEJOY der Firma Beyond Emotion verwendet ein Tablet der Marke Samsung. Für die im Kapitel 4.2.2 durchgeführten Versuche wird ein Tablet der Firma Medion eingesetzt. Es wird keine Programmierung des Tablets vorgenommen und auf verfügbare Apps zur Aufzeichnung und Speicherung von Audiosignalen zurückgegriffen.

Die technischen Daten des Medion-Tablets finden sich im Literaturverzeichnis [31].

Kondensatormikrofon und PC mit Audiointerface

Als Referenzmessung zu den Miniaturmikrofonen des Tablets und dem IMNP441, die zur Geräusch- und Spracherkennung im Wohnumfeld älterer Menschen verwendet werden sollen, wird im Rahmen der experimentellen Studie im Kapitel 4 das Kondensatormikrofon MXL 990 der Firma Marshall Electronics, Inc. eingesetzt.

Das Mikrofon ist ein Großmembranmikrofon. Aufgrund seines großen Frequenzbereichs, geringen Geräuschpegels als auch hohen Signal/Rauschabstands eignet es sich als Referenzgerät zu den Miniaturmikrofonen. Durch seine Baugröße, der analogen Signalverarbeitung und kabelgebundener Anbindung an Verstärker ist ein Einsatz zur Geräuscherkennung im Wohnumfeld nicht sinnvoll.

Die Digitalisierung der Mikrofonaufnahmen erfolgt mit dem PC-Audiointerface U-Phoria UMC 404 der Marke Behringer, das mit einem gängigen Laptop der Marke ASUS verbunden ist. Das Audiointerface überträgt die digitalen Signale über einen USB-Anschluss zum Laptop. Das Audiointerface bietet eine Abtastrate bis zu 192 kHz bei einer Bitrate von 24 Bit und stellt somit eine verlustlose Wandlung der Audioaufnahme sicher.

Die technischen Daten des Kondensatormikrofons und des Audiointerfaces finden sich im Literaturverzeichnis [32] [33].

Zentraler Computer im Wohnumfeld

Im Rahmen dieser Arbeit und der damit verbundenen Studie, deren Ergebnisse im Kapitel 4 beschrieben werden, wird als zentraler Computer für die Datenvorverarbeitung zur Erstellung der Audiorohdaten als auch für die Ausführung der CNN-Modelle ein ZenBook Laptop der Marke ASUS verwendet. Der Computer verfügt über die notwendigen Schnittstellen wie USB und WLAN und liefert mit dem Intel Core i7-Prozessor eine angemessene Rechenleistung. Die übertragenen Audiodaten als auch die Modelle und Trainingsdaten sind sowohl auf der Festplatte des Computers als auch in einer Cloud gespeichert. Alternativ bietet sich als zentraler Computer im Wohnumfeld älterer Menschen ein auf die Berechnung von CNN-Modellen abgestimmtes System wie NVIDIA Jetson Embedded AI an, da es als System-on-Module (SOM) mit GPU, CPU, Speicher, Energiemanagement und Hochgeschwindigkeitsschnittstellen speziell auf KI-Anwendungen abgestimmt ist [34].

Die Datenübertragung mit dem Referenzmikrofon erfolgt, wie oben beschrieben, über das Audiointerface, dass das analoge Mikrofonsignal digitalisiert und kabelgebunden an den Computer sendet.

Im Gegensatz dazu erfolgt die Datenübertragung der Audioaufnahme mit dem MEMS-Mikrofon und dem zugehörigen ESP32-Mikrocomputer über die integrierte WLAN-Schnittstelle der Geräte. Das Tablet speichert die Aufnahmen mit einer App-Software in einer Cloud. Diese Daten werden wiederum vom zentralen Computer ausgelesen.

Die Ergebnisse der Geräusch- und Spracherkennung zum Beispiel in Form von Warnmeldungen oder einem Ampel-Status können, wie in der Zielsetzung in Kapitel 1.3 beschrieben, von dem zentralen Computer über das Internet an einen vernetzten Computer oder ein Smartphone gesendet werden. Das Verfahren im Kontext mit altersgerechten Assistenzsystemen

und die damit verbundene technische Umsetzung würden über den Rahmen hinausgehen und werden in dieser Arbeit nicht betrachtet.

Die technischen Daten des Laptops finden sich im Literaturverzeichnis [35].

Blockschaltbild des Gesamtsystems

Die folgende Abbildung 3.7 zeigt zusammenfassend ein Blockschaltbild der Hardwarekomponenten Aufnahmequelle (Mikrofon, Verstärker, AD-Wandler und digitaler Signalverarbeitung), den Computer (Empfangsquelle der Audiodaten und der Geräusch- und Spracherkennung mit CNN-Modell) sowie einem Smartphone zur Anzeige einer Normal- oder Gefahrensituation auf Basis des erkannten Geräusches.

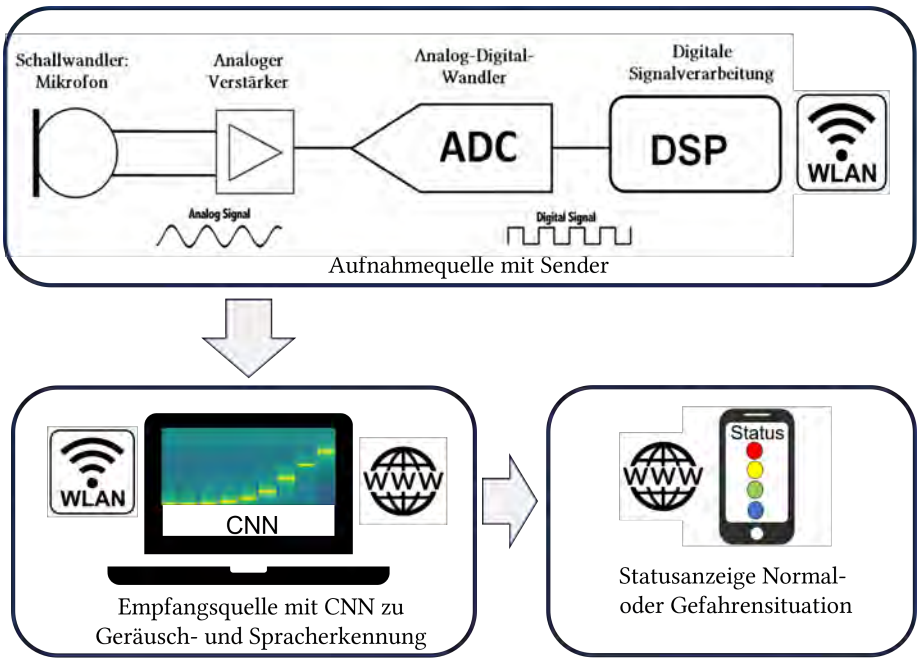


Abbildung 3.7: Blockschaltbild Konfiguration
Quelle: Eigene Darstellung

3.2.3 Software für die eingesetzten Hardwarekomponenten

Im Folgenden werden die verschiedenen Softwareprogramme mit ihren Funktionalitäten beschrieben. Dies beinhaltet die Programme für die folgenden Hardwarekomponenten:

- MEMS-Mikrofon mit ESP32-Mikrocomputer
- Tablet
- Laptop als zentraler Rechner

Aufnahme und Datenübertragung mit dem MEMS-Mikrofon und dem ESP32-Mikrocomputer

Zur Vermeidung von Interferenzen zwischen der Audioaufnahme und der nachfolgenden Datenübertragung der aufgenommenen WAV-Datei werden zwei getrennte Programme für den Mikrocomputer eingesetzt.

Programmcode 1: Audioaufnahme
Im Internet bieten verschiedene Programmbibliotheken ein Grundgerüst zur Programmentwicklung einer Audioaufnahme mit einem MEMS-Mikrofon und einem ESP32-Mikrocomputer. Die Programmierung erfolgt in einer Arduino-IDE. Basis im Rahmen dieser Arbeit ist der Programmcode ESP32-INMP441-RECORDING [36] mit dem dazugehörigen Youtube-Video [37] als Tutorial.

Der durch den Autor modifizierte Programmcode hat den nachfolgenden Ablauf:

1. Einbindung der Treiberbibliotheken
2. Pin-Belegung der Hardwareanschlüsse des ESP32
3. Parametrierung wie Abtast- und Bitrate
4. Setup mit SPIFFS-Speicher- und I²S-Schnittstellenkonfiguration
5. Parametrierung wie Abtast- und Bitrate
6. Übertragung der Datensignale über die I²S-Schnittstelle
7. Speichern der Signale im SPIFFS-Speicher im WAV-Format

Programmcode 2: Datenübertragung auf einen Webserver
Ein zweiter Programmcode liest die im SPIFFS-Speicher abgelegte WAV-Datei aus dem Speicher und sendet die Datei mit einem UDP-Protokoll über WLAN in das Netzwerk. Das UDP-Protokoll hat gegenüber dem HTTP-Protokoll den Vorteil der schnelleren Datenübertragung.

Der Programmcode basiert auf ein FS-Webserver-Template von Gochkov [38] und wurde vom Autor an die Anwendung zur Übertragung der Audiodaten im WAV-Format angepasst.

Der Programmcode hat den nachfolgenden Ablauf:

1. Einbindung der Treiberbibliotheken wie WiFi und SPIFFS
2. Definition der Einwahlparameter für das WLAN
3. Bestimmung der Datengröße und des Dateityps
4. Setup mit SPIFFS-Speicher- und I²S-Schnittstellenkonfiguration
5. Auslesen der Datei aus dem SPIFFS-Speicher
6. Übertragung der Datei an den Webserver

Mit dem Aufruf der Webserver-IP-Adresse und dem Dateinamen wird die auf dem ESP32-Mikrocomputer gespeicherte Audiodatei auf den zentralen Computer heruntergeladen und abgespeichert.

Die beiden oben beschriebenen Programmcodes finden Anwendung bei der Audioaufnahme zur Generierung der Trainingsdaten (Datenvorverarbeitung) als auch zur Aufnahme von Geräuschen und Sprache im Rahmen der Implementierung.

Audioaufnahme mit dem Tablet

Für das Tablet mit Android-Betriebssystem wird zur Aufnahme der Audiorohdaten und der Geräusche und Sprache im Wohnumfeld die App-Anwendung Hi-Q MP3 Recorder von Audiophile [39] verwendet. Die Auswahl begründet sich auf Basis der folgenden Funktionalitäten, die im Rahmen dieser Arbeit erforderlich sind:

- Audioaufnahmen mit 44,1 kHz Abtastrate
- Dateiformate: WAV, MP3 und weitere
- Speichern der Aufnahme auf dem Tablet
- Paralleles Speichern der Aufnahme in einer Cloud wie Google Drive
- Variable Anpassung der Eingangssignallautstärke

Das parallele Abspeichern der Aufnahme in einer Cloud ermöglicht den einfachen Zugriff zur Weiterverarbeitung der Daten in einem CNN-Modell.

Audioaufnahme und Datenverarbeitung auf dem zentralen Rechner

Zur Bearbeitung der Audiorohdaten insbesondere der Anpassung der Länge der Aufnahme und der Veränderung der Abtastrate wird die kostenlose, quelloffene und plattformübergreifende Software Audacity [40] verwendet. Audacity wird ebenfalls als Programm für die Audioaufnahme mit dem Kondensatormikrofon und dem Audiointerface eingesetzt.

3.3 Festlegen der Audioklassen

Wie bereits im Kapitel 2.4.1 “Deep Learning Begrifflichkeiten” als auch in diesem Kapitel im Abschnitt “Berücksichtigung der Raumakustik” beschrieben, benötigt das überwachte Lernen bekannte Zusammenhänge, die als Trainings-, Test- und Validierungsdaten genutzt werden. Ziel ist es, eine hohe Prognose für die Zuordnung einer neuen Eingabe - zum Beispiel einer Aufnahme eines Audiosignals - zu einer Klassifikation zu finden.

Bestandteil dieser Arbeit ist die Festlegung von Audioklassen für Geräusche und für Sprache im Wohnumfeld von älteren Menschen. Jede Klasse beinhaltet eine Vielzahl von aufgenommenen Audiosignalen eines speziellen Geräusches oder eines gesprochenen Wortes in Form eines Musters. Diese Dateien bilden mit ihrer Zuordnung zu den Klassen die Datenbank der Trainingsdatensätze. Eine neue Audioaufnahme im Wohnumfeld wird in einem CNN-Modell als Eingabe mit den Trainingsdatensätzen verglichen und einem Muster / einer Klasse mit einer Genauigkeit zugeordnet.

Die Zuordnung und damit Erkennung dieser Geräusche oder Sprache dient der Fallunterscheidung zwischen normalen Alltagssituationen und einer Gefahren- oder Notfallsituation als auch der Möglichkeit, dass die ältere Person über das System aktiv verbal Hilfe anfordert. Unter normalen Alltagssituationen werden Geräusch- und Sprachsignale verstanden, die zum gewöhnlichen Alltag gehören. Gefahren- und Notfallsituationen sind außergewöhnliche Ereignisse, die eine Unterstützung und Hilfe der älteren Person im Wohnumfeld erfordern.

Nachfolgend werden Geräusch- und Sprachsignale raumbezogen festgelegt und Klassen zugeordnet. Die Anzahl der Signale decken ein breites Spektrum der Audiosignale im Wohnumfeld von älteren Menschen ab. Nach der Festlegung der Audioklassen erfolgt eine kurze Beschreibung der Besonderheiten und der Merkmale der einzelnen Klasse.

Die folgende Tabelle 3.2 ordnet die definierten Audioklassen verschiedenen Räumen im Wohnumfeld zu. Mit insgesamt 20 in dieser Arbeit definierten Klassen fallen auf den Bereich Wohnzimmer 13 verschiedene Klassen.

Aufgrund der Komplexität wurde das Geräusch durch Hinfallen einer Person ausgeschlossen. In diesem Zusammenhang wird, wie in Kapitel 2.5 beschrieben, auf verschiedene Veröffentlichungen zum Beispiel die Studie von Zigel [20] als auch auf Kapitel 5.2 verwiesen.

In der Übersicht 3.3 werden Merkmale und Eigenschaften den Audioklassen zugeordnet und eine Handlungsableitung im Falle einer Gefahrensituation beziehungsweise bei Hilfebedarf festgelegt. Aus den genannten Eigenschaften verschiedener Situationen lassen sich Merkmale unterschiedlicher Audiosignale ableiten. Diese können durch Extraktion und Wandlung zum Beispiel in Spektrogrammen sichtbar gemacht werden und ein Unterscheidungsmerkmal zu anderen Klassen sein.

Tabelle 3.2: Definition und Zuordnung Audioklassen zu Räumen

lfd. Nr.	Audioklasse	Wohnen	Küche	Schlafen	Flur	Bad	Wirtschaftsr.
01	Gespräch	X	X	X	X	X	X
02	Hallo-Ruf	X	X	X	X	X	X
03	Hilfe-Ruf	X	X	X	X	X	X
04	Festnetz-Telefon	X		X	X		
05	Mobil-Telefon	X	X	X	X		
06	Geschirr	X	X				
07	Schritte	X	X	X	X	X	X
08	Musik-Radio	X	X				
09	Zeitung lesen	X	X	X			
10	Tür auf/zu	X	X	X	X	X	X
11	Rauchmelder	X		X	X		X
12	Wasserkocher		X				
13	Staubsauger	X	X	X	X		X
14	WC-Spülung					X	
15	Duschen					X	
16	Hände waschen		X			X	
17	Föhn					X	
18	Tür klopfen				X		
19	Türklingel				X		
20	Hundegebell	X	X	X	X	X	X
	Summe	**13**	**13**	**11**	**12**	**10**	**8**

Tabelle 3.3: Merkmale und Eigenschaften von definierten Audioklassen

lfd. Nr.	Klasse	Merkmal[1]	Eigenschaft	Spektrum[2]	Handlungs-ableitung
01	Gespräch	kontinuierlich	monoton	schmal	
02	Hallo-Ruf	kurz	2-silbig	schmal	Meldung
03	Hilfe-Ruf	kurz	2-silbig	schmal	Alarm
04	Festnetz-Telefon	mittellang	melodisch	breit	
05	Mobil-Telefon	mittellang	melodisch	breit	
06	Geschirr	kurz	hell	schmal	
07	Schritte	kurz	leise	schmal	
08	Musik-Radio	kontinuierlich	melodisch	breit	
09	Zeitung lesen	kurz	leise	schmal	
10	Tür auf/zu	kurz	dynamisch	schmal	
11	Rauchmelder	kurz	monoton	schmal	Alarm
12	Wasserkocher	kontinuierlich	monoton	schmal	
13	Staubsauger	kontinuierlich	monoton	mittel	
14	WC-Spülung	mittellang	veränderlich	mittel	
15	Duschen	kontinuierlich	monoton	mittel	
16	Hände waschen	mittellang	monoton	mittel	
17	Föhn	kontinuierlich	monoton	breit	
18	Tür klopfen	kurz	monoton	schmal	
19	Türklingel	kurz	monoton	schmal	
20	Hundegebell	kurz	dynamisch	schmal	

[1] Merkmal:
Ein kurzes Signal wird in dieser Arbeit mit einer Dauer von bis zu einer Sekunde angesehen. Dabei können Signalanteile kürzer als eine Sekunde sein. Ein mittellanges Signal geht über wenige Sekunden mit dem Merkmal, dass es danach endet.

[2] Spektrum:
Die Breite des Spektrums wird unterschieden in ‘schmal’ für Signale wie die Sprache einer Person, ‘mittel’ für Signale, die tiefe bis hellere Töne abdecken als auch ‘breit’ für melodische, polyphone Telefonsignale mit Obertönen. Dabei wird an dieser Stelle keine feste Frequenzbreite definiert, sondern auf die Darstellung spektraler Unterschiede in Abschnitt 3.4.4 verwiesen.

3.4 Implementierung

Nach der Festlegung der Rahmenbedingungen, der Auswahl der Hard- und zugehörigen Software sowie der Bestimmung der Audioklassen wird nachfolgend die Implementierung der Methoden entlang der Prozesskette beschrieben.

Diese lässt sich in die folgenden Schritte aufteilen:

1. Aufnahme der Audiorohdaten zur Generierung der Trainingsdaten
2. Generierung der Trainingsdaten
3. Audioaufnahme im Wohnumfeld zur Geräusch- und Spracherkennung
4. Extraktion und Aufbereitung der Audiosignale
5. CNN-Modelle und deren Parametrierung

Die Aufnahme der Audiorohdaten und die einmalige Generierung der Trainingsdaten sind Bestandteil der Datenvorverarbeitung (Abb. 3.8). Im Rahmen der Konzeptionierung wurde im oberen Teil dieses Kapitels entschieden, die definierten Audioklassen im Wohnumfeld der älteren Menschen aufzunehmen und nicht auf existierende Datenbanken von Geräuschen und Sprache zurückzugreifen.

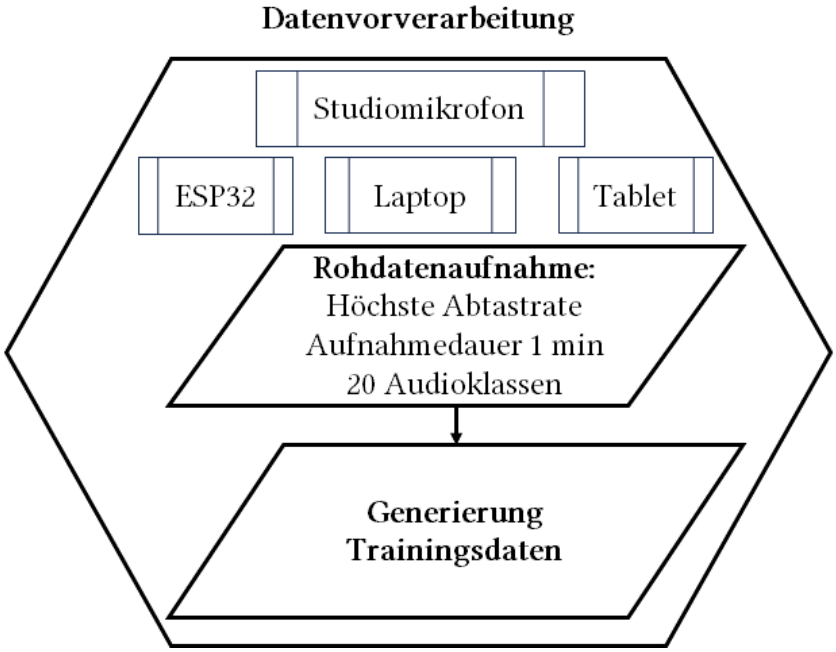


Abbildung 3.8: Datenvorverarbeitung mit Audioaufnahme der Rohdaten und Generierung der Trainingsdaten
Quelle: Eigene Darstellung

Nachdem die Trainingsdaten erstellt sind, erfolgt die Audioaufnahme, Extraktion und Klassifizierung mit CNN-Modellen als Hauptprozess (Abb. 3.9).

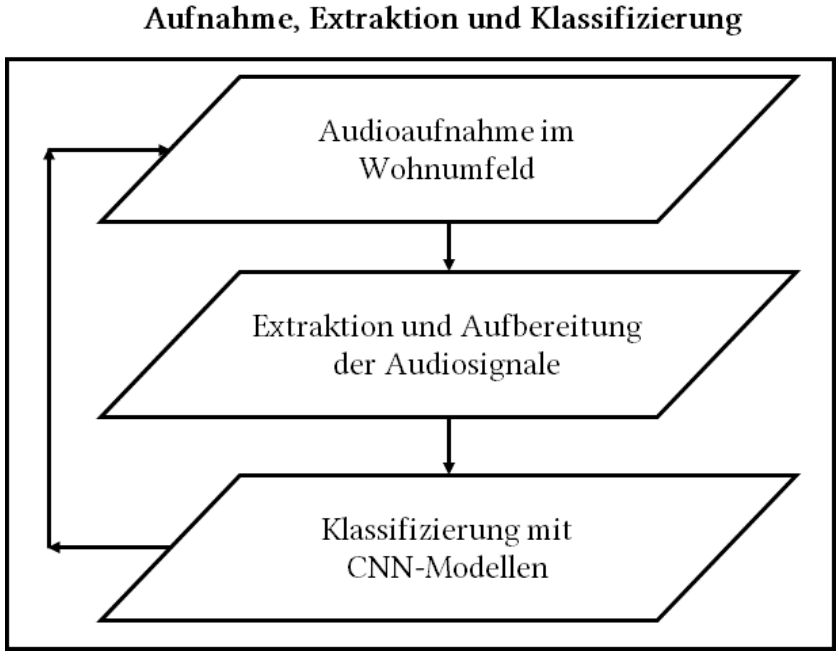


Abbildung 3.9: Hauptprozess Geräusch- und Spracherkennung mit Aufnahme, Extraktion und Klassifizierung
Quelle: Eigene Darstellung

Abschließend werden in diesem Kapitel verschiedene Störgrößen während der Audiosignalaufnahme beschrieben und ihre potentiellen Auswirkungen auf die Erkennung von Geräuschen und Sprache bewertet.

3.4.1 Aufnahme der Audiorohdaten zur Generierung der Trainingsdaten

Die Aufnahme der Audiorohdaten erfolgt mit der im Abschnitt 3.2.2 genannten Hardware und ihrer zugehörigen Software. Nachfolgend werden die Aufnahmen mit den Geräten

- MEMS-Mikrofon mit ESP32-Mikrocomputer
- Tablet mit integriertem Mikrofon
- Kondensatormikrofon mit digitalem Audiointerface und PC
- integriertes Laptop-Mikrofon

und die Parametrierung der eingesetzten Software beschrieben.

