IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 7, 2021, accepted November 24, 2021, date of publication December 1, 2021,
date of current version December 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3132061

C-AVDI: Compressive Measurement-Based Acoustic Vehicle Detection and Identification

BILLY DAWTON[1], (Graduate Student Member, IEEE), SHIGEMI ISHIDA[2], (Member, IEEE), AND YUTAKA ARAKAWA[1], (Member, IEEE)
[1]ISEE, Kyushu University, Fukuoka 819-0395, Japan
[2]Department of Media Architecture, School of Systems Information Science, Future University Hakodate, Hokkaido 041-8655, Japan

Corresponding author: Billy Dawton (bdawton@f.ait.kyushu-u.ac.jp)

This work was supported in part by JSPS KAKENHI under Grant JP19KK0257, and in part by the Cooperative Research Project Program of the Research Institute of Electrical Communication (RIEC), Tohoku University.

ABSTRACT As society grows ever more interconnected, the need for sophisticated signal processing and data analysis techniques becomes increasingly apparent. This is particularly true in the field of intelligent transportation systems (ITSs), where various sensing applications generate data at an exponential rate. In this paper, we present C-AVDI, a compressive measurement-based acoustic vehicle detection and identification architecture capable of extracting information from vehicle audio signals while sampling at sub-Nyquist rates. In addition, we further reduce the overall complexity by performing any necessary signal filtering during the acquisition process, removing the need for a separate filtering stage in the system's front-end. Our results obtained from data collected under a range of weather conditions present an accuracy of 80% with a back-end analog-to-digital converter (ADC) sample rate of 3 kHz, with initial results from a microcontroller (MCU) implementation of our proposed system presenting an accuracy of 72%.

INDEX TERMS Acoustic signal detection, compressive sensing (CS), intelligent transportation systems (ITS), vehicle detection.

I. INTRODUCTION

In recent years, we have seen a rise in the development and adoption of intelligent transportation systems (ITS) technology. Key applications such as traffic flow control, navigation systems, and road safety management are growing ever more sophisticated and gaining increasingly widespread use. These improvements, however, come at a cost: the increasing amount of data created, processed, and used requires expensive, power-intensive hardware to be stored, accessed, and leveraged effectively. Particularly important in ITSs is the vehicle detection and identification (VDI) process, a key functionality underpinning a significant number of applications such as traffic flow and congestion management, electronic toll collection (ETC), and transportation infrastructure monitoring. Lowering the cost and computational requirements associated with the VDI process will have a direct effect on the overall cost and complexity of many ITS applications.

Various low-cost, low-complexity VDI techniques have been proposed, with acoustic vehicle detection and identification (AVDI) in particular being the subject of wide-ranging research [1]–[15]. The low price and simple installation process of AVDI systems make them an attractive alternative to other more expensive and difficult-to-install VDI systems, such as video camera-, radar-, or induction loop coil-based systems. While the installation costs associated with AVDI systems are generally low, the subsequent analysis and leveraging of data is often relatively costly in terms of computational requirements, making the use of such systems in power-critical applications difficult. In current AVDI systems, the computational costs are incurred due to processes occurring in either the acquisition and preprocessing stage as in [2] and [3] (successive discrete Fourier transforms (DFTs) or discrete wavelet transforms (DWTs)), or the classification stage as in [4] and in [5] (use of multi-layer perceptron (MLP) and artificial neural network (ANN) respectively). In a bid to reduce the computational costs associated with AVDI systems, we proposed in our previous work [1] an initial attempt at creating a compressive measurement-based acoustic vehicle sensing architecture, capable of detecting

The associate editor coordinating the review of this manuscript and approving it for publication was Shaohua Wan.

and identifying vehicles while sampling at sub-Nyquist rates. By leveraging certain key properties of compressive sensing (CS), a technique first presented in [16] and [17], the system was able to detect and identify vehicles with an accuracy of 83% for a back-end sample rate of 3 kHz.

The goal of this paper is to present a low-cost, low-complexity alternative to existing AVDI sensing systems that can implemented on a micro-controller (MCU) and whose performance is comparable to that of currently available systems. This is achieved first and foremost by reducing the sample rate at which the vehicle sounds are acquired. Indeed, the biggest limitation of MCUs is the amount of available volatile and nonvolatile memory, and while this is not a problem when working with short signals such as in [18], it makes it impossible to use MCUs in AVDI applications where the signal length is typically on the order of a few seconds. By reducing the number of samples, we lower the memory requirements required for both the implementation of the classifier (nonvolatile memory), and the feature extraction process (volatile memory).

