AI-powered Emergency Keyword Detection for Autonomous Vehicles

Hovannes Kulhandjian†, Brandon Poorman†, Javier Gutierrez†, and Michel Kulhandjian‡

†Department of Electrical and Computer Engineering, California State University, Fresno, Fresno, CA 93740, U.S.A.
E-mail: {hkulhandjian, brandonpoorman, jguti}@mail.fresnostate.edu
‡Department of Electrical and Computer Engineering, Rice University, Houston, TX, 77005, U.S.A.
E-mail: michel.kulhandjian@rice.edu

***Abstract*—Communication systems implemented within vehicles primarily focus on infotainment applications and often lack concrete specifications designed to ensure the safety of both drivers and passengers during emergencies. In this work, we propose and implement an in-vehicle emergency threat keywords detection system using machine learning and audio signal processing to classify potential emergency keywords based on audio waveforms while reducing noise and unprivileged user voice commands from creating a false positive. This is done by taking speech commands from users via a microphone array within the vehicle that is then preprocessed, and features are extracted from Mel spectrogram images. These mel spectrogram images are classified using a convolutional neural network (CNN) that has been previously trained to classify emergency speech keywords. Experimental results reveal a validation accuracy of about 90% is achieved in accurately detecting and classifying the threat keywords. The proposed system can be used in an emergency protocol within the autonomous vehicle by pulling over safely to prevent harm or call first responders.**

***Index Terms*—Autonomous vehicle emergency system, machine learning, audio signal processing.**

I. Introduction

In autonomous vehicles, deep neural networks have been primarily focused on computer vision applications where the vehicle labels and tracks objects in its environment, makes a decision on what that object is and then executes an action based on that object. Different neural networks can be repurposed for audio applications, specifically inside the vehicle. Audio classification using machine learning algorithms is much more difficult to implement due to data variability in sound waveforms, the human voice being highly non-stationary, the unavailability to label images, and a significant amount of digital audio processing to make the data usable for the neural network. Due to the variance in speech patterns, the addition of noise, and numerous other factors that can distort audio waveforms, advanced signal processing techniques need to be utilized such as beamforming and sound source localization (SSL) in order to prevent misclassification or false positives of emergency keyword detection and classification. Additionally, autonomous vehicles require precise controls to execute emergency procedures accurately, ensuring safety for passengers, pedestrians, and objects on the road.

Related Works: In recent years, audio-based classification has been explored using machine learning. The majority of those classification techniques rely on the mel-frequency cepstral coefficients (MFCCs), spectrograms, log-mel spectrum, mel filter bank, and many other types of audio processing techniques. Based on these sound classification schemes, the proposed system is capable of predicting emergency threat keywords in an autonomous vehicle and responding appropriately by preventing harm or injury to passengers, bystanders, and/or infrastructure.

Lord *et al.* [1], suggest a method for vehicular threat detection through audio signal analysis. This method sends alerts to both the pedestrian's wearable device and the vehicle's driver in case of a potential collision.

Santos *et al.* [2], develop an audio-based system that detects violence in the car. One of the challenges of speech processing involves identifying violence in speech among the audio, music, and ambient noise. Santos *et al.* analyzed the accuracy of several different deep learning architectures.

Moreover, Purwins *et al.* [3], discusses the features, data requirements, and computational complexity for different deep learning models in audio signal processing applications. Using this information extracted from a large dataset and feeding it into a neural network can classify sounds in various applications such as human speech, music, and environmental sounds. Purwins *et al.* argue in favor of utilizing distinct signal processing information across various applications and deep learning models, prompting a consideration of the most suitable approach for each scenario.

Hao *et al.* [4], design a system using multiple microphone arrays on a Raspberry Pi for the real-time operation to train a convolutional neural network (CNN) model using noisy speech recordings collected from different rooms and inference on an unseen room to improve SSL. In noisy rooms or environments such as in a vehicle, traditional speech quality suffers immensely resulting in poor performance and/or improper classification of sounds.