Für alle Konfigurationen erfolgt die Aufnahme von Audiorohdaten mit der größten technisch möglichen Abtastrate, aber nicht höher als 44,1 kHz. Hintergrund ist die Tatsache, dass sowohl das integrierte Laptop-Mikrofon als auch das Tablet Audiosignale mit einer fest eingestellten Abtastrate von 44,1 kHz aufnehmen. Wie im Kapitel 2.1.2 beschrieben, wird mit dieser Abtastrate ein Frequenzspektrum von 22 kHz Bandbreite korrekt abgebildet.

Die für die Aufnahmen gewählte Raumgröße ist 18 m². Die Aufnahmedauer beträgt eine Minute. Voruntersuchungen zu dieser Arbeit haben ergeben, dass über diesen Aufnahmezeitraum eine signifikante Menge an Trainingsdaten erzeugt werden können. Für den Fall, dass die Audiorohdaten sequenziell für alle Klassen im Wohnumfeld älterer Menschen aufgenommen werden, ist eine Aufnahmedauer von einer Minute pro Klasse aus Sicht des Autors vertretbar. Kurze Signale wie ein Hilferuf oder Telefonklingeln werden über die gesamte Aufnahmeperiode wiederholt. Das aufgezeichnete Signal wird im digitalen WAV-Format auf dem zentralen Computer abgespeichert und bietet die Grundlage zur Generierung der Trainingsdaten.

MEMS-Mikrofon mit ESP32-Mikrocomputer

Die Aufnahme mit dem MEMS-Mikrofon erfolgt durch eine zentrale Positionierung des Mikrofons im Raum. Die Aufnahmequelle befindet sich im Schallradius der Geräuschquelle.

Das Audiosignal wird mit dem MEMS-Mikrofon aufgenommen und ohne CPU-Unterstützung des Mikrocomputers in den SPIFFS-Speicher des ESP32 geschrieben. Mit Hilfe des FS-Browsers wird der Datensatz auf den Laptop geladen und lokal gespeichert. Versuche im Rahmen dieser Arbeit haben ergeben, dass es bei Abtastraten von über 24 kHz in Kombination mit Aufnahmedauern von mehr als 10 Sekunden zu einem Datenverlust kommen kann. Der Autor hat entschieden, für die Aufnahme der Audiorohdaten die maximale Abtastrate auf 24 kHz und die Aufnahmedauer auf 10 Sekunden zu begrenzen. Es werden pro Audioklasse sechs Aufnahmen gespeichert. Die auf dem Laptop gespeicherten sechs Aufnahmen werden mit der Software Audacity manuell zu einer Sequenz mit einer Gesamtlänge von einer Minute erstellt und abgespeichert.

Tablet mit integriertem Mikrofon

Das Audiosignal wird mit dem integrierten Mikrofon des Tablets aufgenommen. Es besteht bei dem Tablet keine Möglichkeit, die Abtastrate zu verändern. Die installierte Android App “Hi-Q MP3 Voice Recorder” speichert die Aufnahme nach Beendigung automatisch auf dem Tablet als auch in dem definierten Cloudverzeichnis Google-Drive. Die Abtastrate beträgt unveränderbar 44,1 kHz.

Kondensatormikrofon mit digitalem Audiointerface und PC

Die Aufnahme mit dem Kondensatormikrofon gilt als Referenz hinsichtlich der Aufnahmequalität. Sie erfolgt parallel und zeitgleich zur Aufnahme mit dem Tablet, das in einem Abstand von weniger als 30 cm zum Kondensatormikrofon positioniert und in gleicher Richtung zur Schallquelle ausgerichtet ist (Versuchsaufbau siehe Abb. 3.10).

Abbildung 3.10: Versuchsaufbau Tablet, Kondensatormikrofon mit Audiointerface und Laptop

Die Aufnahme der Audiorohdaten erfolgt mit der Software Audacity. Die Aufnahmequellen Kondensatormikrofon als auch das integrierte Mikrofon im Tablet befinden sich im Schallradius der Geräuschquelle. Die Aufnahmen mit dem Kondensatormikrofon und mit dem Tablet werden somit unter den gleichen Bedingungen durchgeführt worden. Die Abtastrate beträgt ebenfalls 44,1 kHz.

Integriertes Laptop-Mikrofon

Als weitere Referenzaufnahmequelle wird im Kontext der Datenvorverarbeitung und Aufnahme der Audiorohdaten zusätzlich das integrierte Mikrofon des eingesetzten Laptops genutzt. Dies hat den Vorteil, die Aufnahmen mit dem Kondensatormikrofon in Bezug auf die Aufnahmequalität und dem eingestellten Signalpegel zu überwachen. Ein weiterer Vorteil ist die Möglichkeit, die Aufnahmequalität mit dem im Tablet integrierten Mikrofon zu vergleichen.

Die Aufnahme der Audiorohdaten erfolgt wie bei der Aufnahme mit dem Kondensatormikrofon mit Hilfe der Software Audacity. Der Laptop wird wiederum zentral im Raum positioniert. Die Aufnahmequelle befindet sich im Schallradius der Geräuschquelle. Die Abtastrate beträgt 44,1 kHz.

Abbildung 3.11 zeigt exemplarisch eine Aufnahme eines sich wiederholenden Hilferufs mit dem integrierten Mikrofon des Laptops und der Software Audacity. Es ist neben der Abtastrate von 44,1 kHz und einer Aufnahmedauer von einer Minute zu erkennen, dass sich der Hilferuf über die Aufnahmeperiode kontinuierlich ohne Pausen von mehr als einer Sekunde wiederholt. Hintergrund ist hier der Algorithmus zur Generierung der Trainingsdaten, der im nachfolgendem Abschnitt 3.4.2 näher erläutert wird.

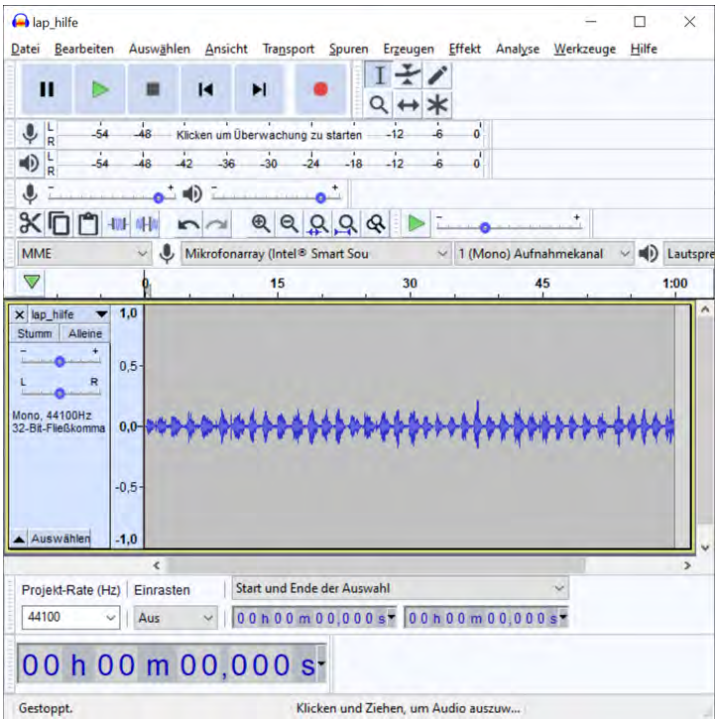


Abbildung 3.11: Aufnahme eines sich wiederholenden Hilferufs mit dem integrierten Laptop-Mikrofon und der Software Audacity
Quelle: Eigene Darstellung

Zwischenergebnis:

Das Ergebnis der Audiorohdatenaufnahme mit den oben beschriebenen, unterschiedlichen Aufnahmequellen ergibt ein Paket aus 80 WAV-Dateien mit 20 verschiedenen Audioklassen von vier unterschiedlichen Geräten und mit einer Länge von jeweils einer Minute.

3.4.2 Generierung der Trainingsdaten aus den Audiorohdaten

Aus dem Paket der 80 WAV-Dateien werden für jede Audioklasse Trainingsdaten generiert, mit denen die CNN-Modelle trainiert werden. Die Anzahl von Trainingsdaten für Spracherkennungsprogramme variiert sehr stark und kann pro Klasse einige tausend Daten beinhalten. So enthält der Datensatz von Google Speech Commands ca. 65.000 eine Sekunde lange Aussprachen von 30 verschiedenen kurzen Wörtern, die von tausenden verschiedenen Personen gesprochen werden [41]. Im Gegensatz dazu umfasst der ESC-50-Datensatz Umweltgeräusche in 50 Klassen und 40 Datensätzen pro Klasse mit einer Länge von 5 Sekunden [42].

Während der Datensatz von Google sich auf die Erkennung von Wörtern konzentriert, die von einer Vielzahl von Personen in unterschiedlicher Aussprache und Deutlichkeit gesprochen werden, hat der ESC-50-Datensatz den Anspruch, Trainingsdaten zur Verfügung zu stellen, um Umweltgeräusche zu unterscheiden. Dies ist aus Sicht des Autors aufgrund der Unterschiedlichkeit der Geräusche weniger anspruchsvoll und kommt der Anforderung einer Erkennung von Geräuschen und Sprache im Wohnumfeld älterer Menschen näher, insbesondere, da sich in diesem Umfeld die gesprochenen Worte auf wenige Personen begrenzen.

Um einen, auf das Wohnumfeld älterer Menschen abgestimmten Trainingsdatensatz zu verwenden, wird mit Hilfe eines im Rahmen dieser Arbeit erstellten Python-Programms ein eigener Datensatz aus im Wohnumfeld aufgenommenen Audiorohdaten generiert. Die Standardparametrierung des Programms verarbeitet bis zu 20 Audioklassen mit einer Länge von einer Minute und erstellt daraus 120 WAV-Dateien pro Audioklasse mit einer Länge von einer Sekunde und einer Abtastrate von 44,1 kHz. Andere Parametrierungen insbesondere die Länge der WAV-Dateien und die Abtastrate sind veränderbar und werden im Kapitel 4.1 im Rahmen der Studie verschiedener Methoden und Parameter näher betrachtet.

Die Ablaufschritte des Programms zur Generierung eines Testdatensatzes von einer Sekunde Länge sind in Abbildung 3.12 dargestellt. Mit der Methodik des Programms können von einer Audiorohdatenaufnahme von 60 Sekunden bei einer Abtastrate von 44,1 kHz über 2,6 Mio. WAV-Datensätze für eine Audioklasse erzeugt werden. Das Programm optimiert die Qualität zur Generierung der Dateien, indem im Bereich einer, durch eine Zufallszahl ausgewählten Stelle ein definierter Schwellwert des Pegels gesucht wird und die Sequenz ab dem Schwellwert abzüglich eines Zeitversatzes von 0,1 s abgespeichert wird. Als Ergebnis werden für alle Audioklassen 120 Datensätze mit einer Länge von einer Sekunde erstellt.

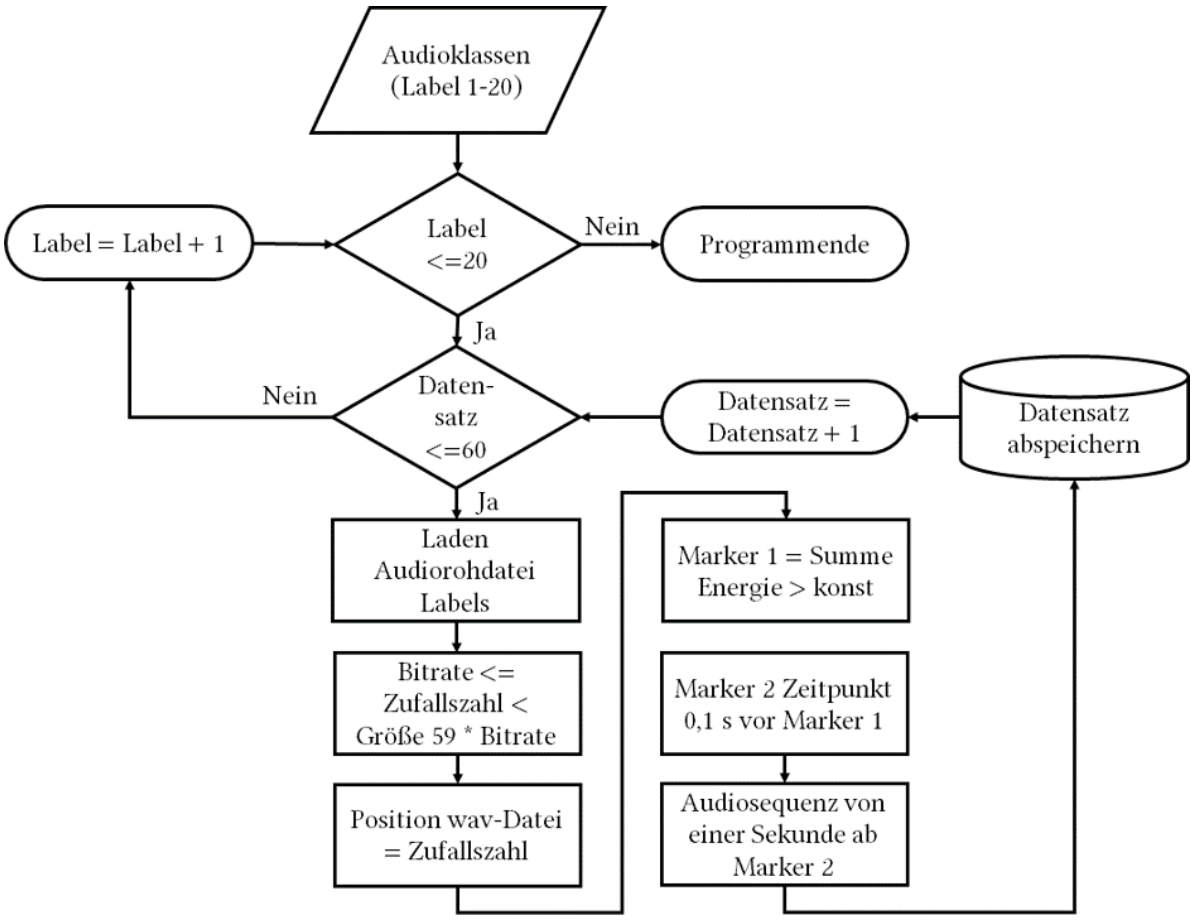


Abbildung 3.12: Flussdiagramm des Python-Programms zur Generierung von Trainingsdaten für alle Audioklassen (Label)
Quelle: Eigene Darstellung

3.4.3 Audioaufnahme im Wohnumfeld zur Geräusch- und Spracherkennung

Nachdem die vorbereitenden Teile “Aufnahme der Audiorohdaten” und “Generierung der Trainingsdaten aus den Audiorohdaten” abgeschlossen sind, erfolgt im Wohnumfeld älterer Menschen die kontinuierliche Erfassung der Geräusche in den festgelegten Räumen mittels Aufnahmequellen wie MEMS-Mikrofone oder einem integrierten Mikrofon im Tablet sowie die Weiterverarbeitung der Aufnahme in Form von Extraktion und Klassifizierung mit einem geeigneten CNN-Modell.

Die Aufnahme im Wohnumfeld erfolgt vergleichbar mit der Aufnahme der Audiorohdaten zur Generierung der Trainingsdaten mit dem Unterschied, dass die Aufnahmedauer auf einen Zeitraum von einer Sekunde begrenzt wird und danach die Audiodatei zur Erkennung über WLAN übertragen und auf dem zentralen Computer im Wohnumfeld abgespeichert wird. Bei

einer Installation von Mikrofonen in verschiedenen Räumen wird pro Raum eine Audioaufnahme erstellt. Die Aufnahme und das Abspeichern der Aufnahme erfolgen kontinuierlich. Für die Methodik zur Erkennung von Geräuschen und Sprache ist es im Rahmen dieser Arbeit unerheblich, ob die Datei mit der vorherigen Aufnahme überschrieben oder ein neuer Dateiname mit einem Zeitstempel vergeben wird.

Die Zeitdauer ist identisch mit den oben genannten Trainingsdaten und der Dauer des Datensatzes von Google Speech Commands. Im Kapitel 4.1.5 werden im Rahmen der Studie die Zeitdauer auf 0,5 Sekunden verkürzt und auf 2 Sekunden verdoppelt und die daraus resultierenden Auswirkungen ermittelt.

Die Aufnahmequalität entspricht hinsichtlich der Bittiefe den technischen Daten der Hardware. Die Abtastrate wird auf die höchst gewählte Rate der Trainingsdaten eingestellt. Dies sind bei der Konfiguration mit dem MEMS-Mikrofon 24 kHz und bei dem integrierten Mikrofon im Tablet 44,1 kHz. Im Kapitel 4.2 werden die Ergebnisse der Parametrierungen mit Versuchsaufbauten mit dem Kondensatormikrofon verglichen. Weiterhin werden in diesem Kontext die Auswirkungen einer Parametrierung von geringeren Abtastraten bis zu 8 kHz untersucht.

3.4.4 Extraktion und Aufbereitung der Audiosignale

Um die kontinuierlich im Wohnumfeld aufgenommenen Audiodateien mit einem CNN-Modell zu klassifizieren, bieten sich verschiedene, im Kapitel 2.8 beschriebene Methoden an, die Informationen der Aufnahme aus der Audiodatei zu extrahieren.

Die im verlustlosen WAV-Format als Audiodatei abgespeicherte Aufnahme besteht aus einer Folge von Amplituden des aufgenommenen Pegels über eine Zeitachse von einer Sekunde. Die grafische Darstellung entspricht einem Amplitude-Zeit-Diagramm.

Amplitude-Zeit-Diagramm, Frequenzspektrum und Spektrogramme

In den folgenden Diagrammen “Amplitude-Zeit” (Abb. 3.13) “Frequenzspektrum” (Abb. 3.14) und “Spektrogramm” (Abb. 3.16) wird das Audiosignal “Festnetztelefon” mit einer Abtastrate von 44,1 kHz und einer Dauer von einer Sekunde dargestellt.

Die Frequenzen und ihre Obertöne sind im Spektrum deutlich zu erkennen und von anderen Spektren wie das Spektrum einer Sprache - zum Beispiel einem Hilferuf (Abb. 3.15) - optisch unterscheidbar. In diesem Fall ist zu erwarten, dass CNN-Modelle, die eine Bildauswertung des Frequenzspektrums durchführen, eine Unterscheidung mit hoher Zuverlässigkeit durchführen können, ohne den zeitlichen Verlauf des Signals über die Aufnahmedauer von einer Sekunde zu berücksichtigen.

In diesem Zusammenhang stellt sich die Frage, inwieweit CNN-Modelle bereits Amplitude-Zeit-Diagramme zur Unterscheidung von Geräuschen und Sprache verwenden können. In diesem Fall sind keine Information über die Frequenzanteile verfügbar. Weiterhin kommt es entlang der Zeitachse aufgrund des unbestimmten Aufnahmezeitpunktes zu einer Verschiebung. Die Ergebnisse dieser Überlegung werden ebenfalls in Kapitel 4 beschrieben.

Das Spektrogramm bietet gegenüber der bildlichen Darstellung des Frequenzspektrums den Vorteil der zeitlichen Abfolge der Signale. In diesem Fall können unterschiedliche Ton- und Melodiefolgen unterschieden werden, obwohl die gleichen Einzeltöne oder der gleiche Klang in der Aufnahme auftaucht. Bei melodischen Signalen wie das Audiosignal eines Telefons sind die polyphonen Obertöne im Frequenzspektrum (Abb. 3.14) als auch im Spektrogramm (Abb. 3.16) erkennbar.

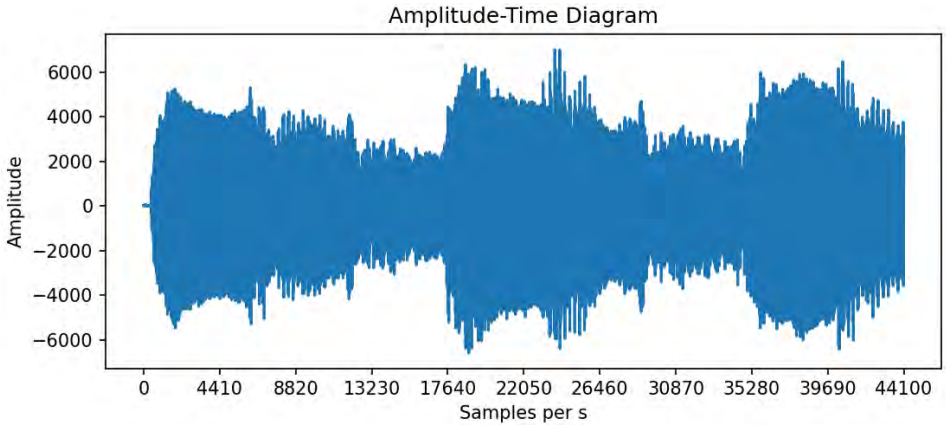


Abbildung 3.13: Amplituden-Zeit Diagramm: Festnetztelefon 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

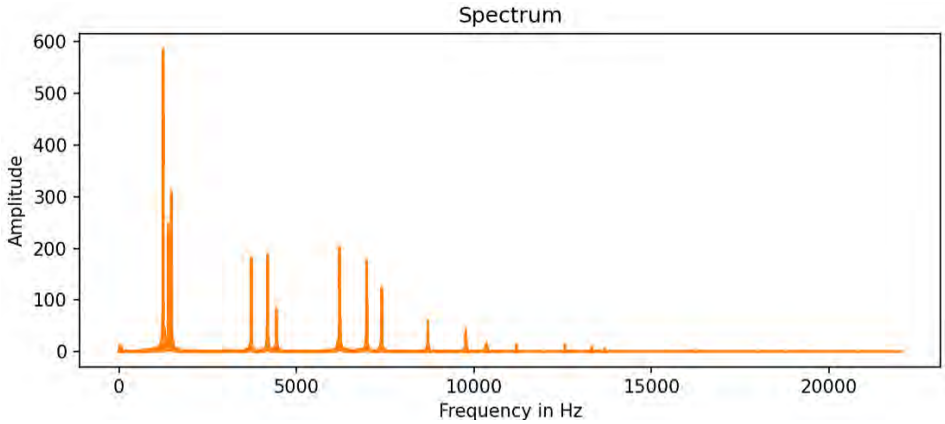


Abbildung 3.14: Spektrum: Festnetztelefon 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

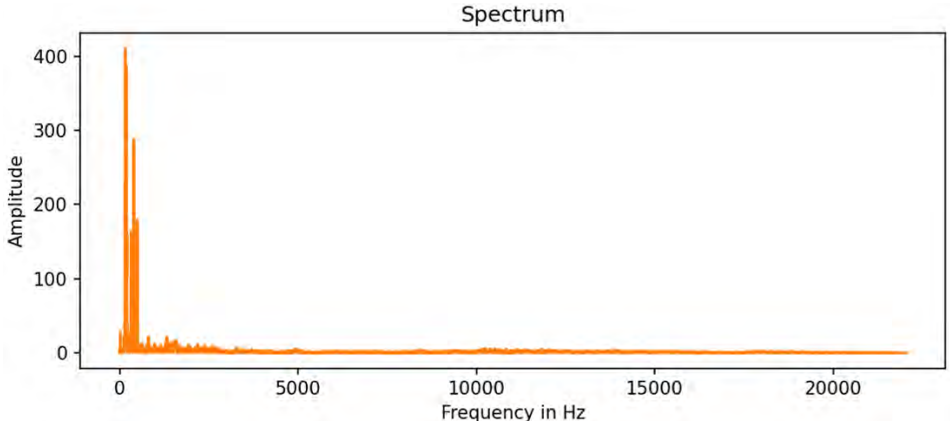


Abbildung 3.15: Spektrum: Hilferuf 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

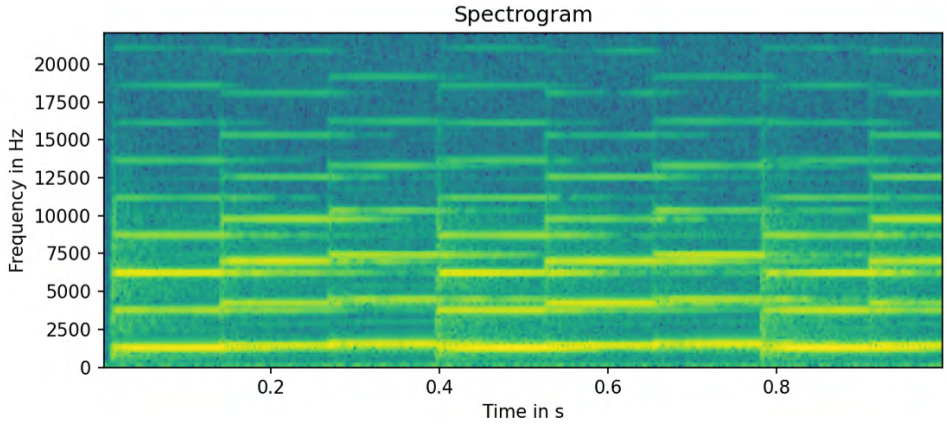


Abbildung 3.16: Spektrogramm: Festnetztelefon 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

MFCC

Die nachfolgende Darstellung der Diagramme “Mel Spektrum” (Abb. 3.17) und “MFCC mit delta und delta²” (Abb. 3.18) für das Audiosignal Festnetztelefon mit einer Abtastrate von 44,1 kHz und einer Dauer von einer Sekunde zeigt die grafischen Ergebnisse der Audiodatenextraktion nach Mel. Die Darstellung ist abhängig vom gewählten Parametersatz und bietet somit die Basis einer Parameteroptimierung vor einer Auswertung durch ein CNN-Modell. Wie im Kapitel 2.8 beschrieben, ist zur Spracherkennung eine Parametrierung von dreizehn Koeffizienten und zur Geräuscherkennung von fünf Koeffizienten verbreitet. Ein wesentlicher Vorteil der Mel-Daten ist die Reduzierung der Datenmenge auf die wesentlichen Eigenschaften des Audiosignals.