We seek to expand and build upon the results obtained by our proof-of-concept system in [1] in a number of ways.

First, we remove the front-end filtering stage and instead filter the input signal directly during the acquisition process using a spectrally tailored bipolar pseudo-random sequence. This reduces the number of components in the system and allows the filtering parameters to be adjusted by modifying the spectrum of the sequence rather than the hardware. Second, we test the system under adverse weather conditions, and observe their effects on the detection and identification performance. Third, we implement the feature extraction and classification processes on an MCU, demonstrating the reduction in both cost and computational complexity presented by our newly proposed system.

Our main contributions are as follows:

- We propose a compressive measurement-based AVDI (C-AVDI) architecture capable of operating at sub-Nyquist rates. Leveraging the inherent structure of vehicle sound signals enables us to reduce the number of samples required to detect and identify passing vehicles by a factor of 16 while maintaining an accuracy comparable to those of existing systems.
- We develop a method to simultaneously sample and filter incoming signals using spectrally shaped bipolar sequences generated from a pair of Markov chains. The shape of these sequences' spectra is controlled by varying the Markov chain length and transition probability. In our paper, we use the term *filtering* to refer to both the attenuation and amplification of frequencies present in a signal.
- Demonstrate the viability of C-AVDI as a low-cost, low-complexity sensing architecture by benchmarking our proposed system against our previous work and prototyping an initial MCU-based implementation of the system's feature extraction and classification processes.

Finally, while the purpose of our paper is to design a C-AVDI system for ITS applications, the underlying concepts can be applied to a wide range of sensing and machine learning (ML) applications. As highlighted in [19] and [20], if we are to ensure the continued sustainability and accessibility of ITS, as well as that of ML and artificial intelligence (AI) in general, then it is necessary to reduce the financial and environmental impact of these technologies. We believe that the techniques outlined in this paper can help contribute towards this goal.

The remainder of this paper is structured as follows: we examine the existing literature in Section II, describe the theory and properties of CS in Section III, and present our proposed system in Section IV. The system evaluation results are shown in Section V and discussed in Section VI.

II. RELATED WORK

There is a wide variety of existing research that uses supervised learning techniques to detect and identify vehicles using features extracted from their acoustic signatures.

The frequency-domain features extracted from vehicle audio signals are used with a support vector machine (SVM) classifier to identify and classify vehicles and their associated parameters in [6], and used to detect vehicles for the purpose of collision avoidance in non-line-of-sight situations in [7].

Mel-frequency cepstral coefficients (MFCCs) are used in conjunction with ML or deep learning (DL) as features in a number of existing AVDI systems: in [8] they are used with a modified MLP, in [5] they are extracted from a specific high energy audio region and used with an ANN and k-nearest neighbors (KNN) classifier, and in [9] they are used in a feature set containing the pitch class profile (PCP) and short-term energy (STE) of vehicle audio signals in a hybrid convolutional neural network (CNN) containing a long short-term memory (LSTM) layer.

In [4], Göksu presents a system capable of analyzing the acoustic signatures of vehicles independently of any changes in engine sound. Using wavelet packet decomposition (WPD) and an MLP classifier, the system obtains engine speed-independent features from the acoustic signals of passing vehicles. On the other hand, the system in [10] identifies different vehicles based on the sound of their engines using modulated per-channel energy normalization (Mod-PCEN) features in tandem with a Siamese neural network (SNN).

All of these aforementioned existing AVDI systems share a common, or very similar, goal and basic approach, but differ in their implementation, applications, and performance. While these systems generally all present good performance metrics, the use of computationally intensive input signal processing or a complex supervised learning method offsets the improvements in system cost, complexity, efficiency and flexibility.

We have proposed several acoustic vehicle detection systems in our previous work.

In [11], we present a sequential acoustic vehicle detector (SAVeD) that operates by fitting S-curves generated using generalized cross-correlation (GCC) to points drawn on a sound map using an estimation method based on random sample consensus (RANSAC). This system presents an F-measure of 83%. An initial attempt at expanding this method to include multilane detection was proposed in [12].