Almaadeed *et al.* [5], combine time-domain, frequency-domain, and joint time-frequency features extracted from a specific category of quadratic time-frequency distributions to perform event detection on roads using audio analysis and processing.

Jonnadula *et al.* [6], study different methods used in identifying the emergency vehicle present on the road. Their main emphasis is on artificial neural networks (ANNs) without delving into the potential of convolutional neural networks, which have been proven to offer significantly enhanced accuracy in detection and classification tasks.

To the best of our knowledge, the detection of emergency keyword threats based on audio signals has not been addressed sufficiently so far specifically for autonomous vehicles. One of the use cases that we will discuss in this

979-8-3503-7099-7/24/$31.00 ©2024 IEEE

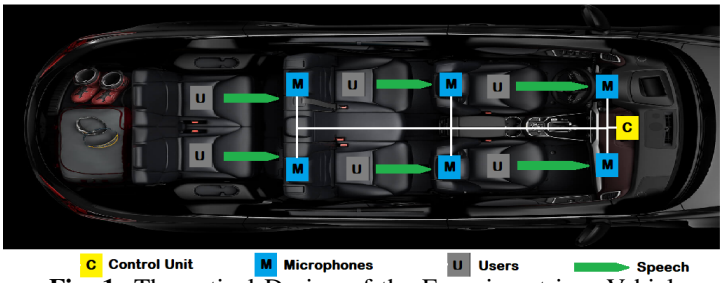

Fig. 1: Theoretical Design of the Experiment in a Vehicle.

manuscript is the emergency situation when a person inside the vehicle has to make an emergency request, the vehicle could recognize the command and take appropriate measures to perform a full stop of the vehicle safely. Additionally, improving overall sound detection and localization inside a vehicle can be ported to improve and enhance the safety of hands-free driving using speech rather than touch on the infotainment system. The significance of this work will improve the overall safety of passengers and drivers alike with a hands-free system to provide redundancies when it comes to autonomous vehicles. In the instance, that a driver becomes incapacitated at the wheel or is unable to stop the vehicle using touch, speech commands are another alternative in the event of an emergency. Additionally, this system may enhance machine learning research and applications in speech and audio processing where less effort has been conducted as compared to computer vision systems.

The rest of the paper is organized as follows. In Section II, we present the design methodology. In Section III, we discuss the digital signal processing performed on the audio signals. In Section IV, we discuss the emergency keywords detection experimental results and analysis before presenting our conclusion in Section V.

II. Design Methodology

The methodology stated below addresses the problems inherent in audio emergency detection in autonomous vehicles. First, the hardware implementation needs to be set up with multiple microphone arrays spread out throughout the inside cabin of the vehicle, as shown in Fig. 1. These microphone arrays are used for spatial selectivity of sound. Sounds at different angles experience constructive interference and destructive interference at other angles. Using beamforming with these microphone arrays allows for the spatial selectivity of sound to distinguish sounds from a particular direction and focus on that sound. In a vehicle environment with substantial sound signal interference, the task of segregating and prioritizing audio waveforms becomes crucial. This prioritization is aimed at isolating the desired audio waveforms, minimizing noise, and preparing the digital audio data for utilization in training and classification models. To properly classify the data from the vehicle for accurate emergency detection, a pre-trained model is needed to distinguish emergency detection sounds from other sounds. This trained model needs to contain many different types of audio sounds and be classified accurately. There are a number of different audio training sets available, such as Youtube-8M. The Youtube-8M dataset contains $237,000$ human-verified labels with 1000 different classes of both sounds and images. Other datasets can be used, but these datasets are used primarily for the verification of sounds and initial training. A large portion of the training set needs to be developed in-house by many different types of people in order to improve accuracy and have a proper and fair representation of the data to train and validate.