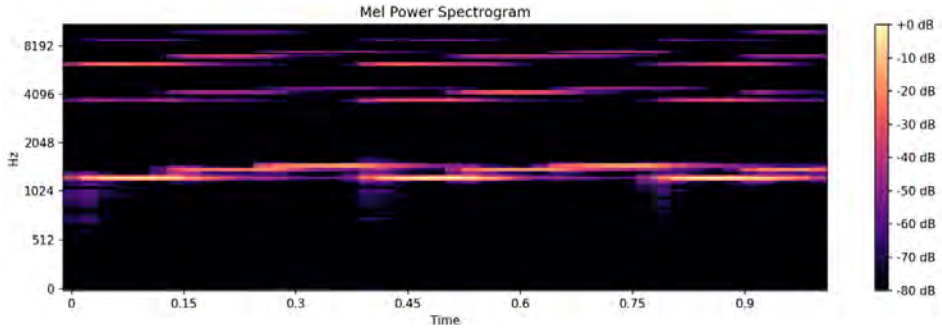


Abbildung 3.17: Mel Spektrum: Festnetztelefon 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

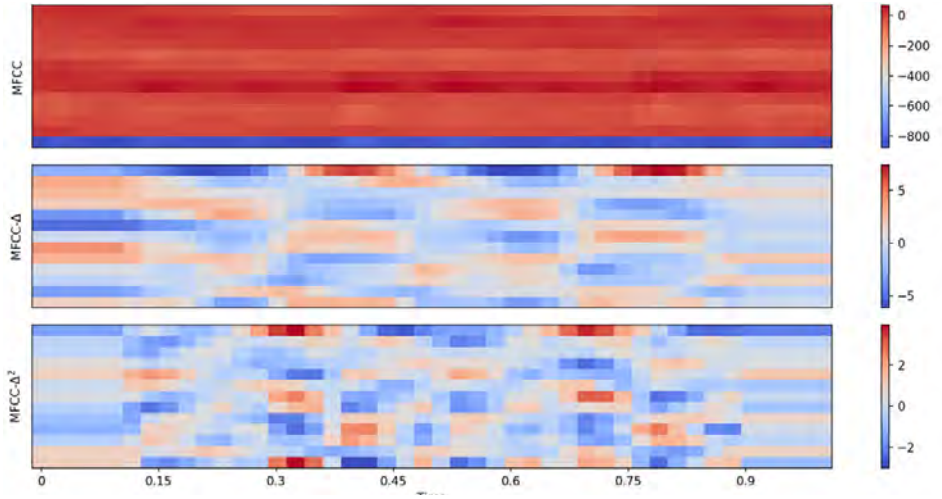


Abbildung 3.18: MFCC delta und delta²: Festnetztelefon 44,1 kHz Abtastrate
Quelle: Eigene Darstellung

Metadaten

Metadaten können als zusätzliche Information die Unterscheidung zwischen Normal- und Gefahrensituation unterstützen. Dies umfasst Daten wie Tageszeit und Dauer des Signals als auch Signal spezifische Eigenschaften wie die im Kapitel 2.3 beschriebenen Frequenzbereich basierenden Audioeigenschaften BER, SC, Korrelation und ZCR.

Extraktionen als Vorbereitung zur Verarbeitung in CNN-Modellen werden im Rahmen dieser Arbeit als vorgelagerter Teil der Software des CNN-Modells erstellt und in Python programmiert. Wie im Kapitel 2.2.2 beschrieben, stehen dem Anwender dafür umfangreiche Bibliotheken zur Verfügung.

3.4.5 Gegenüberstellung verwendeter CNN-Modelle

In dieser Arbeit werden drei CNN-Modelle zur Geräusch- und Spracherkennung gegenübergestellt. Sie unterscheiden sich in der Datenvorverarbeitung und benutzen unterschiedliche Methoden und Algorithmen. Alle mit den Modellen verwendeten Ursprungstrainingsdaten werden durch die im Kontext dieser Arbeit erstellen Trainingsdaten ersetzt.

Nachfolgend werden folgende CNN-Modelle beschrieben:

1. Modell: Simple Audio Recognition: Recognizing Keywords [43]
2. Modell: Speech Command Classification with Torchaudio [44]
3. Modell: ESC-Dataset for Environmental Sound Classification [45]

Zur Erkennung von Geräuschen und von Sprache sind weitere vielfältige CNN-Modelle im Internet als Grundgerüst verfügbar. Der Autor hat sich auf die oben genannten Modelle beschränkt, da sie sowohl Geräusch- als auch Spracherkennung berücksichtigen und unterschiedliche Methoden in der Datenvorbereitung und in den Netzwerken verwenden.

1. Modell: Simple Audio Recognition: Recognizing Keywords

Das erste Modell in dieser Arbeit basiert auf einem TensorFlow-Tutorial zur Erkennung von Schlüsselwörtern. In dem Tutorial wird die Verarbeitung von Audiodateien im WAV-Format durchgeführt. Das Modell verarbeitet Teile des Speech-Commands-Datensatz von Warden [46]. Die Klassifizierung erfolgt mit TensorFlow-Keras. Es werden insgesamt 8 Klassen für das Tutorial verwendet. Die Audiodatensätze haben eine feste Bitrate von 16 kHz und eine Länge von maximal einer Sekunde. Das Modell verwendet insgesamt 1.000 Datensätze pro Klasse und teilt diese im Modell in 80% Trainings- und 20% Validierungsdaten auf. Im Vergleich zu den beiden anderen genannten Modellen verwendet dieses Modell in seiner Ursprungsversion eine kleinere Anzahl an Klassen. Die Anzahl der Datensätze pro Klasse ist deutlich geringer als im zweiten Modell.

Zur bildlichen Darstellung werden Amplitude-Zeit-Diagramme sowie Frequenzspektren und Spektrogramme verwendet. Es werden keine Mel-Spektrogramme und Extraktionsverfahren wie MFCC angewandt. Als CNN-Modell wird zur Erkennung der Spektrogramme ein TensorFlow-Keras-Sequential-Modell eingesetzt. Der Programmcode des Modells ergibt sich wie folgt (3.1):

Codeblock 3.1: 1. Modell mit TensorFlow-Keras-Sequential-Modell

```
1 input_shape = example_spectrograms.shape[1:]
2 print('Input␣shape:', input_shape)
3 num_labels = len(label_names)
```

```
4
5  # Instantiate the `tf.keras.layers.Normalization` layer.
6  norm_layer = layers.Normalization()
7  # Fit the state of the layer to the spectrograms
8  # with `Normalization.adapt`.
9  norm_layer.adapt(data=train_spectrogram_ds.map(map_func=lambda spec, label: spec))
10
11 model = models.Sequential([
12     layers.Input(shape=input_shape),
13     # Downsample the input.
14     layers.Resizing(32, 32),
15     # Normalize.
16     norm_layer,
17     layers.Conv2D(32, 3, activation='relu'),
18     layers.Conv2D(64, 3, activation='relu'),
19     layers.MaxPooling2D(),
20     layers.Dropout(0.25),
21     layers.Flatten(),
22     layers.Dense(128, activation='relu'),
23     layers.Dropout(0.5),
24     layers.Dense(num_labels),
25 ])
26
27 model.summary()
```

Wie im Programmcode ersichtlich, werden zwei Faltungsoperationen durchgeführt. Weitere Methoden sind die Normalisierung zur schnelleren Konvergenz der Gewichtungen und das Dropout, um Overfitting zu vermeiden. Im Tutorial wird ebenfalls der Adam-Optimierer eingesetzt. Als Parametrierung kann die Anzahl der Epochen verändert werden. Als grafische Auswertung werden die Genauigkeit (Accuracy), die Loss-Epochen-Kurve und eine Confusion-Matrix ausgegeben.

2. Modell: Speech Command Classification with Torchaudio

Das zweite Modell erkennt 35 Sprachbefehle auf Basis der Trainingsdaten des bereits oben genannten Google Speech Commands Datensatzes. Der importierte Datensatz von Google stellt Audiodateien mit einer Abtastrate von 16 kHz und einer Länge von einer Sekunde zur Verfügung. Da das Frequenzspektrum der menschlichen Sprache im Wesentlichen unter 8 kHz liegt, ist diese Abtastrate zur Erkennung von Sprache ausreichend. Für Geräusche insbesondere mit hochfrequentem Anteil kann das Modell eingeschränkt sein. Die Auswirkungen werden im Kapitel 4.2.1 im Rahmen der experimentellen Ergebnisse näher beschrieben.

Der Google Speech Commands Datensatz verfügt neben dem Audiosignal im WAV-Format über weitere Metadaten wie Sprecher-ID, Abtastrate und Aussprache (utterance). In dem Originalcode wird die Abtastrate zur schnelleren Datenverarbeitung auf 8 kHz reduziert. Es handelt sich wie bei den Aufnahmen der Originaltrainingsdaten um einkanalige Aufnahmen (Mono).

Das Netzwerk ist ein M5-Modell, das 2016 von Wei Dai et al. als "VERY DEEP CNN-Network" vorgestellt wird [47]. Das Modell wurde von Wei Dai et al. mit einem Datensatz von zehn Umweltgeräuschen aus 8.732 Audioaufnahmen und einer Audiolänge von bis zu 4 Sekunden getestet. Die Abtastrate beträgt aus Gründen der Rechenleistung 8 kHz. Weitere Details wie die Anzahl der Epochen und die Art des Optimierers sind in dem Bericht von Wei Dai et al. beschrieben. Durch die Anwendung des M5-Algorithmus ist für dieses Modell auch die Geräuscherkennung möglich, wobei die Auswirkungen der reduzierten Abtastrate auf die Genauigkeit der Erkennung von Audiosignalen im Wohnumfeld älterer Menschen betrachtet werden muss.

Nach dem Training des Modells mit dem Trainingssatz der Audiodaten, werden die Anzahl der Epochen festgelegt und die Genauigkeit (Accuracy) während des Trainings ausgegeben.

Das M5-Modell verwendet den folgenden Algorithmus:

Codeblock 3.2: M5-Algorithmus vom 2. Modell

```
class M5(nn.Module):
    def __init__(self, n_input=1, n_output=35, stride=16, n_channel=32):
        super().__init__()
        self.conv1 = nn.Conv1d(n_input, n_channel, kernel_size=80, stride=stride)
        self.bn1 = nn.BatchNorm1d(n_channel)
        self.pool1 = nn.MaxPool1d(4)
        self.conv2 = nn.Conv1d(n_channel, n_channel, kernel_size=3)
        self.bn2 = nn.BatchNorm1d(n_channel)
        self.pool2 = nn.MaxPool1d(4)
        self.conv3 = nn.Conv1d(n_channel, 2 * n_channel, kernel_size=3)
        self.bn3 = nn.BatchNorm1d(2 * n_channel)
        self.pool3 = nn.MaxPool1d(4)
        self.conv4 = nn.Conv1d(2 * n_channel, 2 * n_channel, kernel_size=3)
        self.bn4 = nn.BatchNorm1d(2 * n_channel)
        self.pool4 = nn.MaxPool1d(4)
        self.fc1 = nn.Linear(2 * n_channel, n_output)

    def forward(self, x):
        x = self.conv1(x)
        x = F.relu(self.bn1(x))
```

```
        x = self.pool1(x)
        x = self.conv2(x)
        x = F.relu(self.bn2(x))
        x = self.pool2(x)
        x = self.conv3(x)
        x = F.relu(self.bn3(x))
        x = self.pool3(x)
        x = self.conv4(x)
        x = F.relu(self.bn4(x))
        x = self.pool4(x)
        x = F.avg_pool1d(x, x.shape[-1])
        x = x.permute(0, 2, 1)
        x = self.fc1(x)
        return F.log_softmax(x, dim=2)

model = M5(n_input=transformed.shape[0], n_output=len(labels))
model.to(device)
print(model)
```

Wie im oben genannten Codeblock 3.2 zu sehen ist, werden unter anderem 4 Faltungsvorgänge durchgeführt. Die Aktivierungsfunktion ist ReLu.

Im Kapitel 4.3 werden die Ergebnisse der Studie mit dem Modell vorgestellt.

3. Modell: ESC-Dataset for Environmental Sound Classification

Das dritte Modell wurde bereits 2015 von Karol J. Piczak entwickelt und benutzt den ESC-50-Datensatz mit einer Sammlung von 2.000 Aufnahmen aus der Umwelt mit 50 Klassen und 40 Audioaufnahmen pro Klasse. Die Aufnahmen wurden im Rahmen des Freesound.org-Projekts erstellt. Die Aufnahmen mit einer Abtastrate von 44,1 kHz sind 5 Sekunden lang [48]. Neben dem ESC-50-Datensatz existiert ein weiterer reduzierter ESC-10-Datensatz als Teil des ESC-50-Datensatzes.

Piczak bewertet in seinem Bericht zum Modell [42] das Potential von Faltungs-Neuronalen Netzen bei der Klassifizierung kurzer Audioclips von Umweltgeräuschen. Das Modell verwendet zwei Faltungsoperationen mit Max-Pooling und wird auf segmentierte Spektrogramme mit Mel-Deltas trainiert. Das Modell übertrifft laut seinem Bericht Implementierungen, die auf MFCC basieren.

In dem dritten Modell werden nach der Programmierung der Grundeinstellungen und dem Import von Bibliotheken die 50 Klassen (Abb. 3.19) aufgelistet.

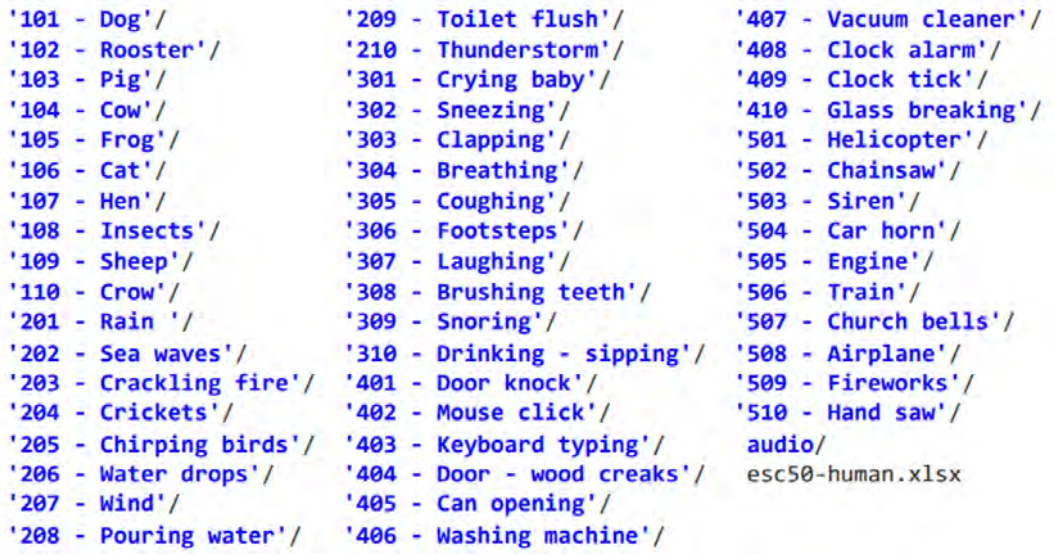


Abbildung 3.19: ESC-50 Klassen aus dem Modell von Karol Piczak
Quelle: eigene Darstellung

Dabei lassen sich die Klassen den folgenden Untergruppen zuordnen:

1. Animals (101-110)
2. Natural soundscapes & water sounds (201-210)
3. Human (non-speech) sounds (301-310)
4. Interior/Domestic Sounds (401-410)
5. Exterior/Urban Noises (501-510)

Aus Sicht des Autors dieser Arbeit besteht damit eine Ähnlichkeit zu den gewählten Klassen im Wohnumfeld älterer Menschen wie die Untergruppe Interior/Domestic Sounds (401-410). Im Gegensatz zum ersten Modell werden keine Datensätze mit Sprache verwendet. Da Piczak das Modell bereits 2015 entwickelt hat, mussten vom Autor verschiedene programmtechnische Veränderungen vorgenommen werden. Weitere Informationen zur Lauffähigkeit des Codes im Rahmen der Studie werden im Kapitel 4.3 beschrieben.

Nach dem Import der Datensätze werden zur Illustration grafisch Amplitude-Zeit-Diagramme und Frequenzspektren verschiedener Signale dargestellt, um die unterschiedlichen Charakteristika der Geräusche zu verbildlichen. Piczak verwendet in seinem Modell unter anderem MFCC zur Extraktion der Aufnahmen.

Vor der Erkennung der Geräusche beschreibt Piczak die Genauigkeit einer Erkennung von Geräuschen durch verschiedene Menschen. In seiner Untersuchung ermittelt er bei der Erkennung von Geräuschen des ESC-10-Datensatzes eine durchschnittliche Genauigkeit von 95,5% und beim ESC-50-Datensatz von nur noch 81,3%.

Als CNN-Modell verwendet Piczak die freie Software-Bibliothek Scikit-Learn [49], mit der Klassifizierer wie k-Nearest Neighbors, Random Forest und Support Vector Machine (SVM) verwendet werden. Mit den Datensätzen ESC-10 und ESC-50 stellt Piczak die Ergebnisse der Erkennung in Form einer Matrix dar. Der Autor dieser Arbeit verwendet im Kontext der Studie die Basis des Modells von Piczak und die Datensätze aus den, im Abschnitt 3.4.2 beschriebenen Trainingsdaten, die aus den Audiorohaufnahmen generiert werden.

Zusammenfassung der Gegenüberstellung der drei vorgestellten Modelle

Die Auswahl der drei oben beschriebenen Modelle bietet aus Sicht des Autors eine Grundlage für Modelle zur Geräusch- und Spracherkennung im Wohnumfeld. Zur Validierung dieser These werden die erstellten Klassen und Trainingsdaten für die Anwendung der drei Modelle hinsichtlich Datenstruktur, Abtastrate und Anzahl der Klassen vorbereitet und die Originalcodes der drei Modelle durch den Autor so modifiziert, dass die generierten Trainingsdaten verwendet und Vorhersagegenauigkeiten zwischen den Modellen verglichen werden können. Die Ergebnisse dieser Untersuchungen werden im folgenden Kapitel 4 vorgestellt.

3.5 Störgrößeneinfluss während der Audiosignalaufnahme

Bei der Audiosignalaufnahme können Störgrößen auftreten, die die Vorhersagegenauigkeit einer Erkennung von Geräuschen und Sprache negativ beeinflussen. Im Wohnbereich können Geräusche und Lärm von außen wie innerhalb des Wohnbereichs einen Pegel erreichen, der eine Erkennung und Klassifizierung des Signals erschwert. Aus verschiedenen Studien geht hervor, dass mit steigenden Hintergrundgeräuschen die Genauigkeit einer korrekten Erkennung von Audiosignalen abnimmt [50] [20].

Ein weiterer wesentlicher Einfluss sind Umgebungsgeräusche und Audiosignale verschiedener Quellen zum gleichen Zeitpunkt. Es kommt zu einer Addition der Signale, so dass eine Aussage, um welche Signale es sich handelt, nur durch aufwendige Algorithmen berechnet werden kann. Aufgrund der Komplexität der Algorithmen wird der Einfluss dieser Störgrößen nicht vertiefend betrachtet, allerdings auf das MFCC-Extraktionsverfahren in Kapitel 2.8 auf Seite 15 hingewiesen, wo festgestellt wird, dass MFCC schlechtere Ergebnisse von Merkmalen bei Hintergrundgeräuschen liefert [14].

Ein weiterer Störgrößeneinfluss sind Geräusche, die einer definierten Klasse nicht zugeordnet werden können. Im Kapitel 3.3 wurden insgesamt 20 verschiedene Klassen in unterschiedlichen Wohnbereichen festgelegt. Sollten andere Geräuscharten regelmäßig im Wohnumfeld

der betroffenen Person zu erwarten sein, bietet es sich an, die Anzahl der Klassen entsprechend zu erweitern.

Am Anfang des Kapitels wurde ausführlich der Einfluss der Raumakustik diskutiert. Raumakustische Besonderheiten wie besondere Fußbodenbeläge, Radio- und Fernsehgeräte oder andere installierte Geräte mit Schallemissionen können die Genauigkeit der Erkennung von Geräuschen negativ beeinflussen. Durch die Festlegung des Autors, an Stelle von vorhandenen Trainingsdatensätzen die Generierung von eigenen Datensätzen durch Aufnahme von Audiorohdaten durchzuführen, ist dieser störende Einfluss als gering einzuschätzen.

Nach Abschluss der Konzeptionierung und Implementierung werden im nächsten Kapitel die Ergebnisse der Studie wie Auswertung der Audioaufnahmen und Auswahl und Parametrierung von CNN-Modellen vorgestellt und diskutiert.