In [2], we design a stereo microphone-based AVDI system (SMBAS) capable of identifying passing vehicles through the use of features extracted from the frequency-domain representation of the vehicles' sound signatures using successive short-time Fourier transforms (STFTs). The signals obtained by a pair of microphones are time-aligned and combined, creating an emphasized sound signal from which features are drawn and used in an SVM classifier. The time alignment and combination process improves the system estimation accuracy, particularly when faced with simultaneously and successively passing vehicles, resulting in a system accuracy of 95%.

Both of these systems present good performance metrics, but again the computational cost associated with the two of them makes low-power, embedded applications difficult.

In [13], we propose an ultra-low power vehicle detector (ULP-VD) capable of detecting passing vehicles with minimal computational cost using logistic regression (LR). This system, however, is only able to detect the presence of passing vehicles and requires an additional stage to identify them.

Most recently, in [1], we proposed an initial attempt at creating a CS-based AVDI system capable of detecting and identifying vehicles while sampling at sub-Nyquist rates. By combining a more traditional sub-Nyquist sampling architecture with a tailored analog front-end filtering section, the system is able to identify and detect vehicles with an accuracy of 83%, and a back-end sampling rate of 3 kHz using a random forest (RF) classifier. The front-end filtering section is an integral part of the system; however, its implementation as a separate stage requires the use of additional components, increasing deployment and implementation costs.

The main characteristics of these AVDI systems are summarized in Table 1. To the best of our knowledge, there is no current AVDI system capable of performing both detection and identification on an MCU.

Sub-Nyquist signal processing using the measurements obtained during the CS process has been explored as an alternative to traditional digital signal processing in a range of existing work.

The authors of [21] present a methodology in which signal processing is performed directly on the compressive measurements obtained during the CS process. The paper demonstrates that filtering, detection and classification can be performed directly on the compressive measurements without recovering the signal beforehand. Similarly, the authors of [22] propose a multiband compressive signal processing architecture in which information acquired at sub-Nyquist rates from different frequency bands is used in a range of applications without prior signal reconstruction.

TABLE 1. Summary of related AVDI work.

System	Can Detect	Can Identify	Run on MCU	Input Signal Processing	Detection /Identification
[1]	✓	✓		CS	RF
[2]	✓	✓		STFT	SVM
[4]	✓	✓		WPD	MLP
[5]	✓	✓		MFCC	KNN /ANN
[6]	✓	✓		DFT	SVM
[7]	✓			STFT	SVM
[8]	✓	✓		MFCC	MLP
[9]	✓	✓		MFCC /PCP/STE	CNN /LSTM
[10]	✓	✓		Mod-PCEN	SNN
[11]	✓			GCC	RANSAC
[12]	✓			GCC	RANSAC
[13]	✓		✓	DWT	LR

There is also existing research that focuses on the creation of sampling strategies used to optimize the performance of CS-based systems.

Both [23] and [24] propose a method for shaping the spectra of the pseudorandom bipolar sequences used in various CS architectures to improve signal reconstruction performance. The first proposes a Markov chain-based method inspired by run-length limited (RLL) sequences to both shape the spectrum of the bipolar sequence and limit the rate at which it switches polarity. This paper lays the groundwork for subsequent research, but does not fully explore the filtering effects of such a sequence in the context of CS-based sensing, instead focusing on optimizing signal reconstruction. The second proposes a method based on convex relaxation to create a binary sequence whose spectrum resembles that of a notch filter. While this method is able to produce sequences with well-defined spectra, the complexity associated with this method makes it unsuitable for low-power applications.

III. COMPRESSIVE SENSING

A. BASICS OF COMPRESSIVE SENSING

First presented in [16] and [17], CS is a method to efficiently sample sparse or compressible signals (a signal is called sparse if it contains only a few nonzero components compared to its total length, and is called compressible if it contains a large number of nonzero components, but only a few of significant magnitudes), enabling us to directly acquire the information of interest in a given signal at sub-Nyquist rates rather than sample the signal at the Nyquist rate and subsequently discard unwanted information.

Let us begin by defining a sparse signal $x \in \mathbb{R}^N$ of length T_s whose highest frequency component is $W/2$ Hz and with $N = WT_s$. This signal can be expressed as a combination of discrete coefficients $\alpha \in \mathbb{C}^N$ and vectors ψ_n that form the columns of an orthonormal basis matrix $\Psi \in \mathbb{C}^{N \times N}$ for a given time window:

$$x(t) = \sum_{n=1}^{N} \alpha_n \psi_n(t)\ ,\ t \in [0, T_s) \tag{1}$$

with the coefficients computed as $\alpha_n = \langle x, \psi_n \rangle$.