Next, the data from the microphone arrays are processed using beamforming and noise cancellation to produce a suitable waveform to be processed into an MFCC for the neural network. The extracted features from the microphone data will be fed into the pre-trained neural network to classify the data to its appropriate class. If it is an emergency class, then the autonomous vehicle will perform the emergency procedure. If it is not an emergency class, then the vehicle will not perform the emergency procedure.

Based on the research conducted, there are no datasets that are specific to the speech commands such as “Stop the car!” and “Pull over!” for training our proposed model. Therefore, our experimental demonstration will be based on some real constraints such as hardware configurations and audio dataset of two different male personals.

III. Digital Signal Processing

Through audio processing techniques, information can be extracted from raw speech waveforms such as MFCCs, spectrograms, the log-mel spectrum, and the mel-filter bank. Speech signals from the time domain need to be preprocessed before being used for feature extraction. Once processed they are placed in a feature vector that will be used as an input into the neural network, as shown in Fig. 2. These features can be used to train and classify sounds to perform an action inside the vehicle, such as in an emergency situation.

A. Beamforming

In order to process time-domain signals into the necessary information, there needs to be some digital signal processing

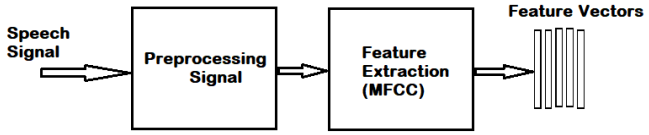

Fig. 2: Audio processing from raw signal into features.

done before they can be used as input into the neural network. Beamforming is a significant digital signal processing technique that can make a particular incoming signal on a specific channel more distinguishable over other signals that are captured by the microphone array. Specifically, minimum variance distortionless response (MVDR) beamforming stood out as one of the more useful spatial beamforming techniques for our specific application. MVDR beamforming minimizes array output power and maintains a distortionless response toward the desired signal. A weighting vector guides a signal toward a target direction to filter out unwanted external noise, music, or other speech signals. MVDR was selected due to the known angular direction of users' speech input within the vehicle. The beamforming algorithm prioritizes and enhances the channel from the direction of known users while reducing noise from other channels.

B. Noise Cancellation

Noise cancellation is performed with two major components: the estimation of the noise power spectrum and the estimation of speech. Because speech can be estimated as noise in certain environments, especially reverberant environments, we don't want those important speech signals suppressed. The speech estimator is performed using an optimally modified log spectral amplitude (OM-LSA) developed by Cohen [7]. The noise estimator is performed using a minimum controlled recursive averaging (MCRA) function [7]. Noise estimators for speech enhancement are usually performed by estimating the noise power spectrum over the parts of a signal that are known to have noise and do not contain speech or by looking at specific frequencies in a signal that are known to have noise. However, this severely limits the ability of the noise estimator to perform in a more dynamic environment, where there may not be speech and where noise can occur independently and randomly. Probability and statistical models perform much better when noise can occur at any point throughout the signal. Figure 3 shows the linear time-invariant system of the noise cancellation process from noisy speech into clean speech.

C. MFCCs Signal Processing

The time-domain signals need to be framed, windowed and converted to the frequency spectrum to produce spectrograms. These spectrograms can be further processed to obtain information to be used for feature extraction, such as the MFCCs, the log-mel spectrum, and the mel-filter bank. We first process the audio by padding the signals in order to make them all the same length. Then the audio signals

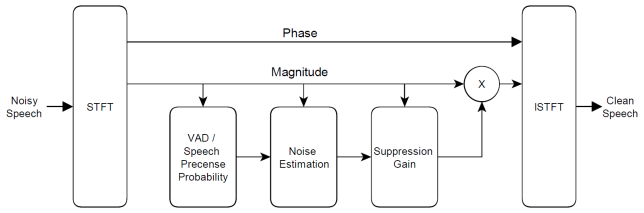

Fig. 3: Noise Cancellation Block Diagram.