4 Studie verschiedener Methoden zur Geräusch- und Spracherkennung sowie deren Bewertung

Dieses Kapitel beschreibt ausgewählte Methoden und unterschiedliche Parametrierungen der Audiosignalverarbeitung zur Erkennung und Unterscheidung von Geräuschen und Sprache. Dies umfasst sowohl die Auswahl der Aufnahmequelle als auch Untersuchungen im Bereich der digitalen Weiterverarbeitung von Audiosignalen und deren Extraktion. Anschließend erfolgt die Klassifizierung der Signale mit verschiedenen CNN-Modellen und eine Bewertung der Ergebnisse.

Aufgrund der Vielschichtigkeit der im Kapitel 3.2.2 vorgestellten Hardwarekomponenten und der Softwareparameter wird der folgende Ablauf der Studie festgelegt:

1. Grundkonfigurationen auf Basis verschiedener Parameter
   - Kondensatormikrofon als Referenzaufnahmequelle
   - Testaufnahmen zur Überprüfung der Vorhersagegenauigkeit
   - Parametrierung und CNN-Modell
   - Abtastraten: 44,1 kHz / 16 kHz / 8 kHz
   - Signalaufnahmedauer: 1 s / 2 s / 0,5 s
   - Anzahl Klassen: 10 / 20
   - Anzahl WAV-Dateien pro Klasse: 60 / 120
2. Variation der Aufnahmequellen
   - MEMS-Miniaturmikrofon und ESP32-Mikrocomputer
   - Tablet mit integriertem Mikrofon
3. CNN-Modelle und deren Parametrierung
   - Simple Audio Recognition
   - Torch Audio Speech Command Classification
   - Environmental Sound Classification

4.1 Grundkonfigurationen auf Basis verschiedener Parameter

Im ersten Schritt wird eine Grundkonfiguration als Referenz festgelegt. Die Referenz vergleicht die Ergebnisse dieser Audiosignalerkennung mit Ergebnissen alternativer Hardwarekomponenten, veränderlicher Parameter und unterschiedlicher CNN-Modelle.

4.1.1 Kondensatormikrofon als Referenzaufnahmequelle

Als Aufnahmequelle wird das Kondensatormikrofon mit dem zugehörigen Audiointerface als Referenz gewählt, da es aufgrund der technischen Daten und Ausführung die beste Aufnahmequalität aller eingesetzten Quellen liefert.

4.1.2 Testaufnahmen zur Überprüfung der Vorhersagegenauigkeit

Im Rahmen dieser Arbeit wurde für jede Klasse eine Testaufnahme erstellt, um die Vorhersagegenauigkeit dieser Aufnahme zu einer Klasse zu prüfen und die Ergebnisse grafisch darzustellen.

Die Testaufnahmen haben die gleiche Aufnahmedauer wie die Trainingsdaten und dienen als Referenzmuster zur Überprüfung der Vorhersage. Die Audiosignale wurden vollständig und nicht fragmentiert aufgenommen. Der abgebildete Amplituden-Zeit-Verlauf für die Testaufnahme des Wortes “Hallo” zeigt die Vollständigkeit des Signals (Abb. 4.1). Das Signal startet nach 0,1 Sekunden und dauert in etwa 0,5 Sekunden. Andere erstellte Testaufnahmen weisen ähnliche Eigenschaften auf und sind damit für eine Überprüfung der Vorhersagegenauigkeit repräsentativ.

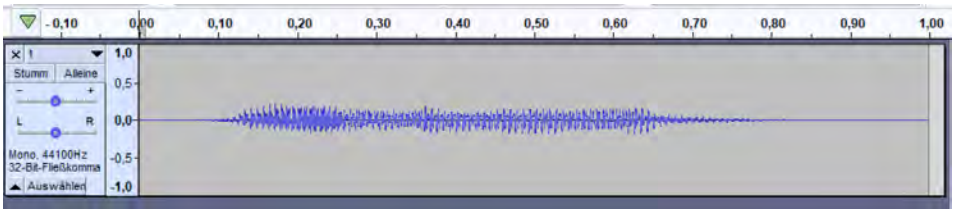


Abbildung 4.1: Beispiel Testaufnahme “Hallo”
Quelle: Eigene Darstellung

4.1.3 Parametrierung und CNN-Modell

Die Parameter zur Audiosignalverarbeitung werden zum Start der Studie auf Basis der Grundlagen aus Kapitel 2.1.2 und der Konzeptionierung aus Kapitel 3.1 wie folgt festgelegt:

- Abtastrate: 44,1 kHz
- Signaldauer: 1 Sekunde
- Anzahl Klassen: 10
- Anzahl Trainingsdaten pro Klasse: 120

Als erstes Modell wird in der Studie das im Kapitel 3.4.5 auf Seite 55 vorgestellte CNN-Modell "Simple Audio Recognition" eingesetzt. Nach Bestimmung der optimalen Signalverarbeitungsparameter werden die anderen vorgestellten CNN-Modelle mit diesem Modell verglichen.

"Simple Audio Recognition" ermittelt Kennzahlen für den Loss-Faktor und die Genauigkeit (Accuracy) der Erkennung. Weiterhin erstellt das Modell eine Confusion-Matrix "Vorhersage pro Klasse". Die Grafiken "Loss-Accuracy" (4.2) und "Confusion-Matrix" (4.3) zeigen die Ergebnisse der Modellberechnung.

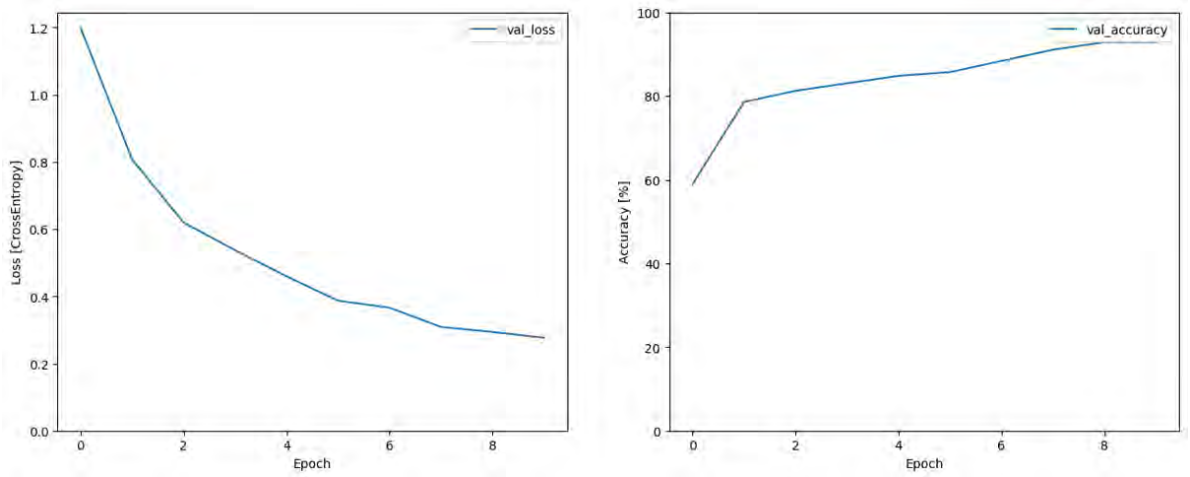


Abbildung 4.2: Kondensatormikrofon: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Wie der Grafik "Loss-Accuracy" zu entnehmen ist, reduziert sich der Loss kontinuierlich und die Accuracy steigt nach 10 Epochen auf einen Wert von 95%.

Aus der Confusion-Matrix lässt sich erkennen, dass die Vorhersage für die Klasse "Gespräch" teilweise das Wort "Hallo" liefert. Weitere fehlerhafte Vorhersagen treten für die Klassen "Festnetz", "Mobil" und "Schritte" ein.

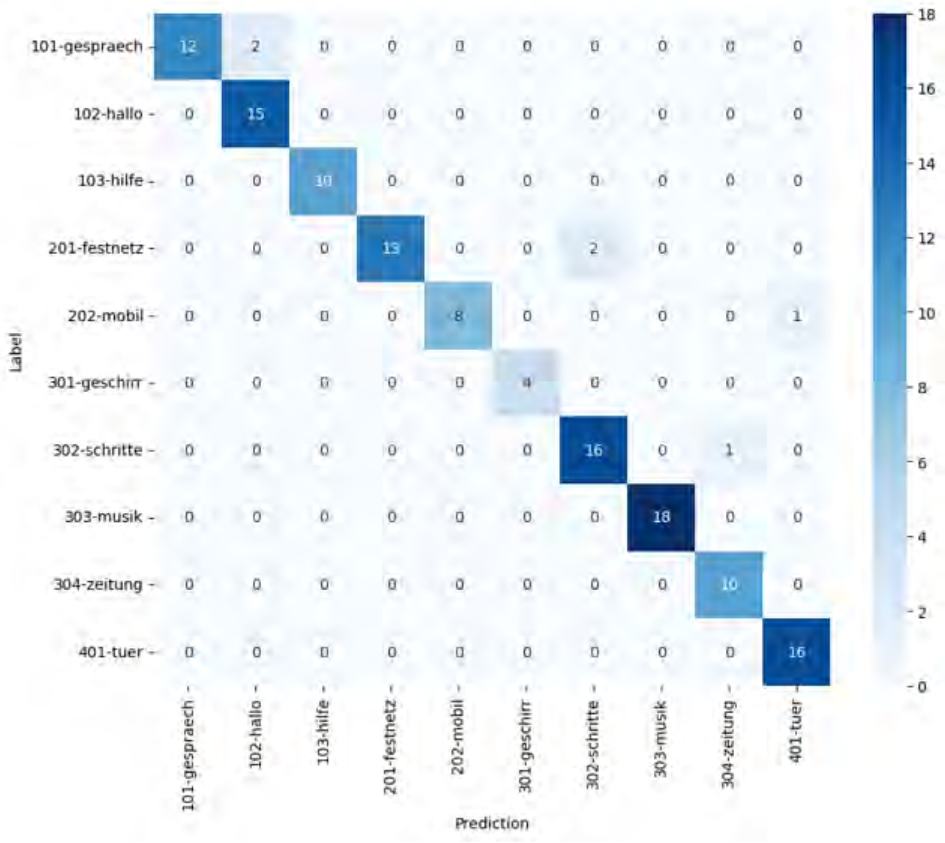


Abbildung 4.3: Kondensatormikrofon: Confusion-Matrix über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Eine Programmerweiterung des Autors speist die oben genannten Testaufnahmen sequentiell als Eingabe in das CNN-Model ein und gibt grafisch die abgebildete “Verteilung Vorhersage” für jede Klasse aus. Wie in der Abbildung 4.4 ersichtlich, sind die Vorhersagegenauigkeiten der Erkennung bis auf die Klassen “Schritte” und “Tür” bei annähernd 100%. Die geringere Vorhersagegenauigkeit der Klassen “Schritte” und “Tür” erklärt sich durch die Herausforderung der Signalerfassung dieser Geräusche im Vergleich zu einfacher zu identifizierenden Geräuschen wie “Festnetztelefon” oder “Musik”.

Zusammenfassung der Grundkonfigurationen

Die gewählten Parameter für die Abtastrate von 44,1 kHz und die Anzahl von 120 Dateien pro Klasse sind eine gute Referenz für Vergleichsmessungen. Die Anzahl der Klassen wurde auf 10 begrenzt, da, wie im Kapitel 3.3 bereits beschrieben, im Wohnumfeld in einem Raum 8-13 verschiedene Klassen erkannt werden sollen. Als CNN-Modell wird für die weiteren

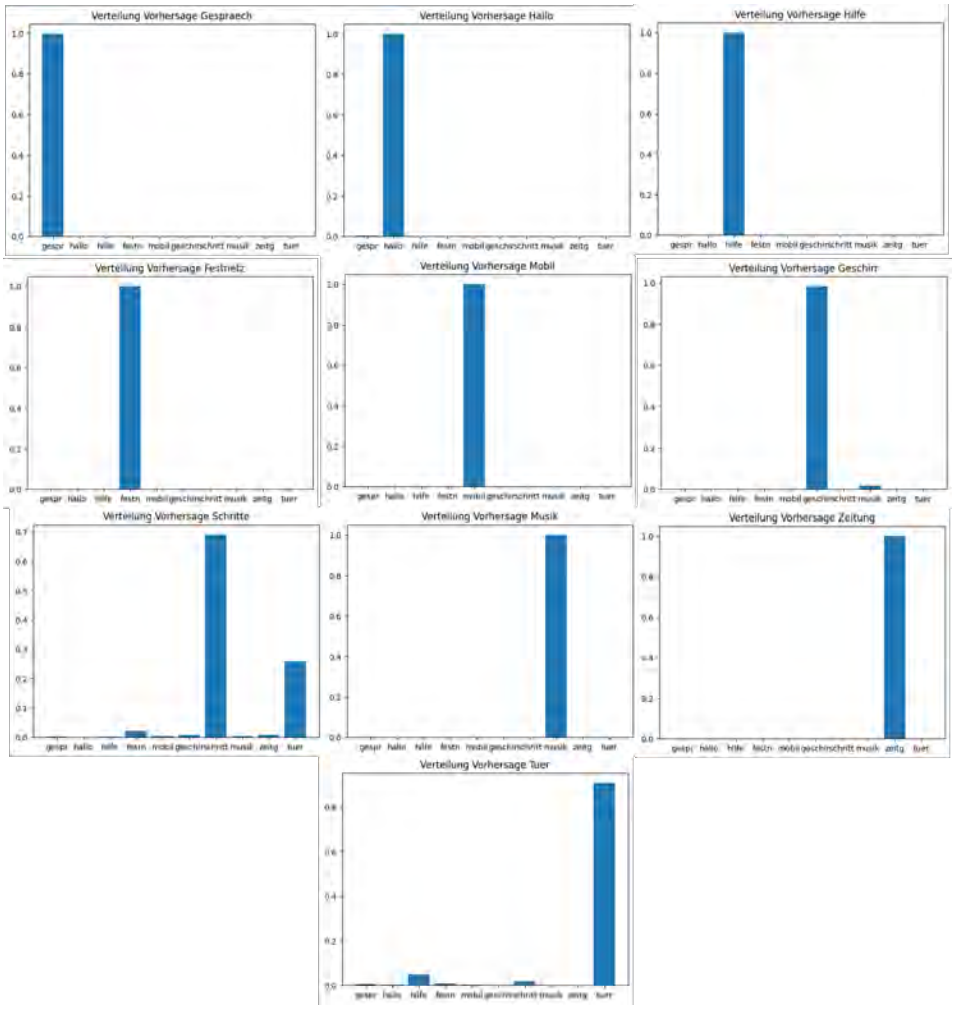


Abbildung 4.4: Kondensatormikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Versuche zur Ermittlung geeigneter Signalverarbeitungsparameter “Simple Audio Recognition” eingesetzt. Die aus dem Modell errechneten Kennzahlen “Loss und Accuracy” und “Verteilung Vorhersage” sind dabei in dieser Studie das Kriterium für die Güte der Geräusch- und Spracherkennung.

4.1.4 Abtastraten

Wie in den vorherigen Kapiteln beschrieben, bewirkt eine zu geringe Abtastrate einen Informationsverlust des Signals der Schallquelle. Im Folgenden werden die reduzierten Abtastraten von 16 kHz und 8 kHz der Grundkonfiguration von 44,1 kHz gegenübergestellt. Alle an-

deren Parameter als auch die Testaufnahmen bleiben konstant. Abbildung 4.5 zeigt “Loss und Accuracy” bei einer Abtastrate von 16 kHz. Die Werte sind mit der Abtastrate von 44,1 kHz vergleichbar und zeigen keine Verschlechterung der Erkennung auf Grundlage dieser Kennzahlen. Die Accuracy ist vergleichbar mit der Abtastrate von 44,1 kHz bei 95%.

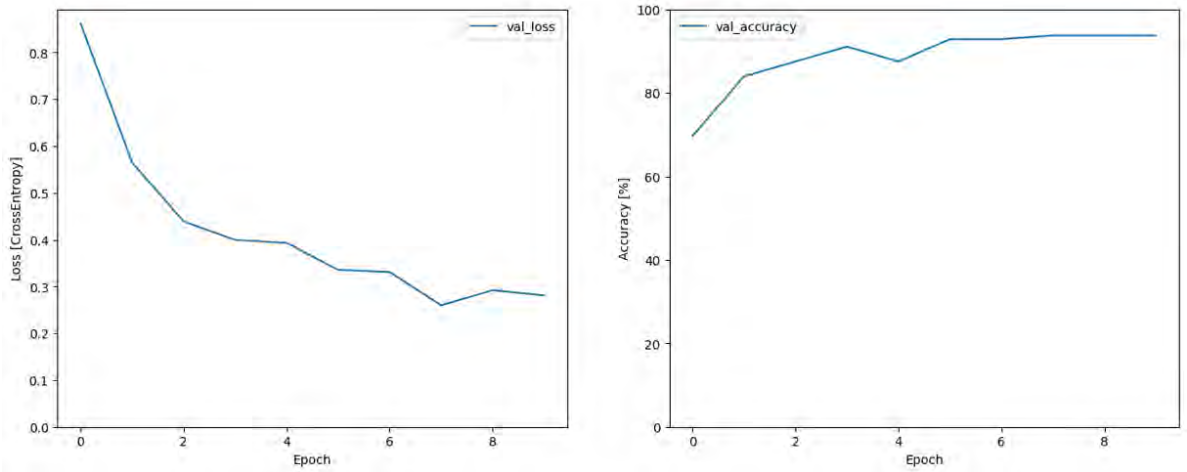


Abbildung 4.5: Kondensatormikrofon: Loss-Accuracy über 10 Epochen
Parameter: 16 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

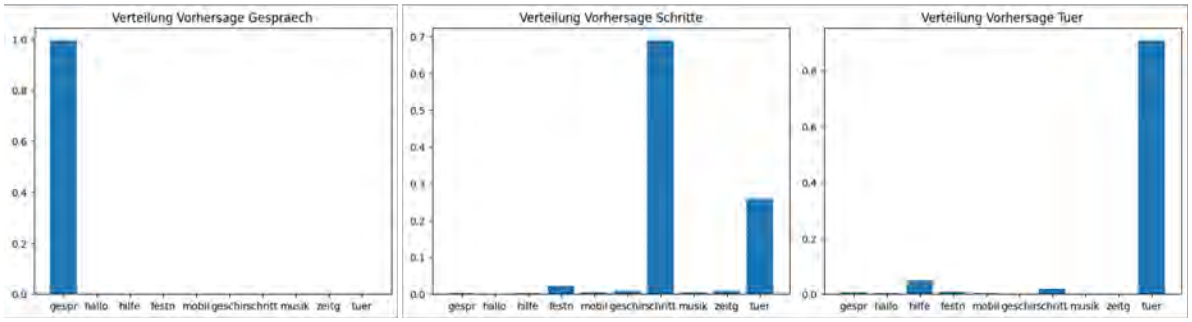
Der Vergleich der Verteilung der Vorhersage zeigt bei der Betrachtung der Audiosignale “Sprache”, “Schritte” und “Tür” größere Abweichungen (Abb. 4.6) als bei einer Abtastrate von 44,1 kHz. Alle anderen Klassen werden mit einer Genauigkeit von über 90% erkannt.

Die kleinste, im Rahmen dieser Arbeit eingesetzte Abtastrate von 8 kHz zeigt im Vergleich zu höheren Abtastraten eine noch größere Ungenauigkeit der Vorhersage bei der Erkennung des Signals “Tür”. Das Audiosignal wird nicht korrekt, sondern mit höherer Wahrscheinlichkeit als “Schritt” erkannt und hat eine Vorhersagegenauigkeit von unter 50% (Abb. 4.7), wobei andere Klassen eine Vorhersagegenauigkeit von über 90% haben.

Zusammenfassung der verglichenen Abtastraten

Erwartungsgemäß liefert eine Abtastrate von 44,1 kHz die beste Erkennung bei ähnlichen Signalen wie “Gespräch” im Vergleich zu dem Wort “Hallo” oder das Öffnen und Schließen einer Tür im Vergleich zu Schritten auf einem Fußboden. Aus den Untersuchungen ergibt sich, dass eine Abtastrate von 16 kHz unter Verwendung der weiteren Basisparametrierung zur Erkennung ausreichend ist. Dies bestätigen auch die in der Praxis eingesetzten CNN-Modelle zur Geräuscherkennung, die, wie im Kapitel 3.4.5 beschrieben, mit der Abtastastrate von 16 kHz arbeiten.

44,1 kHz Abtastrate



16,0 kHz Abtastrate

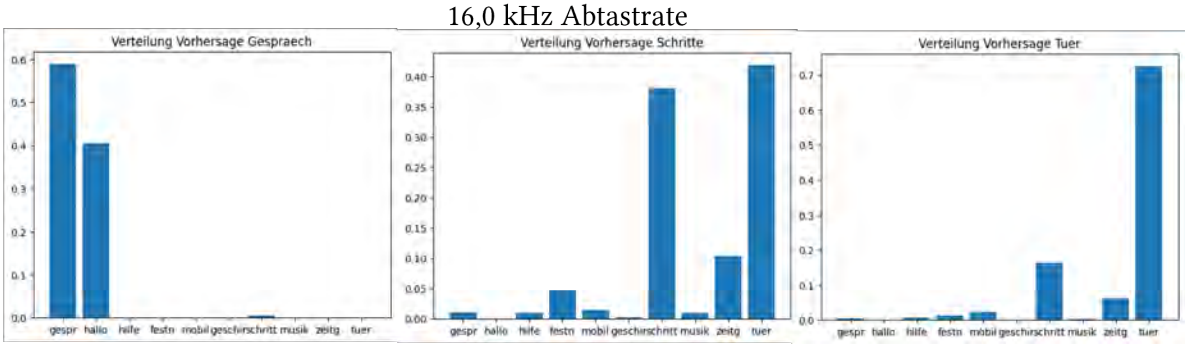


Abbildung 4.6: Kondensatormikrofon: Vergleich Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 und 16,0 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

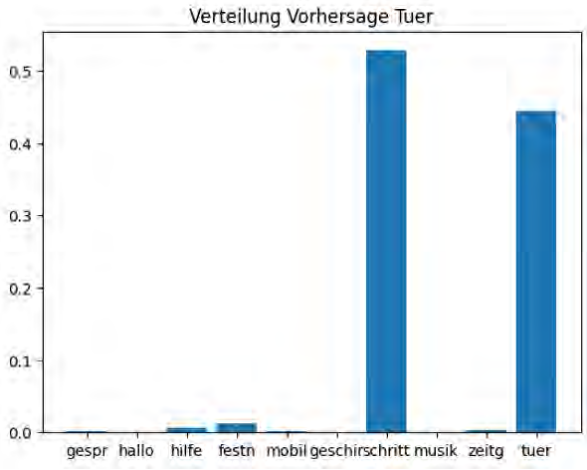


Abbildung 4.7: Kondensatormikrofon: Verteilung Vorhersage “Tür” über 10 Epochen
Parameter: 8 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

4.1.5 Signalaufnahmedauer

CNN-Modelle wie “Torch Audio Speech Command Classification” verwenden eine Länge der Trainingsaudiodaten von einer Sekunde, während das ESC-Modell von Piczak, wie im Kapitel 3.4.5 erwähnt, für Umweltgeräusche eine Länge von fünf Sekunden verwendet. Versuche mit veränderten Längen der Audioaufnahmen liefern die folgenden Ergebnisse:

Länge der Trainingsaudiodaten: 2,0 Sekunden

Die Verdoppelung der Signallänge verbessert die Erkennung von Geräuschen und Sprache. Demgegenüber steht eine Verdoppelung des Datenvolumens. Einzelne Worte wie “Hilfe” und “Hallo” werden in der Regel von Personen in weniger als einer Sekunde gesprochen. Aufgrund der kontinuierlich rollierenden Erfassung der Signale im Wohnumfeld, kommt es zum Abschneiden von Wortlauten bei der Erfassung von Sprache. Bei Geräuschen wie Musik ist die Genauigkeit der Erkennung geringer beeinflusst, da das Signal über einen längeren Zeitraum erzeugt wird und es somit seltener zu einem Abschneiden des Signals kommt. Insgesamt liegt die Erkennung der Audiosignale erwartungsgemäß hoch. Abbildung 4.8 zeigt zwei ausgewählte Klassen, bei denen die Erkennung unter 90% liegt.

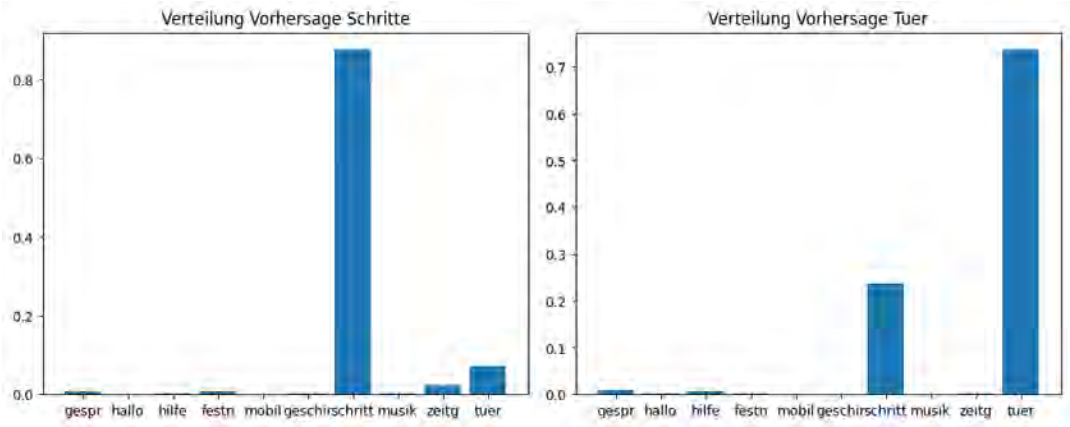


Abbildung 4.8: Kondensatormikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 2 Sekunden
Quelle: Eigene Darstellung

Länge der Trainingsaudiodaten: 0,5 Sekunden

Eine Halbierung der Länge der Trainingsaudiodaten reduziert proportional das Datenvolumen und verkürzt damit die Verarbeitungszeiten der Audiodatenextraktion und Aufbereitung für das CNN-Modell als auch die Berechnungszeit des CNN-Modells. Auf der anderen Seite

besteht die Gefahr, dass zu kurze Signale aufgrund der begrenzten Anzahl von Merkmalen wie Bandbreite und Signalverlauf durch ein CNN-Modell nicht korrekt klassifiziert werden. Im Rahmen der Studie werden für eine Audiosignallänge von 0,5 Sekunden die in Abbildung 4.9 berechneten Werte für Loss (0,58) und Accuracy (85%) erreicht. Diese Werte liegen deutlich niedriger als für Trainingsdaten mit einer Länge von einer oder mehr Sekunden.

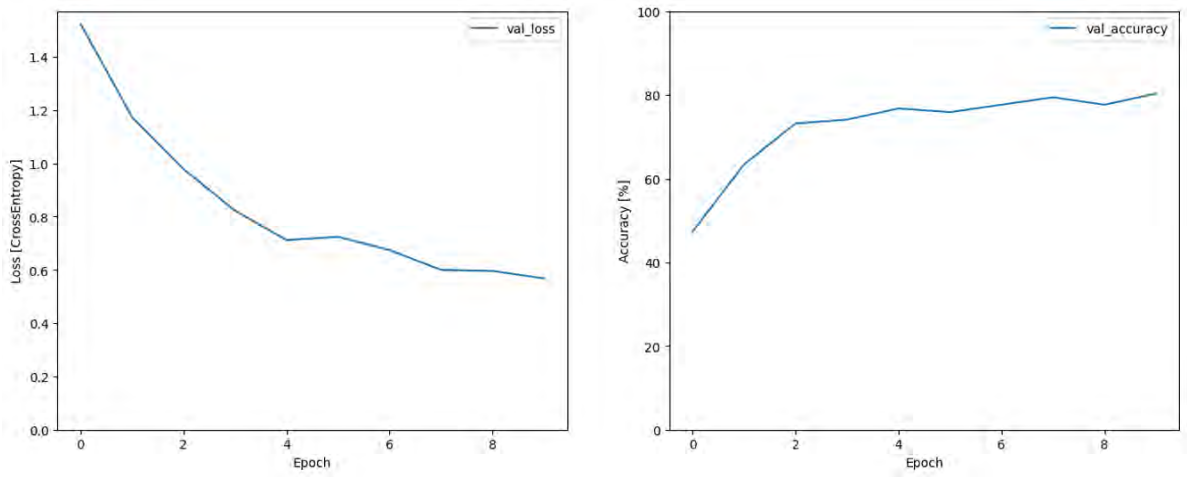


Abbildung 4.9: Kondensatormikrofon: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 0,5 Sekunden
Quelle: Eigene Darstellung

Abbildung 4.10 zeigt ausgewählte Klassen und ihre Vorhersagegenauigkeit auf Grundlage der Testaufnahmen. Die Klasse “Schritte” wird durch das Modell nicht erkannt. Türgeräusche werde teilweise als Sprache interpretiert. Wie bereits für die Dateien mit einer Länge von 2 Sekunden beschrieben, sind weitere Parameter zur besseren Vergleichbarkeit unverändert geblieben.

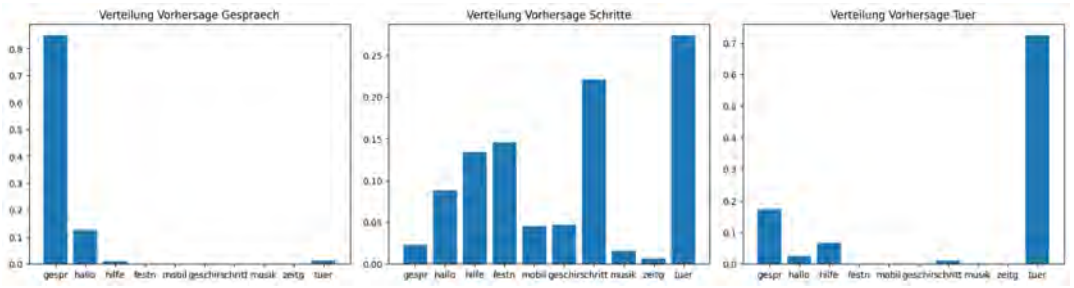


Abbildung 4.10: Kondensatormikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 0,5 Sekunden
Quelle: Eigene Darstellung

4.1.6 Anzahl Klassen

Tabelle 3.2 im Kapitel 3.3 ordnet einem Raum maximal 13 Audioklassen zu. Insgesamt betrachtet diese Arbeit 20 unterschiedliche Klassen. Um die Auswirkungen einer Vergrößerung der Anzahl Klassen zu berücksichtigen, wurden in der Studie bei sonst konstanter Parametrierung die Anzahl der Klassen im CNN-Modell verdoppelt und die Ergebnisse mit einer Anzahl von 10 Klassen verglichen. Einflüsse längerer Berechnungszeiten des Modells werden in dieser Arbeit nicht vertieft. Durch die Verdoppelung der Klassen verwendet das CNN-Modell für das Training insgesamt 2.400 Audiodateien mit einer Länge von einer Sekunde.

Abbildung 4.11 zeigt die Grafik "Loss-Accuracy" für 20 Klassen. Die Loss-Rate beträgt 0,1694 nach 10 Epochen und ist damit geringer als bei 10 Klassen. Die Accuracy liegt mit 96% vergleichbar hoch wie das Modell mit 10 Klassen.

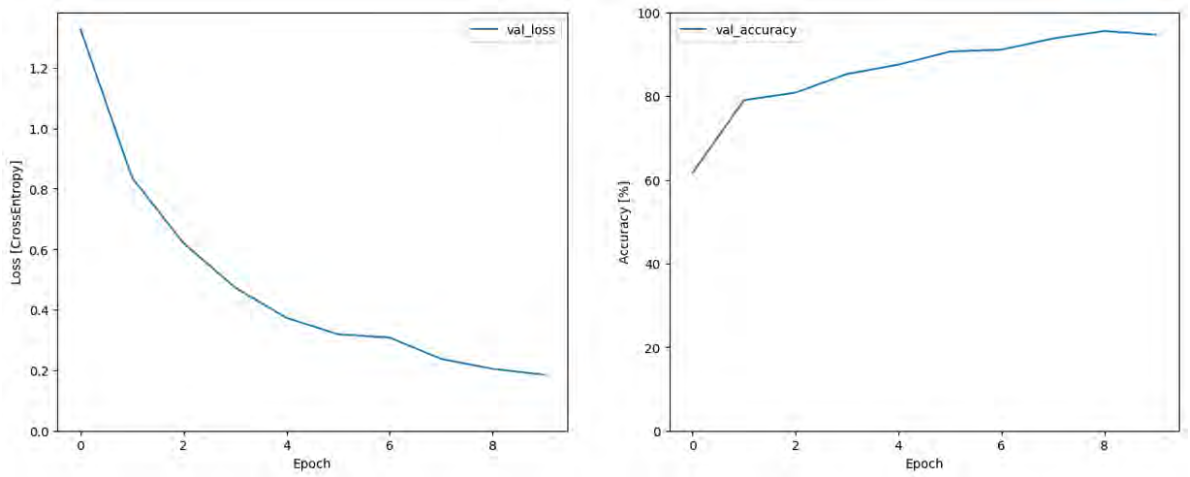


Abbildung 4.11: Kondensatormikrofon: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 20 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Mit der Verdoppelung der Klassen steigt die Möglichkeit einer fehlerhaften Erkennung deutlich an. Abbildung 4.12 zeigt die Confusion-Matrix mit 20 Klassen. Auffällig ist hier, dass einige Klassen fehlerhaft als "Schritte" vorhergesagt werden. Zur Überprüfung der Vorhersagegenauigkeit jeder Klasse werden wiederum die Testaufnahmen verwendet und auf 20 Klassen erweitert. Abbildung 4.13 zeigt ausgewählte Vorhersagen mit einer Genauigkeit von unter 100%.

Zusammenfassend zeigt sich bei der Verwendung von 20 statt 10 Klassen, dass die Erkennung einer Klasse ungeachtet des höheren Bedarfs an Rechenleistung oder Rechenzeit in fast allen Fällen über 98% liegt. Ausnahmen bilden die Klassen "Gespräch", "Schritte", "Zeitung", "Tür auf-zu" und "WC". Tabelle 4.1 zeigt eine Übersicht der Vorhersagegenauigkeit für alle 20 Klassen auf Basis der Testaufnahmen.

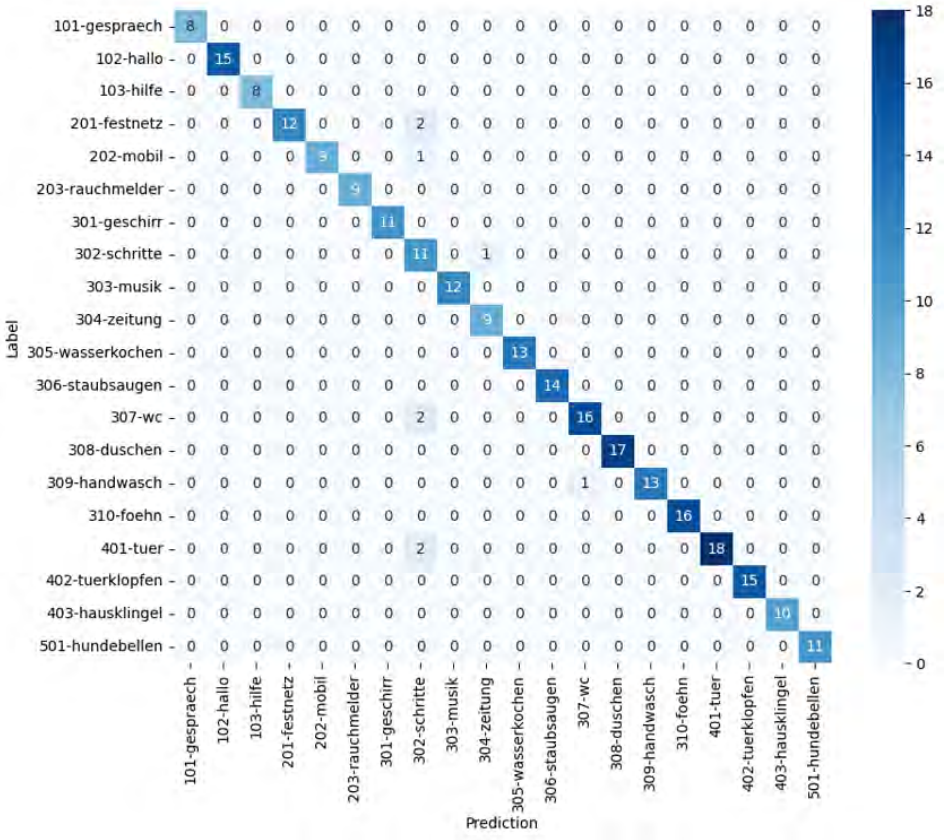


Abbildung 4.12: Kondensatormikrofon: Confusion-Matrix über 10 Epochen
Parameter: 44,1 kHz, 20 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Tabelle 4.1: Definition und Zuordnung Audioklassen zu Räumen

Audioklasse 1-10	**Genauigkeit**	**Audioklasse 11-20**	**Genauigkeit**
Gespräch	87,6%	Rauchmelder	99,9%
Hallo-Ruf	99,9%	Wasser kochen	99,9%
Hilfe-Ruf	100%	Staubsaugen	100%
Festnetz-Telefon	100%	WC	94,9%
Mobil-Telefon	99,9%	Duschen	100%
Geschirr	98,7%	Hand waschen	99,8%
Schritte	62,3%	Föhn	99,2%
Musik	99,9%	Türklopfen	98,1%
Zeitung lesen	89,7%	Klingeln	100%
Tür auf-zu	56,4%	Hundegebell	100%

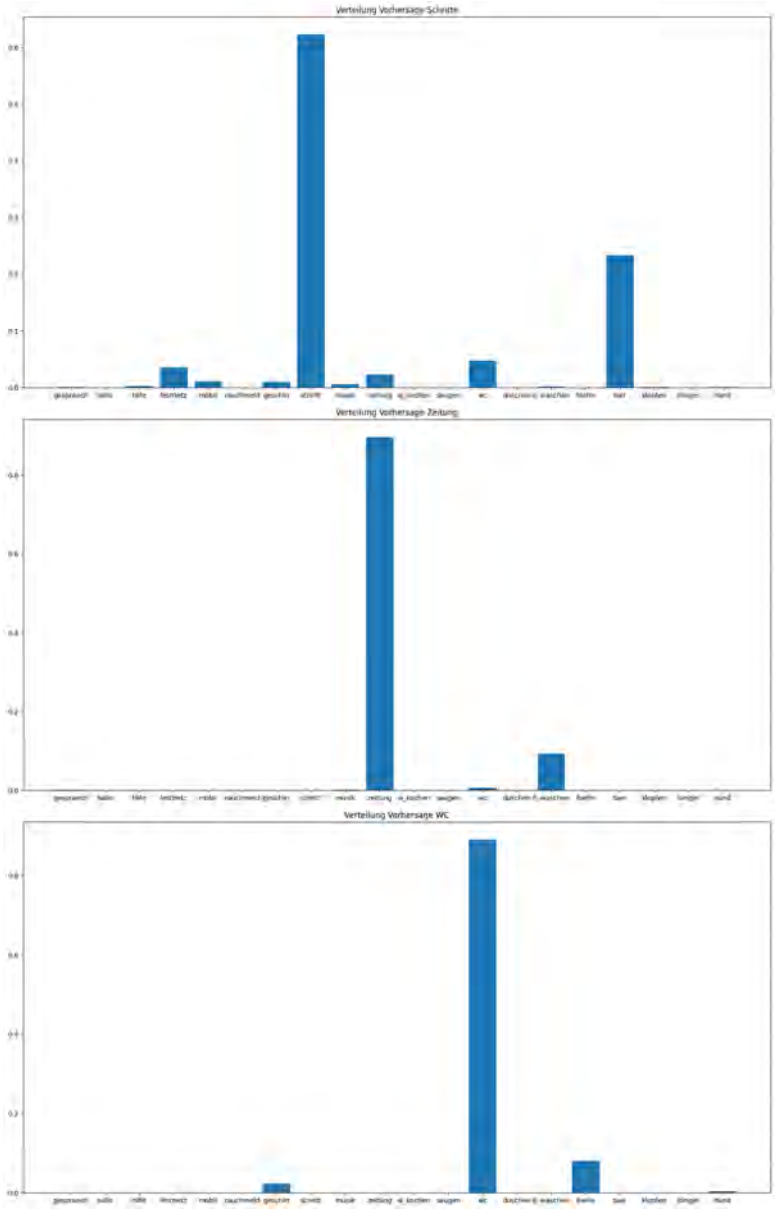


Abbildung 4.13: Kondensatormikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 kHz, 20 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

4.1.7 Anzahl WAV-Dateien pro Klasse

Das im Kapitel 3.4.2 beschriebene Python-Programm zur Generierung der Trainingsdaten aus Audiorohdaten ist in der Lage, über 2 Millionen Datensätze pro Klasse zu erzeugen. Die Anzahl an Trainingsdaten pro Klasse bestimmt neben der Qualität der Audioaufnahmen die Vorhersagegenauigkeit. Das verwendete “Simple-Audio-Recognition”-CNN-Modell greift in

seiner Originalversion auf 1.000 Trainingsdaten pro Klasse zurück. Im Rahmen dieser Studie werden 120 Trainingsdaten pro Klasse generiert. Im Folgenden werden die Ergebnisse einer Halbierung der Trainingsdaten auf 60 pro Klasse betrachtet.

Die Kennzahlen von Loss (0,2758) und Accuracy (89,0%) sind etwas geringer als die gleiche Konfiguration mit dem Kondensatormikrofon bei 120 Audiodateien pro Klasse. Die Accuracy von fast 90% wird bereits nach 4 Epochen erreicht. Danach pendelt der Wert um diesen Bereich. Damit erreicht das Modell bereits nach wenigen Epochen und nach einer kurzen Rechenzeit gegenüber dem Modell mit 120 Audiodateien pro Klasse eine hohe Genauigkeit (Abb. 4.14).

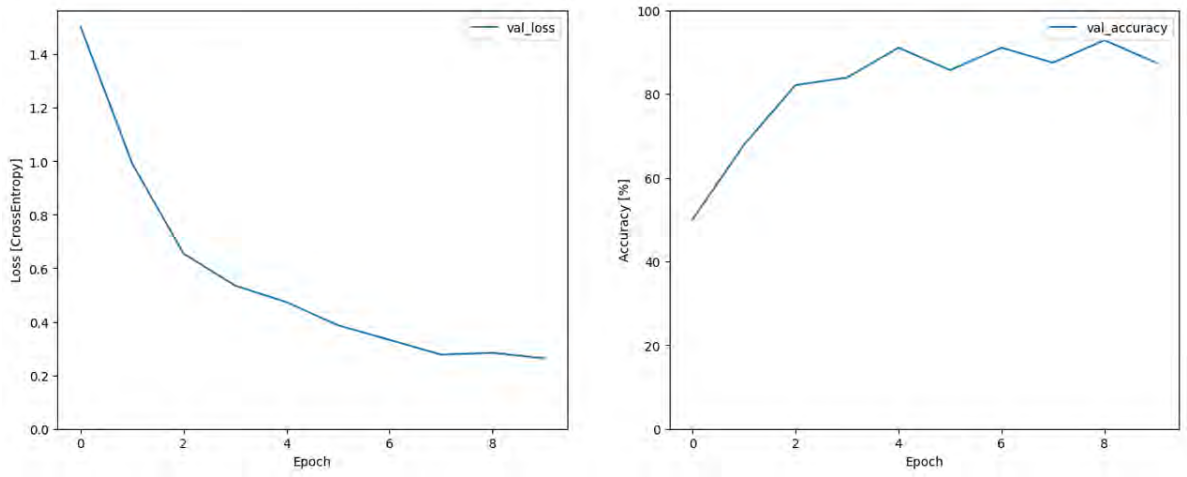


Abbildung 4.14: Kondensatormikrofon: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 60 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Aus der Confusion-Matrix ist zu erkennen, dass die Vorhersage für die Klassen “Zeitung” und “Tür” teilweise fehlerhaft ist (Abb. 4.15).

Zur Überprüfung der Vorhersagegenauigkeit jeder Klasse werden dieselben Testaufnahmen verwendet wie für die Grundkonfiguration. Abbildung 4.16 zeigt ausgewählte Vorhersagen mit einer Genauigkeit von unter 100%.

Die Erkennung von Geräuschen der Klassen “Gespräch”, “Zeitung” und “Tür” liegen bei etwa 70%. Aus diesem Grund ist es erforderlich, eine höhere Anzahl von Trainingsdateien pro Klasse zu verwenden. Eine Anzahl von 120 Dateien pro Klasse ist bei einer Anzahl von 10 Klassen zur Klassifizierung von Geräuschen und Sprache ausreichend.