are normalized by making the volume of the audio files to a standard set level for all the audio signals utilized in our training and data sets. In order to boost the signal's high-frequency components and leave out the low-frequency components in their original states we utilize a finite impulse response (FIR) as follows,

$$P(z) = 1 - 0.97z + z^2. \tag{1}$$

Framing is done by converting a single audio array into successive smaller audio arrays. We used Hamming window,

$$w(t) = 0.5 + 0.5\cos(2\pi t/T), \tag{2}$$

to smooth out audio frames. We perform a short-time Fourier transform and then convert each frequency to mel scale. More specifically, the mel scale is used to mimic the non-linear human ear perception of sound by being more discriminative at lower frequencies and less discriminative at higher frequencies. The conversion process from frequency to mel scale is done as follows,

$$mel = 1127\log(1.0 + f/700), \tag{3}$$

where the constants 700 Hz and 1127 Hz are defined for the mel low and high frequencies. By this process, we actually increase the power of spectral components of frequencies within the human speech frequency range and reduce spectral components in sound signals outside the human speech frequency range. For the neural network model, we use the MFCC representation of features. The MFCCs represent the short-term power spectrum of a sound based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.

MFCCs are useful for extracting features in speech recognition systems. The combination of these different signal processing techniques can be used as useful information to be used in a CNN. Additional considerations need to be taken into account when trying to process these audio/speech waveforms, especially in noisy environments such as in a car when music is playing, outside noise around the vehicle, and with children talking.

IV. Experimental Results

The goal of the subsequent sections is to show the proper progression of a raw waveform with significant undesirable signals into a signal that can be properly classified without significant noise, music, and other human speech present in the signal. Conditions used for this included a reverberating environment similar to the cabin of a vehicle with music at a much higher amplitude than the desired human speech.

A. Recording

A one-second audio waveform was recorded using the Respeaker USB Mic Array v2.0 connected to the Jetson Nano using a Python script. The parameters utilized included a $16,000$ sampling rate and 4 separate channels. Each channel is one of the four microphone arrays on the Respeaker. A higher sampling rate would have been ideal, but due to the limitations of the hardware, it can only support up to $16,000$ samples per second. Figure 4 is a typical waveform that

would be analyzed inside the cabin of a vehicle with multiple sound sources. Three separate sound sources were producing sound with two of the sound sources being music/noise and the other being one person talking with another person. The third sound source originates from a human user that we want to focus on saying the speech command "Stop". The signal in Fig. 4 is most present between approximately 0.5 and 0.80 seconds. The music being played maintained a consistent amplitude, as did the undesired background human speech. Upon direct auditory perception of the unprocessed audio waveform, the intended human speech lacks clarity and struggles to be discerned by the human ear, while the music stands out prominently, along with the undesirable human speech.

B. MFCCs Results

Figures 5 and 6 portray the MFCCs of the raw and beamformed waveforms, respectively. Due to significant power levels at many different frequencies throughout the time range of the signal in Fig. 5, the model will not be able to distinguish the human speech that is required to classify the signal properly.

The MVDR Python module that we have implemented focuses in a specific direction, specifically on the location of the desired human speech command. The chosen direction for this particular implementation was $225°$. The direction of arrival (DOA) for each microphone had to be calculated and validated through a separate program that uses DOA estimation to calculate the particular direction that the Respeaker recognizes in order to ensure that MVDR beamforming is performing with accurately chosen parameters. For reference, each microphone on the Respeaker is separated by $90°$ degrees with MIC1 located at $45°$, MIC2 located at $135°$, MIC3 located at $225°$, and MIC4 located at $315°$.

C. Beamforming

There is an improvement in the overall spectral components of the human speech command desired when analyzing Fig. 6. More power is present at frequencies from 512 Hz to 2048 Hz (the human speech frequencies within the mel scale) in the time window of 0.5 seconds to 0.80 seconds than in the signal from Fig. 6. Additionally, most of the power present in the lower and higher frequencies below and above the mel scale has been reduced, respectively.