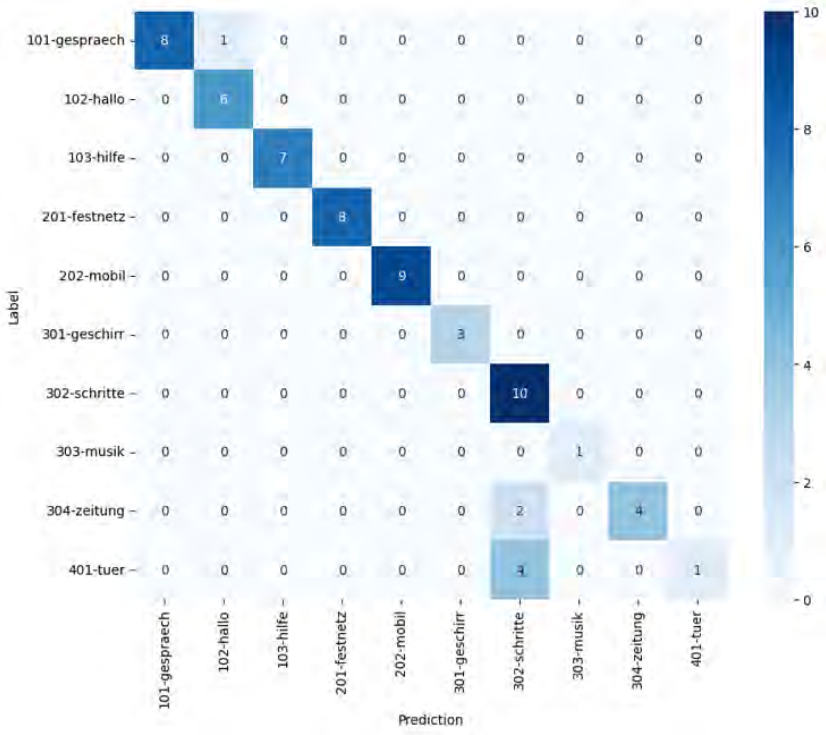


Abbildung 4.15: Kondensatormikrofon: Confusion-Matrix über 10 Epochen
Parameter: 44,1 kHz, 20 Klassen mit 60 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

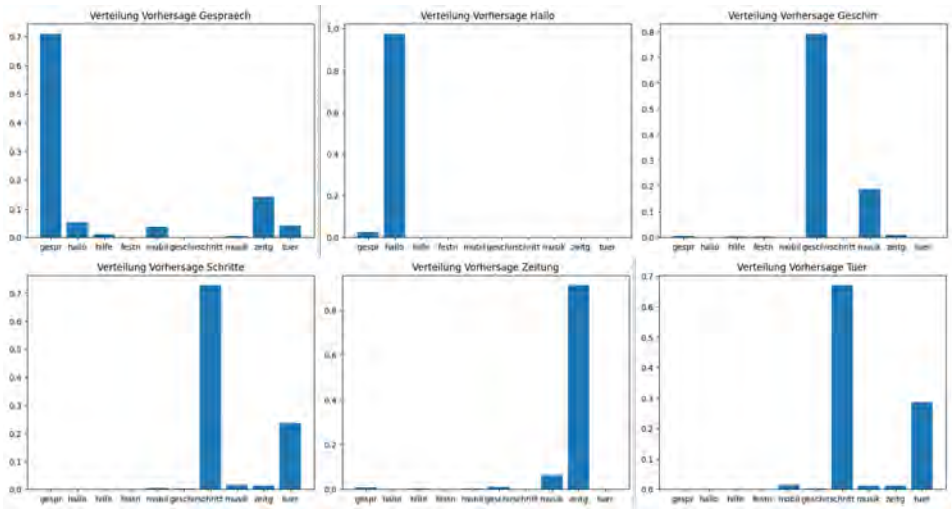


Abbildung 4.16: Kondensatormikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 44,1 kHz, 20 Klassen mit 60 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

4.2 Variation der Aufnahmequellen

Am Anfang der Studie wird das Kondensatormikrofon mit dem Audiointerface als Referenzaufnahmequelle verwendet. Zur Geräusch- und Spracherkennung im Wohnumfeld älterer Menschen eignen sich diese Hardwarekomponenten aus den in den Kapiteln 2.1.1 und 3.2.2 beschriebenen Gründen nicht. Im Wohnumfeld kommen Miniaturmikrofone zu Einsatz. Im Folgenden wird untersucht, wie die beiden eingesetzten Miniaturmikrofone MEMS-Mikrofon und integriertes Tabletmikrofon die Erkennung von verschiedenen Audiosignalen im Vergleich zum Kondensatormikrofon als Referenzaufnahmequelle gewährleisten.

Um die Vergleichbarkeit der Untersuchung sicherzustellen, bleiben alle Audiosignalparameter gegenüber der Grundkonfiguration unverändert. Die Testaufnahmen zur Überprüfung der Vorhersagegenauigkeit werden mit den Miniaturmikrofonen durchgeführt. Sie folgen hinsichtlich der Merkmale wie Vollständigkeit des Signals den oben beschriebenen Anforderungen und sind mit den Testaufnahmen des Kondensatormikrofons vergleichbar.

4.2.1 MEMS-Miniaturmikrofon und ESP32-Mikrocomputer

Mit dem CNN-Modell “Simple Audio Recognition” werden Aufnahmen des MEMS-Mikrofons mit den Abtastraten 24 kHz, 16 kHz und 8 kHz untersucht. Die weiteren Parameter der Signalverarbeitung wie Länge des Signals als auch die Anzahl der Klassen und Audiodateien pro Klasse entsprechen der Grundparametrierung und bleiben konstant.

Abtastrate 24 kHz

Die höchste für das MEMS-Mikrofon gewählte Abtastrate beträgt aus den in Kapitel 3.4.1 beschriebenen Gründen 24 kHz. Das CNN-Model berechnet für diese Abtastrate die in Abbildung 4.17 ermittelten Werte für Loss und Accuracy. Nach 10 Epochen erreicht das Loss einen Wert von 0,1398 und die Accuracy liegt bei über 96%.

Die Confusion-Matrix (Abb. 4.18) belegt ebenfalls die hohe Erkennung der Audiosignale und die damit verbundene korrekte Klassifizierung.

Abschließend werden, wie bereits mit den Aufnahmen des Kondensatormikrofons durchgeführt, die Testaufnahmen zur Vorhersagegenauigkeit jeder Klasse sequentiell in das CNN-Modell gegeben. Die höchsten Falschvorhersagen werden für die Klassen “Gespräch”, “Hallo”, “Schritte” und “Zeitung” gemacht (Abb. 4.19), wobei die Vorhersage für die Klasse “Schritte” unter 70% liegt.

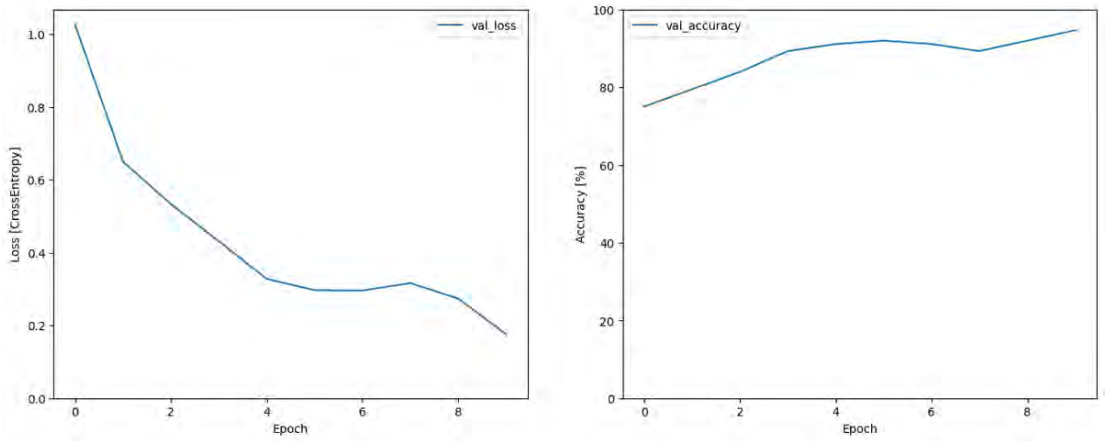


Abbildung 4.17: MEMS-Mikrofon: Loss-Accuracy über 10 Epochen
Parameter: 24 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

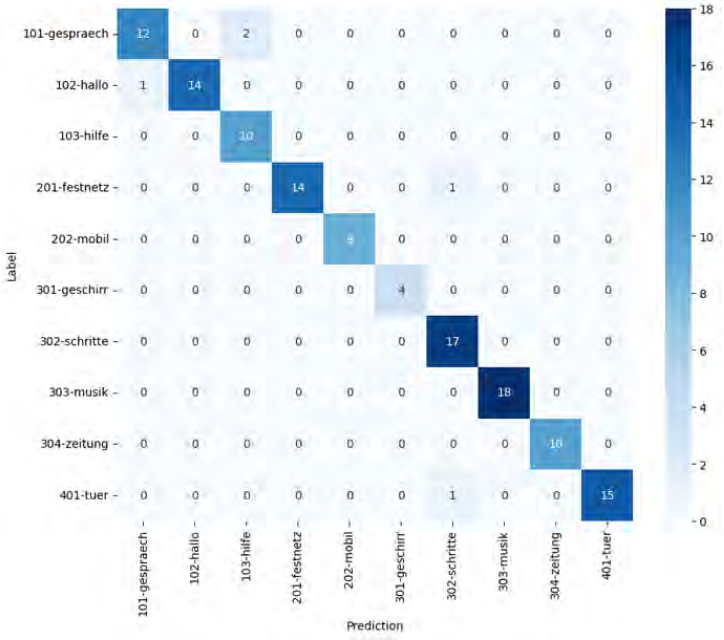


Abbildung 4.18: MEMS-Mikrofon: Confusion-Matrix über 10 Epochen
Parameter: 24 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Die Erkennung von Aufnahmen mit dem MEMS-Mikrofon und einer Abtastrate von 24 kHz ist erwartungsgemäß geringer als die Erkennung von vergleichbaren Aufnahmen mit dem Kondensatormikrofon und einer Abtastrate von 44,1 kHz. Sie erfüllen aus Sicht des Autors allerdings die Anforderung einer Klassifizierung von Geräuschen und Sprache im Wohnumfeld auf Basis der gewählten weiteren Parameter.

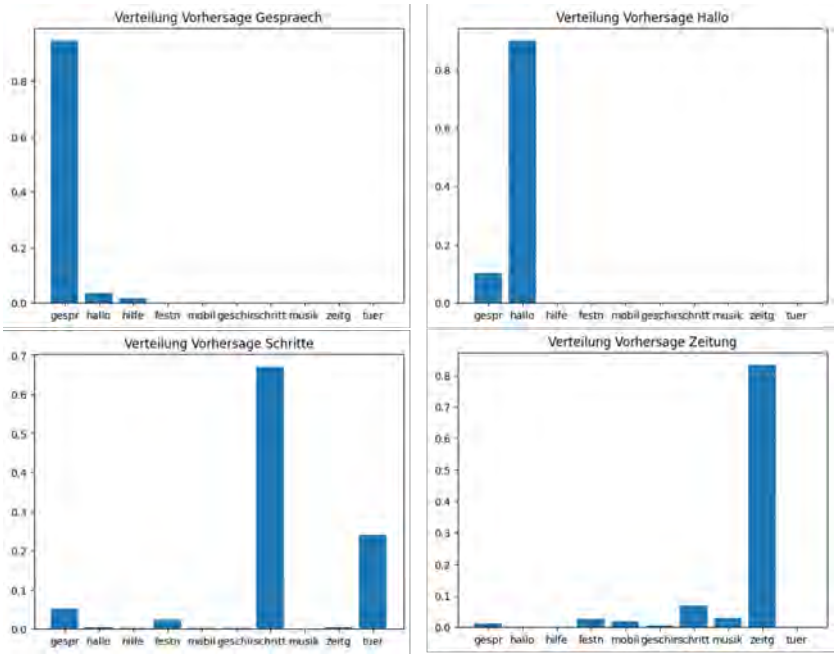


Abbildung 4.19: MEMS-Mikrofon: Verteilung Vorhersage über 10 Epochen
Parameter: 24 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Abtastrate 16 kHz

Eine Abtastrate von 16 kHz bietet die Möglichkeit eines direkten Vergleichs der Erkennung von Audiosignalen zwischen dem Kondensatormikrofon und dem MEMS-Mikrofon. Die Auswertung der Berechnungen ergeben für das MEMS Mikrofon vergleichbare Ergebnisse wie für das Kondensatormikrofon. So liegt die Accuracy in beiden Fällen bei 94,5%.

In der Abbildung 4.20 werden die Klassen mit der geringsten Erkennung für das MEMS- und das Kondensatormikrofon beispielhaft gegenübergestellt.

Die Erkennung von Signalen, die mit dem MEMS-Mikrofon mit einer Abtastrate von 16 kHz aufgenommen wird, ist in diesem Fall - unerwarteter Weise - besser als mit dem Kondensatormikrofon als Referenz. Insbesondere bei der Erkennung der Schritte ist das MEMS-Mikrofon in der Studie besser geeignet. Eine Erklärung könnte in der unterschiedlichen Testaufnahme liegen. Da die Qualität der Erkennung für eine Anwendung im Wohnumfeld ausreicht, werden an dieser Stelle keine vertiefenden Untersuchungen zur Ursache durchgeführt.

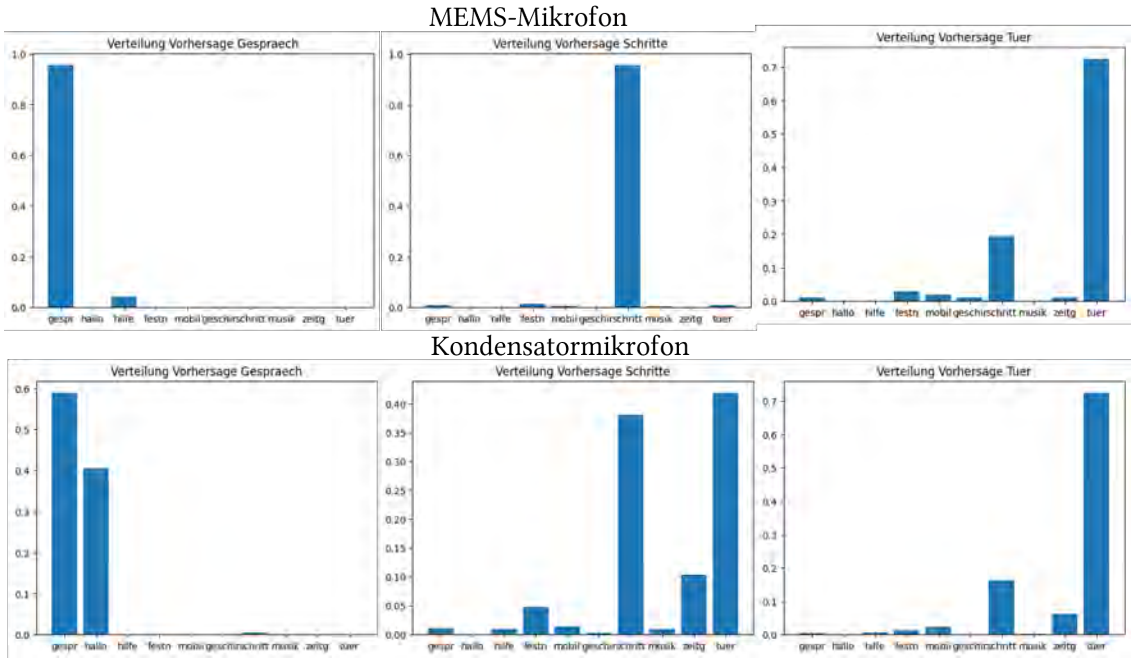


Abbildung 4.20: MEMS-und Kondensatormikrofon: Vergleich der Vorhersage über 10 Epochen
Parameter: 16 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Abtastrate 8 kHz

Durch die guten Ergebnisse mit einer Abtastrate von 16 kHz bietet sich eine Gegenüberstellung der Vorhersagen zwischen dem MEMS- und dem Kondensatormikrofon für eine Abtastrate von 8 kHz an. Wie oben ermittelt, ist diese Abtastrate für die Sprach- und Geräuscherkennung grenzwertig und kann zu einer falschen Klassifizierung führen. Abbildung 4.21 zeigt den Vergleich bei einer Abtastrate von 8 kHz.

Auch in diesem Fall ist die Vorhersage der korrekten Klasse vergleichbar oder besser als die Klassifizierung der Testaufnahme des Kondensatormikrofons, das bei der Vorhersage “Tür” mit einer höheren Wahrscheinlichkeit das Signal als “Schritte” erkennt.

Die Auswertung der Grafiken ergibt für das MEMS-Mikrofon, dass die Klassen “Hilfe” und “Zeitung” nicht deutlich erkannt werden (Abb. 4.22). Ein Hilferuf soll einen Alarm generieren. Eine falsche Klassifizierung dieses Signals als False Positive im Kontext einer Confusion-Matrix wird als kritisch bewertet. Aus diesem Grund wird von einer Abtastrate von 8 kHz abgeraten.

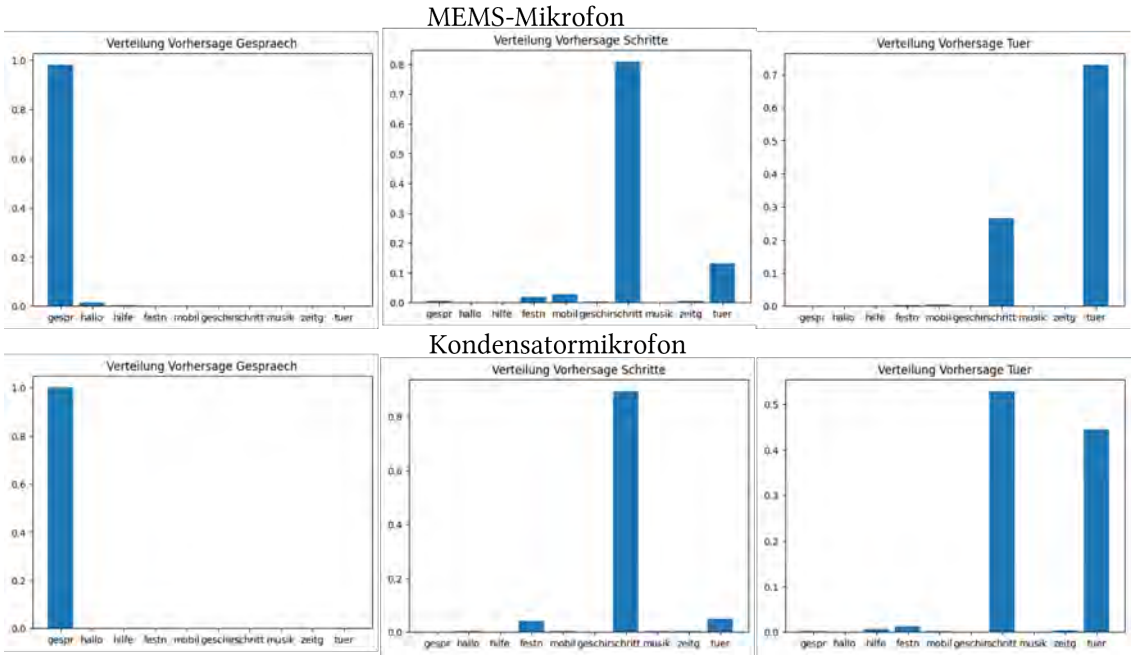


Abbildung 4.21: MEMS-und Kondensatormikrofon: Vergleich der Vorhersage über 10 Epochen
Parameter: 8 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

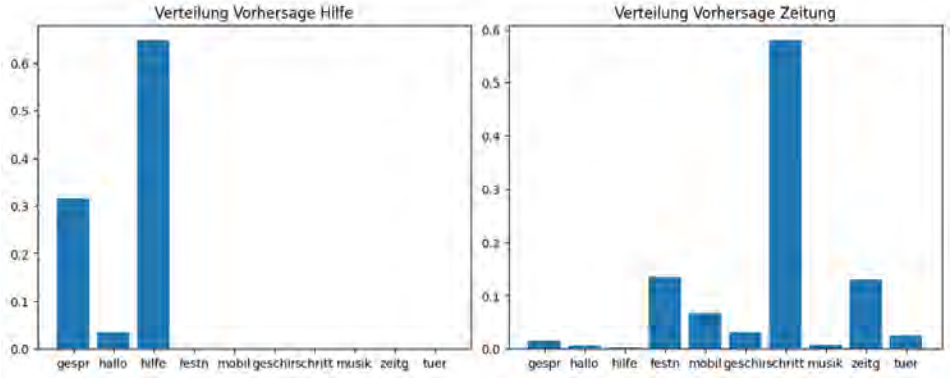


Abbildung 4.22: MEMS-Mikrofon: Vorhersage Hilfe und Zeitung über 10 Epochen
Parameter: 8 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

4.2.2 Tablet mit integriertem Mikrofon

Der Einsatz eines Tablets zur Erkennung von Geräuschen und Sprache bietet sich aus verschiedenen Gründen an. Tablets können parallel, wie im Kapitel 2.5.1 vorgestellt, als Intelligenter Bilderrahmen genutzt werden, um mit einer visuellen Erfassung die Stimmungslage älte-

rer Menschen zu erkennen. Dasselbe Gerät kann die Audiofunktion zur Geräusch- und Spracherkennung nutzen und die bestehende Applikation in ihrer Vorhersage unterstützen.

Im Folgenden werden die Ergebnisse der Untersuchung mit Testaufnahmen mit dem in Kapitel 3.2.2 beschriebenen Medion Tablet vorgestellt und mit Ergebnissen der Klassifizierung mit dem Kondensatormikrofon verglichen.

Abtastrate 44,1 kHz

Abbildung 4.23 zeigt die erwartungsgemäß guten und über den Ergebnissen des Kondensatormikrofons liegenden Werten für Loss (0,1345) und Accuracy (96,9%) über 10 Epochen mit einer Abtastrate von 44,1 kHz.

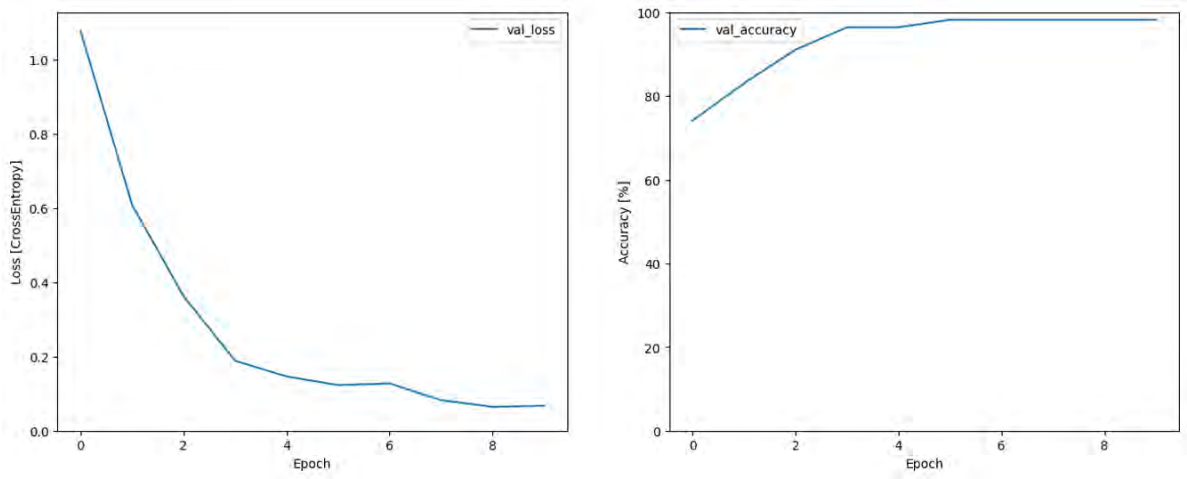


Abbildung 4.23: Integriertes Tabletmikrofon: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Die Confusion-Matrix zeigt im Vergleich zum Kondensatormikrofon keine nennenswerten Auffälligkeiten in der Matrix Klasse (label) zu Vorhersage (prediction) (Abb. 4.24).

Das CNN-Modell sagt für die Testaufnahmen mit dem Tablet für alle Klassen eine Genauigkeit von über 98% voraus (Abb. 4.25). Da die Audiorohdaten zur Generierung der Trainingsdaten parallel und im gleichen Abstand zur Schallquelle aufgenommen wurden, sind die Kennzahlen wie Loss und Acccuracy der Mikrofone des Tablets und des Kondensatormikrofons vergleichbar.

Die Ergebnisse der Erkennung von Geräuschen und Sprache mit dem Tablet bei einer fest eingestellten Abtastrate von 44,1 kHz demonstrieren die Eignung zum Einsatz im Wohnumfeld.

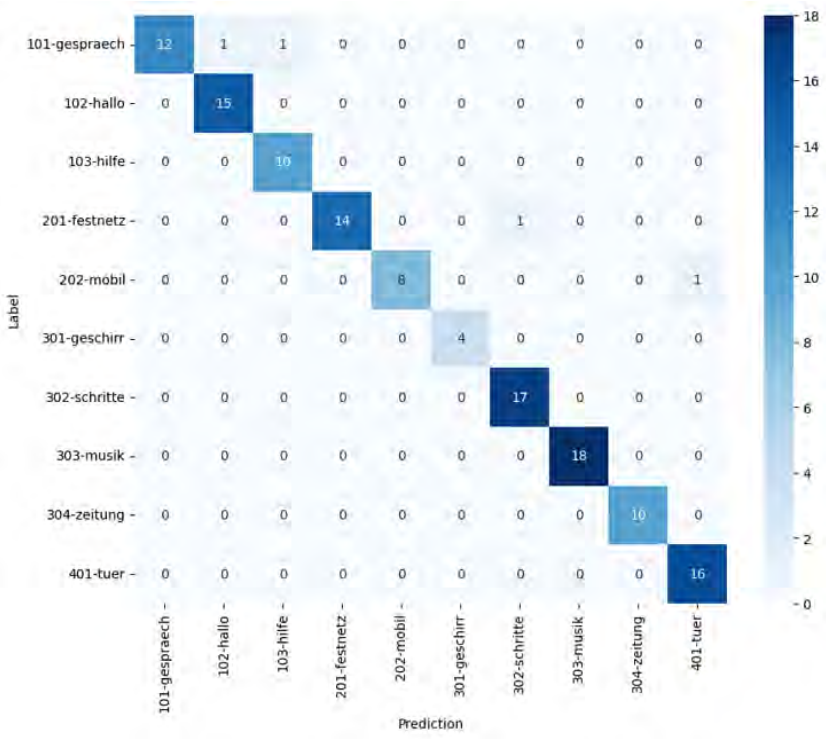


Abbildung 4.24: Integriertes Tabletmikrofon: Confusion-Matrix über 10 Epochen
Parameter: 24 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Die Qualität der Erkennung ist auf Basis der Berechnungen des CNN-Modells und der Testaufnahmen besser als mit dem Kondensatormikrofon.

4.3 CNN-Modelle und deren Parametrierung

Am Anfang der Studie wird zur Basiskonfiguration und zur Bestimmung der optimalen Signalverarbeitungsparameter das erste CNN-Modell "Simple Audio Recognition" eingesetzt. Für die beiden weiteren, im Kapitel vorgestellten Modelle "Speech Command Classification with Torchaudio" und "ESC-Dataset for Environmental Sound Classification" werden nachfolgend Versuche mit 10 Klassen der generierten Trainingsdaten beschrieben und deren Eignung zur Geräusch- und Spracherkennung im Wohnumfeld geprüft.

2. Modell Speech Command Classification with Torchaudio

Wie im Kapitel 3.4.5 beschrieben, konzentriert sich dieses zweite Modell auf die Erkennung von 35 Sprachbefehlen. Die zugehörigen Trainingsdaten werden von mehr als 2.500 ver-

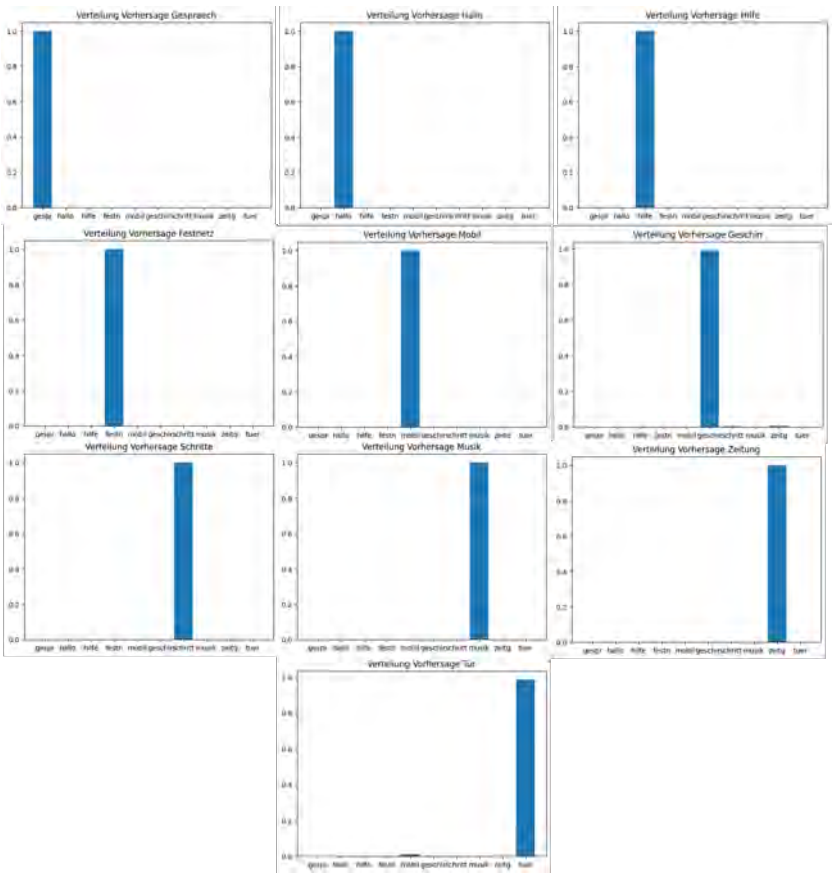


Abbildung 4.25: Integriertes Tabletmikrofon: Vorhersage Hilfe und Zeitung über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

schiedenen Personen gesprochen und bestehen aus über 100.000 Datensätzen (sogenannten Ein-Wort-Äußerungen oder “utterances”). Als Netzwerk wird das M5-Modell verwendet.

Im Rahmen der Studie wird überprüft, inwieweit das Modell neben Sprachbefehlen auch Geräusche korrekt klassifizieren kann. Als Trainingsdaten werden die generierten 44,1 kHz Audiodateien verwendet, die mit dem Kondensatormikrofon aufgenommen wurden. Alle für das CNN-Modell verwendeten Audiosignalparameter wie Abtastrate, Aufnahmedauer und Anzahl WAV-Dateien pro Klasse bleiben zur Vergleichbarkeit der Modelle unverändert.

Das CNN-Modell “Speech Command Classification” verwendet wie das erste Modell “Simple Audio Recognition” den Adam-Optimierer. Als Parameter kann die Learning Rate und das Weight-Decay angepasst werden, wobei ein Scheduler die Learning Rate nach 20 Epochen um den Faktor 10 verkleinert. Im ersten Modell “Simple Audio Recognition” wurden diese Parameter nicht verändert und die Anzahl konstant auf 10 Epochen eingestellt. Das Modell besitzt eine aktivierte Early-Stopping-Funktion, die die Anzahl der Epochen nach Erreichung

einer stabilen Accuracy begrenzt. Im zweiten Modell “Speech Command Classification” ist diese Funktion nicht implementiert.

Die Loss- und Accuracy-Werte werden im Originalprogrammcode “Speech Command Classification” nicht grafisch ausgegeben. Abbildung 4.26 zeigt eine durch den Autor erstellte Auswertung von Loss und Accuracy über 10 Epochen. Die Accuracy von 95% wird bereits nach 7 Epochen erreicht.

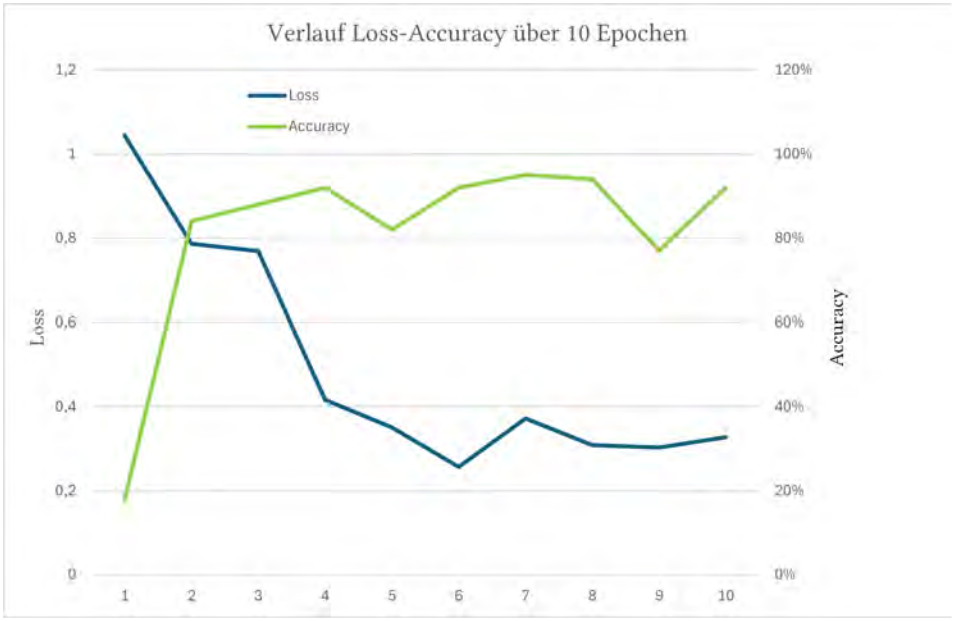


Abbildung 4.26: Modell “Speech Command Classification”: Loss-Accuracy über 10 Epochen
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde
Quelle: Eigene Darstellung

Die Confusion-Matrix (Abb. 4.27) des Modells beweist, dass es für eine Erkennung von Sprache als auch von Geräuschen geeignet ist. Diese Studie arbeitet mit der höchsten Abtastrate der Trainingsdaten von 44,1 kHz. In seiner Ursprungsversion verwendet das Modell den Datensatz von Speech-Commands mit einer Abtastrate von 16 kHz. Für alle Klassen wird eine Accuracy von über 98% erreicht. Durch weitere Optimierungen wie Early-Stopping als auch Anpassung der Learning Rate kann das Modell weiterentwickelt werden.

Das zweite Modell “Speech Command Classification” liefert im Vergleich zum ersten Modell etwas bessere Werte für die Accuracy. Der Datensatz kann durch Metaparameter wie Sprecher und Abtastrate erweitert werden. Mit einer Ergänzung des Modells wie einem Optimierer zur Begrenzung der Epochen kann das Modell weiter optimiert werden.

Klasse	101-gespr	102-hallo	103-hilfe	201-festn	202-mobil	301-gesch	302-schrit	303-musik	304-zeitg	401-tuer
101-gespraech	99,58%	0,42%								
102-hallo		100,00%								
103-hilfe		0,17%	99,42%				0,17%			0,25%
201-festnetz				98,83%			1,08%			0,08%
202-mobil				0,25%	99,50%		0,25%			
301-geschirr				0,17%		99,75%	0,08%			
302-schritte							100,00%			
303-musik				0,08%				99,92%		
304-zeitung							0,92%		99,08%	
401-tuer				0,08%			0,08%			99,83%

Abbildung 4.27: Modell “Speech Command Classification”: Confusion-Matrix
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde, 10 Epochen
Quelle: Eigene Darstellung

ESC - Environmentral Sound Classification

Das dritte Modell “ESC-Dataset for Environmental Sound Classification” von Karol Piczak wurde im Jahr 2015 vorgestellt und liefert im Programmverlauf umfangreiche Informationen über Extraktionsmethoden von Audiosignalen. Aufgrund zahlreicher Updates von Programmbibliotheken ist nur eine begrenzte Anzahl von Programmteilen und nicht das gesamte Programm lauffähig.

Karol Piczak kommentiert den Status seines Programms nach Rücksprache mit dem Autor dieser Arbeit mit den Worten “... especially with the dataset and the supporting code being outdated. For modern applications, I would recommend exploring newer frameworks like TensorFlow or PyTorch.” [51].

Im Rahmen der Studie wird der Programmcode durch den Autor angepasst und ESC-Datensätze durch die generierten Audiodateien der 10 Klassen ersetzt. Dies ermöglicht grafische Auswertungen der Aufnahmen unterschiedlicher Klassen.

Die wesentlichen grafischen Darstellungen beziehen sich auf die im Kapitel 2.2.2 beschriebenen Extraktionsverfahren im Bereich der Mel-Spektrogramme, die MFCC sowie die Zero-Cross-Rate ZCR.

Abbildung 4.28 zeigt den Vergleich einer aggregierten Verteilung der ersten 13 MFCC von verschiedenen Geräusch- und Sprachaufnahmen. Insbesondere die Unterschiede in den ersten 5 Koeffizienten und das ZCR bieten die Möglichkeit einer Unterscheidung mit Hilfe von CNN-Modellen.

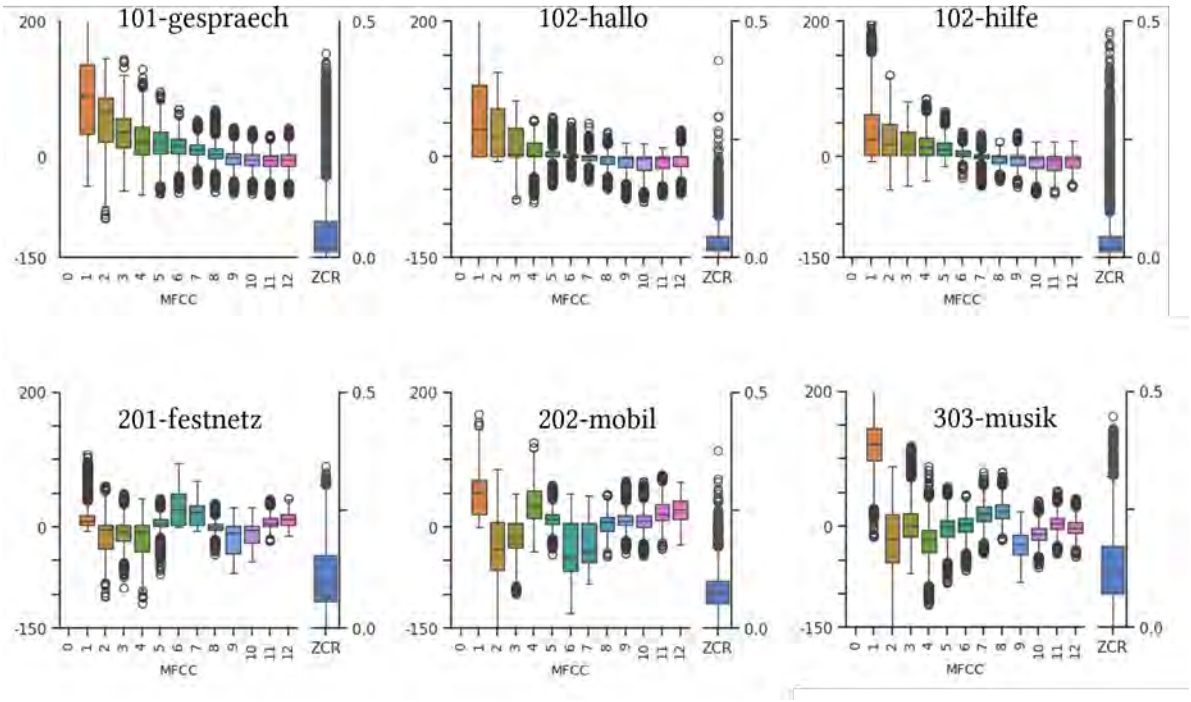


Abbildung 4.28: Modell “ESC50-Environmental Dataset”: Vergleich MFCC und ZCR
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde, 10 Epochen
Quelle: Eigene Darstellung

Für eine Klassifizierung bieten sich weiterhin die grafische Betrachtung der ersten und zweiten MFCC-Mittelwerte (Abb. 4.29) oder des ZCR im Vergleich zum ersten MFCC-Mittelwert an (Abb. 4.30).

Für jede Klasse bilden sich in den Diagrammen Cluster aus. CNN-Modelle können neue Eingaben dem 2-dimensionalen Cluster zuordnen und einer Klasse zuweisen. Obwohl der Programmcode “ESC - Environmentral Sound Classification” nicht mehr aktuell ist, demonstriert er Methoden, die zur Klassifizierung von Audiosignalen insbesondere von Umweltgeräuschen verwendet werden können. Detailliertere Untersuchungen und Implementierungen dieser Methoden in das erste oder zweite CNN-Modell würden den Rahmen dieser Arbeit überschreiten.

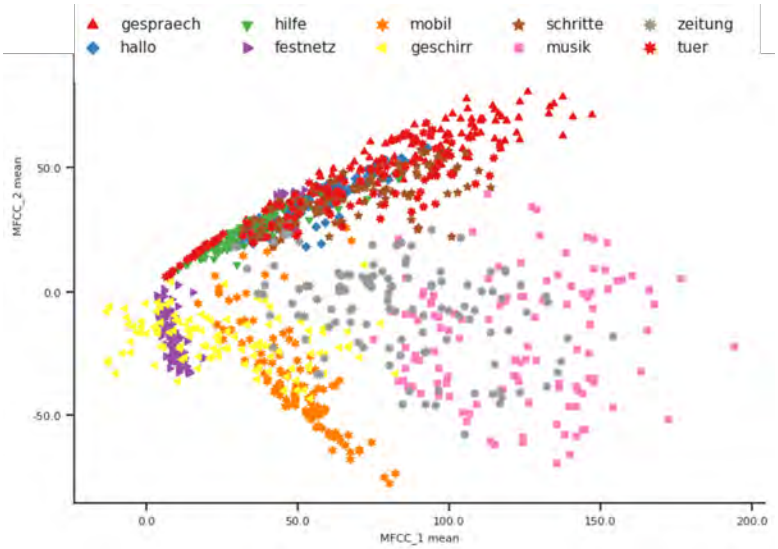


Abbildung 4.29: Modell “ESC50-Environmental Dataset”: Verhältnis MFCC 2 zu MFCC 1
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde, 10 Epochen
Quelle: Eigene Darstellung

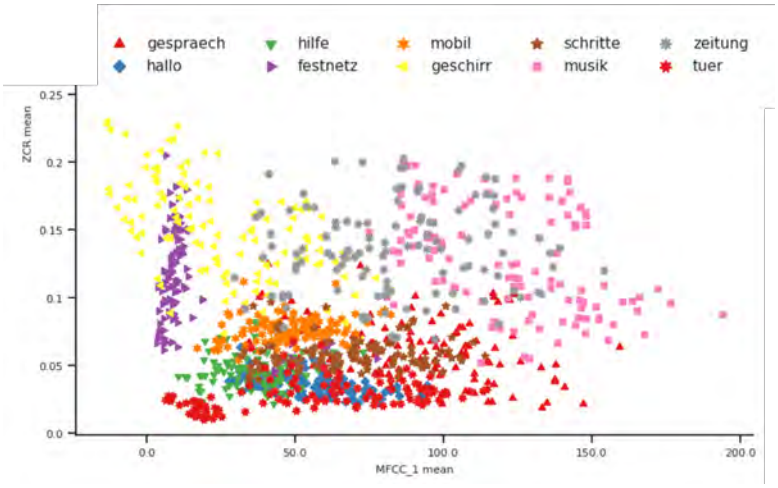


Abbildung 4.30: Modell “ESC50-Environmental Dataset”: Verhältnis ZCR zu MFCC 1
Parameter: 44,1 kHz, 10 Klassen mit 120 Dateien, Länge 1 Sekunde, 10 Epochen
Quelle: Eigene Darstellung

4.4 Diskussion der Studie

4.4.1 Vorgehensweise

Nach der Festlegung von Rahmenbedingungen wie betroffene Wohnbereiche, Raumakustik als auch die Positionierung von Aufnahmequellen wurden verschiedene Hardwarelösungen zur Audiosignalaufnahme und -verarbeitung untersucht, die für den Einsatz im Wohnumfeld in Frage kommen. Zur Erkennung von Geräuschen und Sprache wurden für die CNN-Modelle 20 Klassen definiert.

In der Studie wurden daraufhin unterschiedliche Parametrierungen der Audiosignalverarbeitung insbesondere die Reduzierung der Abtastrate vorgenommen und ausgewertet. Die Klassifizierung der Signale erfolgte mit dem ersten CNN-Modell als Referenz. Die Ergebnisse der Erkennung und Unterscheidung von Geräuschen und Sprache wurden danach mit einem zweiten Modell verglichen. Abschließend wurden Extraktionsverfahren aus einem dritten Modell verwendet, um eine Klassifizierung zum Beispiel durch Metadaten zu verbessern. Tabelle 4.2 zeigt eine Übersicht der Methoden, Parametrierungen und untersuchten CNN-Modelle.

Tabelle 4.2: Methoden, Parametrierungen und CNN-Modelle der Studie

Methoden:			
Aufnahmequellen:	Kondensatormikrofon	MEMS-Mikrofon	Tablet
Parametrierungen:			
Abtastrate	44,1/24 kHz	16 kHz	8 kHz
Signalaufnahmedauer	0,5 s	1,0 s	2,0 s
Anzahl Klassen	10	20	
Anzahl Dateien / Klasse	60	120	
CNN-Modelle:	Simple Audio Rec.	Speech Command Class.	ESC-50

4.4.2 Ergebnisse

Aus der Studie ergeben sich die folgenden, praxistauglichen Hardwarelösungen, Verfahren zur Erfassung, Verarbeitung und Datenextraktion von Audioaufnahmen sowie CNN-Modelle:

1. Hardwarelösung:
   - Im Wohnbereich und Küche: Tablet
   - In anderen Räumen: MEMS-Mikrofon mit Mikrocomputer
2. Audiosignalerfassung und -verarbeitung
   - Abtastrate: ≥16 kHz
   - Bittiefe: ≥16 Bit
   - Aufnahmedauer: 1-2 Sekunden
   - Anzahl Klassen: ≤14
   - Anzahl Trainingsdaten pro Klasse: ≥ 100
3. CNN-Modelle:
   - Simple Audio Recognition oder
   - Speech Command Classification
   - Optimierung durch Metadaten sinnvoll

In Wohnbereichen, die häufig frequentiert werden, bietet sich zur Geräusch- und Spracherkennung der Einsatz eines Tablets mit integriertem Mikrofon an. In diesem Zusammenhang sollten weitere Funktionen des Tablets wie ein digitaler Bilderrahmen genutzt werden. Dadurch wird eine gute Positionierung der Hardware zu Schallquellen im Raum ermöglicht. In den anderen im Kapitel 3.1.1 auf Seite 26 vorgestellten Räumen findet das MEMS-Mikrofon durch seine kleine Baugröße und geringe Kosten eine gute Anwendung. In diesen Bereichen ist mit einer geringeren Anzahl verschiedener Audiosignale und damit Klassen zur Erkennung zu rechnen.

Zur Audiosignalverarbeitung sollte zur Unterscheidung und Klassifizierung der Signale eine hohe Abtastrate gewählt werden. Die Untersuchungen in der Studie als auch Modelle zur Spracherkennung zeigen, dass eine Abtastrate von 16 kHz ausreichend ist.

In der Studie wurde die im Kapitel 3.4.2 verwendete Generierung von Trainingsdaten aus den im Wohnumfeld aufgenommenen Audiorohdaten verwendet. Diese Methode hat für eine zuverlässige Generierung von Trainingsdaten gesorgt. Die Anzahl der Trainingsdaten wurde auf 120 Dateien pro Klasse begrenzt. Das erste Modell wurde ebenfalls mit einer Anzahl von 60 Daten pro Klasse geprüft. Das erstellte Python-Programm erfüllte die Anforderung zur Generierung von Trainingsdaten. Einige Audiosignale wurden jedoch nicht immer optimal auf die Länge der Aufnahme zugeschnitten. An dieser Stelle besteht für eine praxistaugliche Anwendung Optimierungsbedarf.