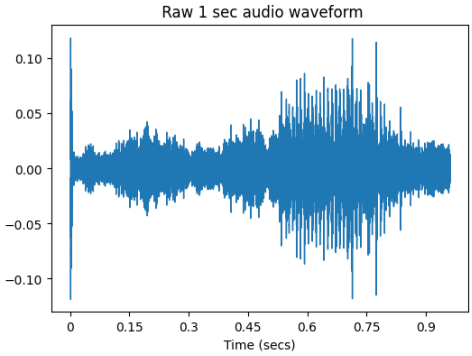

Fig. 4: Raw One Second Recorded Waveform.

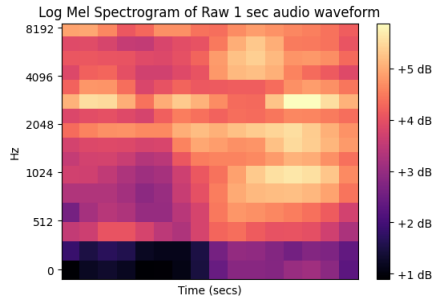

Fig. 5: Logarithmic mel frequency spectrogram of the raw one recorded second waveform.

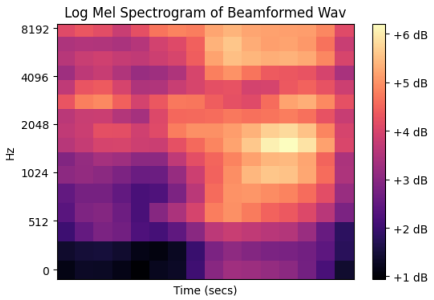

Fig. 6: Logarithmic mel frequency spectrogram of the beamformed waveform.

The difference in audio quality of the desired human speech command is more apparent when listening to the audio signal with the human ear. By listening to the beamformed audio signal, which is sampled and saved we observe a significantly improved audio signal with the human speech signal coming through clearer than before but the music and other noise in the different focused directions is reduced. However, the music is still within the signal because of the inherent near proximity of the microphones on the Respeaker Mic Array. Fully eliminating all power from undesired sound sources without physical components attached to the microphone array is challenging. Nonetheless, beamforming notably enhanced audio in specific directions compared to the original recording. Additionally, digital signal processing can further refine the extraction of desired audible signals.

D. Noise Cancellation and Representation of Feature Vectors

After beamforming, noise cancellation is utilized and the MFCC is applied as shown in Fig. 7. The noise cancellation appears to improve the audio at a degree greater than the beamforming at first analysis. However, without the beamforming, the noise-canceled signal would have a significant amount of powerful spectral components that affect the noise estimation and degrade the overall amount of meaningful speech that may or not be estimated as noise at particular frequencies.

The MVDR reduces noise in specific channels, while noise cancellation enhances the overall waveform quality. Despite extensive testing, complete elimination of high-amplitude

music and noise wasn't achieved with MVDR beamforming and noise cancellation combined.

E. CNN Training Model

Using the "wav" files stored in folders of each word the MFCC's are computed and stored in their respective folders. The next step involves subjecting these input MFCC images to a feature extraction process using a convolutional network.

Figure 8 shows the training and validation accuracy results of the classification deep neural network. The results show a validation accuracy (solid line) of about 90% is achieved with the given 30 epochs.

F. Real Time Testing

Using the Nvidia Jetson Nano Board and the Mic Array, the sound was captured at 1 second intervals. This is done in a Python program with the properly configured settings for the Respeaker Mic array. A while loop command first prints "record" to the command prompt to alert the user that it is about to record. This while loop also has a record and a predict function. The record function is what handles the recording which is saved into an "output.wav" file. Once the sound is recorded then the prediction process can take the next step. MFCCs are generated from the sound clip recorded, these are used to predict based on the trained model. MFCCs are generated from the sound clip recorded which are used to predict based on the trained model uploaded to the Nvidia Jetson Nano board. When the program has finished the user gets a list of predicted words and corresponding percentages. The program has the greatest confidence in displaying the word with the highest percentage as the predicted word. This is followed by an action that should be taken. The start time and the execution time are also printed to aid debugging and show the amount of time it would take to do all of the recording, sound processing, beamforming, noise cancellation, and then prediction based on the model. It is to be noted that the overall time range was not consistent in testing the continuous real-time and occurred between the range of 3.2 seconds and 5.7 seconds from observation. The most common occurrence of the average time of completion of the system was approximately 4.1 seconds. Moreover, the one second of these times was when the Respeaker Mic Array was recording for one second. "Stop" and "Off" were chosen as the speech commands that would initiate the interruption for "Emergency, pulling over". All other words are classified as non-emergency and the system keeps repeating. The real-time testing demonstrated fairly accurate results, as we anticipated from our developed model.

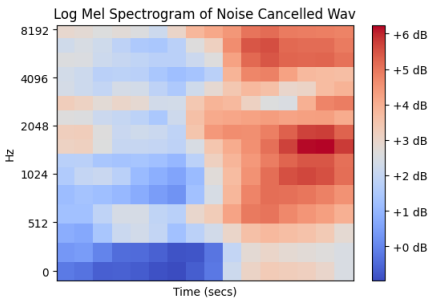

Fig. 7: Logarithmic mel frequency spectrogram of the beamformed noise canceled waveform.

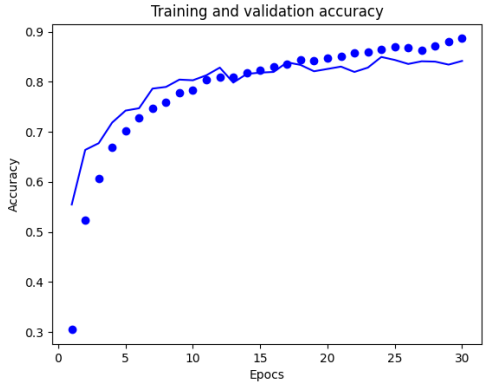

Fig. 8: Training (dotted line) and validation (solid line) accuracy results of the CNN model.

V. Conclusion

In this paper, we introduced and implemented an in-vehicle system for detecting emergency threat keywords using machine learning and audio signal processing. This system is designed to classify potential emergency keywords based on audio waveforms while minimizing the impact of noise and unintended user voice commands, reducing the likelihood of false positives. We gathered and processed speech commands from vehicle users via a built-in microphone array, extracting key features from Mel spectrogram images. These images were classified using a specialized pre-trained CNN for emergency speech keywords. Results show over 90% average validation accuracy, ensuring precise detection and classification of threat terms. This system seamlessly integrates into an autonomous vehicle's emergency protocol, enabling safe stops to prevent harm or prompt swift response from first responders.

References

[1] R. T. Lord, R. W. Lord, N. P. Myhrvold, C. T. Tegreene, R. A. Hyde, L. L. Wood, M. Y. Ishikawa, V. Y. Wood, C. Whitmer, P. Bahl *et al.*, "Vehicular threat detection based on audio signals," Aug. 11 2015, uS Patent 9,107,012.

[2] F. Santos, D. Durães, F. S. Marcondes, N. Hammerschmidt, S. Lange, J. Machado, and P. Novais, "In-car violence detection based on the audio signal," in *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 2021, pp. 437–445.

[3] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

[4] Y. Hao, A. Küçük, A. Ganguly, and I. M. Panahi, "Spectral flux-based convolutional neural network architecture for speech source localization and its real-time implementation," *IEEE Access*, vol. 8, pp. 197 047–197 058, 2020.

[5] N. Almaadeed, M. Asim, S. Al-Maadeed, A. Bouridane, and A. Beghdadi, "Automatic detection and classification of audio events for road surveillance applications," *Sensors*, vol. 18, no. 6, p. 1858, 2018.

[6] E. P. Jonnadula and P. M. Khilar, "Comparison of various techniques for emergency vehicle detection using audio processing," 2019.

[7] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal processing*, vol. 81, no. 11, pp. 2403–2418, 2001.