Sowohl das CNN-Modell “Simple Audio Recognition” als auch das Modell “Speech Command Classification” liefern eine hohe Quote korrekter Klassifizierungen auf Basis der oben gewählten Parameter. Weitere Optimierungen an den Modellen wie die Learning Rate oder ein

Optimizer zur Begrenzung der Anzahl der Epochen können beide Modelle noch leistungsfähiger machen.

Ein wichtiges Kriterium in Bezug auf die Erkennung und Klassifizierung von Klassen, die zur Generierung eines Alarms wie der Hilferuf oder eines Rauchmelders genutzt werden, ist die Vermeidung von falschen Alarmen. Die oben beschriebenen Methoden haben diese Anforderung betrachtet und bei der Parametrierung durch Auswertung der Confusion-Matrix als auch der Vorhersagegenauigkeit berücksichtigt. Dies gilt ebenfalls für die Betrachtung, dass Klassen zur Erzeugung eines Alarms mit hoher Genauigkeit erkannt werden, damit der Alarm an ein externes Gerät übermittelt wird.

Das folgende Kapitel gibt einen Ausblick auf mögliche Systemerweiterungen in Kombination mit anderen Sensoriken wie die Aufnahme von Geräuschen und Sprache.

5 Zusammenfassung und Ausblick

5.1 Zusammenfassung

Ziel dieser Arbeit ist es, Methoden zur Audiosignalerfassung und -verarbeitung und ein geeignetes Deep-Learning-Verfahren zu finden, das Geräusche und Sprache im Wohnbereich älterer Menschen erkennt und klassifiziert, damit der Person im Falle einer Notsituation durch eine Mitteilung an Angehörige oder Personen von Pflegeeinrichtungen geholfen werden kann.

Grundlage dieser Arbeit sind die im Kapitel 2 betrachteten wissenschaftlichen und technischen Grundlagen im Bereich Audiosignalaufnahme- und verarbeitung und Deep-Learning-Methoden zur Klassifizierung von Audiosignalen als auch ausgewählte Systembeispiele von Raumüberwachung und Geräuscherkennung. Auf dieser Basis wurden im Kapitel 3 ein Konzept erarbeitet und ein Implementierungsweg beschrieben, der im Rahmen der im Kapitel 4 durchgeführten Studie untersucht und auf seine Machbarkeit validiert wurde. Die Ergebnisse zeigen geeignete Methoden zur Geräusch- und Spracherkennung im Wohnumfeld älterer Menschen und liefern damit eine Orientierung, wie eine praxistaugliche Implementierung realisiert werden kann.

Der Abschluss dieser Arbeit ist ein technischer Ausblick, wie die ermittelten Methoden in bestehende Systeme integriert oder mit anderen Sensoriken kombiniert werden können als auch ein persönlicher Ausblick, welche Inhalte im Rahmen von Folgearbeiten weiterentwickelt werden sollten.

5.2 Technischer Ausblick

Im nachfolgenden Abschnitt werden Integrationsmöglichkeiten der Geräusch- und Spracherkennung in bestehende Systeme, die teilweise bereits im Kapitel 2.5 vorgestellt wurden, als auch der Einsatz in Kombination mit weiteren Sensoriken im Wohnumfeld beschrieben.

Videoraumüberwachung

Eine Integration in ein System wie der in Kapitel 2.5.1 beschriebene Intelligente Bilderrahmen von Beyond Emotion stellt eine sinnvolle Erweiterung des Funktionsumfangs dar. Eine im Tablet integrierte Geräusch- und Spracherkennung ergänzt die Beurteilung der Stimmungslage einer älteren Person und informiert Verwandte über das gleiche technische System, falls eine Notsituation im Wohnumfeld erkannt wird. Als weitere Ausbaustufe kann unter Einhaltung der Privatsphäre eine bidirektionale Sprachverbindung aufgebaut und die Kamera im Tablet eingeschaltet werden.

Einer kontinuierlichen Videoraumüberwachung im Wohnumfeld älterer Menschen steht der Schutz der Privatsphäre entgegen. Wie bereits im Kapitel 2.5.2 zum Thema Telemonitoring beschrieben, würde die Methode nur eine geringe Akzeptanz finden. Während in der Studie das Telemonitoring als Alternative zur Videoraumüberwachung präsentiert wird, besteht ebenfalls die Möglichkeit, parallel zu einem MEMS-Mikrofon eine Miniaturkamera auf einen ESP32-Mikrocomputer zu integrieren. Diese wird ausschließlich bei der Erkennung einer Gefahren- oder Notsituation aktiviert. Damit wäre der Schutz der Privatsphäre weiterhin gewährleistet.

Vibrationssensorik

Wie im Kapitel 2.5.3 vorgestellt, beschreibt Zigel [20] in seiner Studie die Erkennung des Fallens von Personen mit einem, am Fußboden installierten Vibrationssensor und zusätzlicher Geräuschüberwachung. Aus den im Kapitel 4.2.1 auf Seite 79 beschriebenen Versuchen geht hervor, dass die Erkennung von Geräuschen wie Schritte, Tür öffnen und schließen als auch andere kurze Audiosignale im unteren Frequenzspektrum schwer zu klassifizieren sind. Eine falsche Klassifizierung in Kombination mit einem Alarmierungssystem hat entweder zur Folge, dass ein falscher Alarm ausgelöst wird (False Positive) oder dass kein Alarm ausgelöst wird, obwohl eine Person gestürzt ist (False Negative).

Im Zusammenwirken mit einem Vibrationssensor besteht die Möglichkeit, die Aussagegenauigkeit zu verbessern. Aus Sicht des Autors sollten die Signale des Sensors die primäre Entscheidung eines Fallereignisses liefern und die Geräuscherkennung zwischen dem Fallen einer Person und dem Umfallen von Gegenständen unterscheiden. Ergänzend sollte das Gesamtsystem nachgelagert weitere Geräusche erfassen und bei einer Erkennung von Hilferufen einen Alarm auslösen.

Microsoft Kinect

Erik Stone et al. [21] setzen in ihrer, im Kapitel 2.5.4 beschriebenen Studie statt eines Vibrationssensors einen Microsoft-Kinect-Sensor ohne Anwendung weiterer Methoden wie eine Audioüberwachung ein. Stone verweist auf eine hohe Anzahl falscher Alarme. Eine Geräusch- und Spracherkennung würde wie im Fall des oben beschriebenen Systems mit einem Vibrationssensor das Ergebnis einer korrekten Erkennung verbessern. Bei einem Einsatz eines Microsoft-Kinect-Systems könnten Miniaturkameras in Kombination mit ESP32-Mikrocomputern zusätzlich niedrig aufgelöste Bilder des betroffenen Bereichs zur Verifikation an das Alarmsystem senden.

Bluetooth-Präsenz und IR-Überwachung

Falls eine Person in einem Raum einen Sender mit Bluetooth-Funktion trägt, kann die Stärke des Signals eine Information zur Präsenz und Bewegung der Person im Raum liefern. Die Bluetooth-Funktion ist in ESP32-Mikrocomputern integriert. Der Mikrocomputer erkennt, ob sich eine Bluetooth-Gerät wie eine Smartwatch mit Vitalfunktion im Raum befindet. Als weitere Option bietet sich die Erkennung von Personen und deren Bewegung mit Hilfe einfacher IR-Sensoren an, die ohne Probleme in bestehende Mikrocomputer integriert werden können.

5.3 Persönlicher Ausblick

Wie im oberen Abschnitt 5.1 dargestellt, liefert das Ergebnis dieser Arbeit geeignete Methoden zur Erkennung von Geräuschen und Sprache im Kontext altersgerechter Assistenzsysteme mit Deep Learning zur Klassifizierung der Audiosignale. In diesem Zusammenhang wurden vielfältige Überlegungen hinsichtlich der Raumakustik, der Hardwareauswahl, der Signalverarbeitung als auch der Parametrierung und Klassifizierung mit CNN-Modellen angestellt. Insbesondere die Vorhersagegenauigkeit der CNN-Modelle bei Audioaufnahmen mit geringer Abtastrate haben die Erwartungen des Autors übertroffen.

Aus seiner Sicht sollten die Ergebnisse motivieren, die folgenden Themen in Form von Folgearbeiten zu vertiefen:

- **Anzahl Aufnahmequellen pro Raum**
  In der Arbeit wird aufgrund von Risiken einer Interferenz nur ein Mikrofon pro Raum eingesetzt. Es sollte untersucht werden, inwieweit ein Einsatz von mehreren Quellen ein Vorteil in der Erkennung in größeren Räumen bietet.

- **Optimierung der MEMS-Mikrofonaufnahme**
  Das Miniaturmikrofon liefert bei Abtastraten von 16 kHz gute Ergebnisse für eine Signalerkennung. Aus Sicht des Autors besteht noch Optimierungsbedarf bei der Speicherung und Übertragung der Daten an den zentralen Computer.

- **Generierung der Trainingsdaten**
  Die Generierung von 120 WAV-Dateien pro Klassen hat zu einer aussagefähigen Erkennung mit den CNN-Modellen geführt. Bei der Überprüfung einzelner Audiodateien ist aufgefallen, dass der Algorithmus zur Generierung bei einigen Geräuschen verbessert werden kann. Dies betrifft vor allem Audiosignale mit einem geringen Schwellwert, einer niedrigen Frequenz und einer geringen Bandbreite.

- **Parametrierung CNN-Modelle**
  Die ausgewählten CNN-Modelle liefern hohe Vorhersagegenauigkeiten einer korrekten Erkennung. Aus Sicht des Autors wird eine weitere Optimierung der CNN-Modelle mit Hilfe von Transfer Learning als auch Hyperparameter Tuning die Accuracy weiter erhöhen mit dem Effekt, dass besonders bei schwer zu klassifizierenden Signalen wie Schritte eine stabilere Vorhersage erzielt wird.

Literatur

[1] R. Wartala, *Praxiseinstieg Deep Learning, Mit Python, Caffee, Tensorflow und Spark eigene Deep-Learning-Anwendungen erstellen*, 1. Aufl. Heidelberg: d.verlag GmbH, 2018, S. IX, ISBN: 978-3-96009-054-0.

[2] DESTATIS-Statistisches-Bundesamt. „Fast 6 Millionen ältere Menschen leben allein, Pressemitteilung Nr. N 057 vom 29.09.2021.“ (2021, abgerufen 18.01.2024), Adresse: https://www.destatis.de/DE/Presse/Pressemitteilungen/2021/09/PD21_N057_12411.html.

[3] V. Moder-Siegmeth und K. Hofer, „Assistive Technologien für ältere Menschen: Nutzen für EndanwenderInnen und Herausforderungen im Einsatz,“ Jg. 53, Nr. 1, S. 57–72, 2013, abgerufen 18.01.2024. Adresse: https://nbn-resolving.org/urn:nbn:de:0168-ssoar-426517.

[4] REHADAT. „Lexikon zur beruflichen Teilhabe: Ambient Assisted Living (AAL).“ (abgerufen 09.02.2024), Adresse: https://www.rehadat.de/lexikon/Lex-Ambient-Assisted-Living-AAL/.

[5] T. Görne, *Mikrofone in Theorie und Praxis*, 8. Aufl. Aachen: Elektor Verlag, 2007, S. 42–49, ISBN: 978-3-89576-189-8.

[6] O. Curdt, „MEMS Mikrofone, Studiengang: AM7, Vorgelegt von: Henni Lotta Scheitz,“ *Veranstaltung: Tonseminar*, Adresse: https://curdt.home.hdm-stuttgart.de/PDF/MEMS-Mikrofone.pdf.

[7] S. Weinzierl, *Handbuch der Audiotechnik*, 1. Aufl. Berlin: Springer Verlag, 2008, S. 785, ISBN: 978-3-540-34300-4.

[8] S. Weinzierl, *Handbuch der Audiotechnik*, 1. Aufl. Berlin: Springer Verlag, 2008, S. 787–790, ISBN: 978-3-540-34300-4.

[9] S. Weinzierl, *Handbuch der Audiotechnik*, 1. Aufl. Berlin: Springer Verlag, 2008, S. 790–795, ISBN: 978-3-540-34300-4.

[10] S. S. Stevens, J. E. Volkmann und E. B. Newman, „A Scale for the Measurement of the Psychological Magnitude Pitch,“ *Journal of the Acoustical Society of America*, Jg. 8, S. 185–190, 1937. Adresse: https://api.semanticscholar.org/CorpusID:122448736.

[11] B. P. Bogert et al., „The Quefrency Alanysis of Time Series for Echoes: Cepstrum, Pseudo Autocovariance, Cross-Cepstrum and Saphe Cracking," 1963, Chapter 15, 209–243.

[12] H. Niemann, *Klassifikation von Mustern, Uni Erlangen PDF-Download*. 2003, S. 213. Adresse: https://www5.informatik.uni-erlangen.de/fileadmin/Persons/NiemannHeinrich/klassifikation-von-mustern/m00-www.pdf.

[13] A. Scheidler, „Low Level Descriptoren," *Seminar Musik TU Dortmund*, 2006. Adresse: https://www-ai.cs.tu-dortmund.de/LEHRE/SEMINARE/MUSIK/2006/scheidler_2006a.pdf.

[14] J. R. Stadermann, „Automatische Spracherkennung mit hybriden akustischen Modellen, Dissertation zur Erlangung des akademischen Grades eines Doktor-Ingenieurs," Diss., Technischen Universit¨at München, München, 2005, S. 8.

[15] E. Hamid, M. K. Molla und K. Hassan, „A Method for Voiced/Unvoiced Classification of Noisy Speech by Analyzing Time-Domain Features of Spectrogram Image," *Science Journal of Circuits, Systems and Signal Processing*, Jg. 6, S. 11, Okt. 2017. DOI: 10.11648/j.cssp.20170602.12.

[16] H. Ernst et al., *Grundkurs Informatik, Grundlagen und Konzepte für die erfolgreiche IT Praxis - Eine umfassende, praxisorientierte Einführung*, 7. Aufl. Berlin: Springer, 2020, S. 826, ISBN: 978-3-658-30330-3. DOI: 10.1007/978-3-658-30331-0.

[17] L. Wuttke. „Transfer Learning: Grundlagen und Einsatzgebiete." (2024, abgerufen 29.01.2024), Adresse: https://datasolut.com/was-ist-transfer-learning/.

[18] „Beyond Emotion AI, Analyse von 17 verschiedenen Emotionen und Gesichtsausdrücken." (abgerufen 09.04.2024), Adresse: https://beyond-emotion.de/.

[19] E. Castelli et al., „Habitat Telemonitoring System based on the Sound Surveillance," *1st International Conference on Information Communication Technologies in Health*, S. 141–146, 11. Juli 2003. Adresse: https://core.ac.uk/download/pdf/51945717.pdf.

[20] Y. Zigel et al., „A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound," *IEEE Transactions on Biomedical Engineering*, Jg. 56, Nr. 12, S. 2858–2867, 1. Dez. 2009, ISSN: 1558-2531. DOI: 10.1109/TBME.2009.2030171. Adresse: https://ieeexplore.ieee.org/document/5223652.

[21] E. E. Stone und S. M., „Fall Detection in Homes of Older Adults Using the Microsoft Kinect," *IEEE Journal of Biomedical and Health Informatics*, Jg. 19, Nr. 1, S. 290–301, 1. Jan. 2015, ISSN: 2168-2208. DOI: 10.1109/JBHI.2014.2312180. Adresse: https://ieeexplore.ieee.org/document/5223652.

[22] Wikipedia. „Kinect." (abgerufen 11.02.2024), Adresse: https://de.wikipedia.org/wiki/Kinect.

[23] M. I. Ramadani et al., „On-Device MFCC-CNN Voice Recognition System with ESP-32 and Web-Based Application," in *2023 15th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 2023, S. 161–166. DOI: 10.1109/ICITEE59582.2023.10317720. Adresse: https://ieeexplore.ieee.org/document/10317720.

[24] S. Weinzierl, *Handbuch der Audiotechnik*, 1. Aufl. Berlin: Springer Verlag, 2008, S. 182–183, ISBN: 978-3-540-34300-4.

[25] T. Ziemer, *Psychoakustische Schallfeldsynthese für Musik, Abb. 6.7 Kapitel 6.2*, 1. Aufl. Berlin: Springer Verlag, 2023, S. 173, ISBN: 978-3-031-26862-5.

[26] Y. Li, K. C. Ho und M. Popescu, „A Microphone Array System for Automatic Fall Detection," *IEEE Transactions on Biomedical Engineering*, Jg. 59, Nr. 5, S. 1291–1301, 2012. DOI: 10.1109/TBME.2012.2186449.

[27] Freepik. „Bild von macrovector auf Freepik." (abgerufen 09.02.2024), Adresse: https://de.freepik.com/vektoren-kostenlos/innenansicht-von-oben_2870920.htm%5C#query=grundriss%5C&position=38%5C&from_view=keyword%5C&track=sph%5C&uuid=4b932c92-5db0-4e17-856d-caa27cc28ec7.

[28] atomic14. „ESP32 Audio Input Using I2S and Internal ADC." (2020, abgerufen 09.02.2024), Adresse: https://www.youtube.com/watch?v=pPh3_ciEmzs.

[29] InvenSense. „INMP441, Omnidirectional Microphone with Bottom Port and I2 S Digital Output." Version Version 1.1. (2014, abgerufen 09.02.2024), Adresse: https://pdf1.alldatasheet.com/datasheet-pdf/download/1244625/ETC1/INMP441.html.

[30] Espressif-Systems. „ESP32-MINI-1, Datasheet." Version Version 1.3. (2023, abgerufen 09.02.2024), Adresse: https://www.espressif.com/sites/default/files/documentation/esp32-mini-1_datasheet_en.pdf.

[31] Medion. „LIFETAB E10604 TABLET, Technische Daten." (abgerufen 09.02.2024), Adresse: https://www.medion.com/de/shop/p/tablets-geraete-medion-lifetab-e10604-tablet-25-7-cm-10-1--fhd-display-android-8-1-32-gb-speicher-quad-core-prozessor-lte-inkl-multimode-case-mit-integrierter-tastatur--b-ware-30025831B#technicalData.

[32] MXL-Microphones. „MXL 990, Tech Specs." (abgerufen 09.02.2024), Adresse: https://mxlmics.com/products/mxl-990/.

[33] Music-Tribe-FZE. „Behringer UMC404HD, Audiophile 4x4, 24-Bit/192 kHz USB Audio/-MIDI Interface with Midas Mic Preamplifiers, Product Features." (abgerufen 09.02.2024), Adresse: https://www.behringer.com/product.html?modelCode=0805-AAT.

[34] NVIDIA. „Eingebettete Systeme mit Jetson, Produkte." (abgerufen 09.02.2024), Adresse: https://www.nvidia.com/de-de/autonomous-machines/embedded-systems/.

[35] ASUS. „Zenbook Flip 13 UX363, Tech Specs." (abgerufen 09.02.2024), Adresse: https://www.asus.com/laptops/for-home/zenbook/zenbook-flip-13-ux363/techspec/.

[36] Eric, *ESP32-INMP441-RECORDING, ESP32 INMP441 Tutorial - Part.4 Capturing audio from i2s mic to save WAV file (I2S interface)*, Version 1.0, 9. Juni 2020. Adresse: https://github.com/0015/ThatProject/blob/master/ESP32_MICROPHONE/ESP32_INMP441_RECORDING/ESP32_INMP441_RECORDING.ino.

[37] That-Project. „ESP32 INMP441 Tutorial - Part.4 Capturing audio from i2s mic to save WAV file.“ (2020, abgerufen 09.02.2024), Adresse: https://www.youtube.com/watch?v=qmruNKeIN-o.

[38] H. Gochkov, *Arduino-32 Libraries Webserver FSBrowser, FSWebServer - Example WebServer with FS backend for esp8266/esp32*, Version 2.0.0, 15. Apr. 2021. Adresse: https://github.com/espressif/arduino-esp32/blob/master/libraries/WebServer/examples/FSBrowser/FSBrowser.ino.

[39] Audiophile. „Hi-Q MP3 Voice Recorder, Features.“ (abgerufen 09.02.2024), Adresse: https://www.hiqrecorder.com/features/.

[40] Audacity. „Audacity, Professionelles Aufnehmen und Editieren.“ (abgerufen 09.02.2024), Adresse: https://www.audacity.de/.

[41] B. McMahan und D. Rao, „Listening to the World Improves Speech Command Recognition,“ *Proceedings of the AAAI Conference on Artificial Intelligence*, Jg. 32, Nr. 1, Apr. 2005. DOI: 10.1609/aaai.v32i1.11284. Adresse: https://ojs.aaai.org/index.php/AAAI/article/view/11284.

[42] K. J. Piczak, *ESC: Dataset for Environmental Sound Classification*, Version V2, 2015. DOI: 10.7910/DVN/YDEPUT. Adresse: https://doi.org/10.7910/DVN/YDEPUT.

[43] The-TensorFlow-Authors. „Simple audio recognition: Recognizing keywords.“ (2020, abgerufen 29.01.2024), Adresse: https://github.com/tensorflow/docs/blob/master/site/en/tutorials/audio/simple_audio.ipynb.

[44] Pytorch-Tutorials. „Speech Command Classification With Torchaudio.“ (2022, abgerufen 29.01.2024), Adresse: https://pytorch.org/tutorials/intermediate/speech_command_classification_with_torchaudio_tutorial.html.

[45] K. J. Piczak. „ESC: Dataset for Environmental Sound Classification.“ (2015, abgerufen 29.01.2024), Adresse: https://nbviewer.org/github/karoldvl/paper-2015-esc-dataset/blob/master/Notebook/ESC-Dataset-for-Environmental-Sound-Classification.ipynb.

[46] P. Warden, „Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition,“ *CoRR*, Jg. abs/1804.03209, 2018. arXiv: 1804.03209. Adresse: http://arxiv.org/abs/1804.03209.

[47] C. D. Wei Dai et al., „VERY DEEP CONVOLUTIONAL NEURAL NETWORKS FOR RAW WAVEFORMS,“ 2016. DOI: 10.48550/arXiv.1610.00087. arXiv: 1610.00087 [cs.SD]. Adresse: https://arxiv.org/pdf/1610.00087.pdf.

[48] K. J. Piczak, *ESC: Dataset for Environmental Sound Classification*, Version V2, 2015. DOI: 10.7910/DVN/YDEPUT. Adresse: https://doi.org/10.7910/DVN/YDEPUT.

[49] F. Pedregosa, G. Varoquaux, A. Gramfort et al., „Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, Jg. 12, S. 2825–2830, 2011.

[50] A. W. Ramadhan, A. Wijayanto und H. Oktavianto, „Implementation of Audio Event Recognition for The Elderly Home Support Using Convolutional Neural Networks," *2020 International Electronics Symposium (IES)*, S. 91–95, 2020. Adresse: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=%5C&arnumber=9231702.

[51] K. J. Piczak, „Email-Antwort Paper-2015-ESC-Dataset," 13.02.2024.

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit mit dem Titel

Methoden zur Erkennung von Geräuschen und Sprache im Kontext altersgerechter Assistenzsysteme - Deep Learning zur Klassifizierung von Audiosignalen im Wohnumfeld älterer Menschen

selbstständig und nur mit den angegebenen Hilfsmitteln verfasst habe. Alle Passagen, die ich wörtlich aus der Literatur oder aus anderen Quellen wie z. B. Internetseiten übernommen habe, habe ich deutlich als Zitat mit Angabe der Quelle kenntlich gemacht.

Kai-Michael Wolters

Glückstadt, 15. April 2024