More often than not, $x(t)$ itself is not sparse and instead has a sparse representation in Ψ. Given prior knowledge of Ψ, we need only obtain the information contained in the sparse coefficient vector α to be able to reconstruct the original signal. This information is obtained by drawing a set of compressive samples $y \in \mathbb{R}^M$ from the original signal, where $M \ll N$.

The CS acquisition process can be described mathematically as:

$$y = \Phi x \tag{2}$$

where $\Phi \in \mathbb{R}^{M\times N}$ represents the consecutive sampling and operations performed on x.

Additionally, we define the reconstruction matrix as: $\Theta \in \mathbb{C}^{M\times N} = \Phi\Psi$.

From (1) and (2):

$$\begin{aligned} y &= \Phi\Psi\alpha && (3)\\ &= \Theta\alpha && (4) \end{aligned}$$

The sparse coefficient vector and thus the original signal can be recovered from the compressive measurements by solving the l_1 minimization problem:

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_1 \quad \text{such that} \quad y = \Theta\alpha \tag{5}$$

where $\hat{\alpha}$ is the estimated coefficient vector and $\|\alpha\|_1$ is the l_1 norm (sum of the absolute vector values) of α.

For complete recovery of x, it is necessary to design Θ in such a way that it satisfies the incoherence and RIP (restricted isometry property) conditions outlined in [25], thereby ensuring that all the relevant information present in x is preserved in the measurements y. In [26], it is stated that a matrix such as Θ in (4) satisfies these conditions with overwhelming probability if Ψ is an orthonormal basis, Φ is drawn randomly from a suitable distribution such as the Gaussian distribution, and if the number of measurements M is higher than a lower bound defined as:

$$M \geq SK \log \frac{N}{K} \tag{6}$$

where K is the sparsity level of the input signal and S is a positive constant [27].

B. COMPRESSIVE MEASUREMENTS AS DIMENSIONALITY REDUCTION

In our proposed system, we are not looking to reconstruct the original input signal x. The computational complexity associated with the CS procedure occurs predominantly during the reconstruction process, and bypassing this procedure by detecting and identifying passing vehicles based on information extracted directly from the compressive measurements y enables us to significantly reduce the computational complexity of our system compared to more traditional CS-based systems. We thus consider the procedure $\Phi: \mathbb{R}^N \rightarrow \mathbb{R}^M$ purely as a dimensionality reduction operation as it produces a lower dimension representation of our input signal.

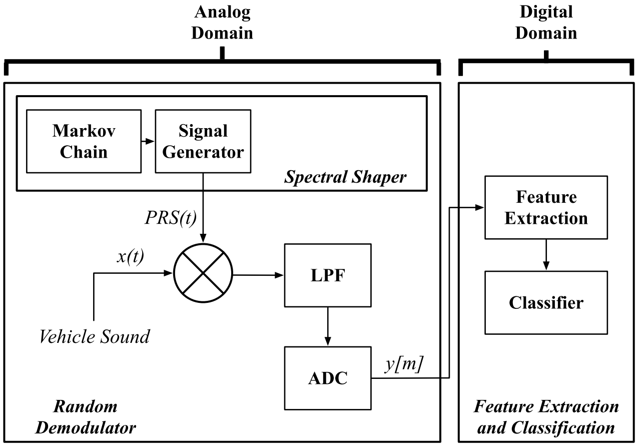


FIGURE 1. Proposed system overview. A key feature of our proposed system is the simultaneous sampling and filtering of an input signal $x(t)$ during the signal acquisition process using a spectrally shaped bipolar pseudorandom sequence $PRS(t)$ generated by the *Spectral Shaper*.

For us to be able to consider compressive measurements as low-dimensionality representations of an input signal, we must ensure that certain conditions are met, most importantly that for any two distinct signals x_1 and x_2, $\Phi x_1 \neq \Phi x_2$, and thus $\Theta\alpha_1 \neq \Theta\alpha_2$. This is guaranteed if Φ and Θ follow the criteria outlined in [25]. Furthermore, as stated in Section III-A, ensuring that these two matrices follow the aforementioned criteria guarantees that the information contained in α (and thus, x) is present in the measurements y.

The term *compressive signal processing* (CSP) was first coined by the authors of [21] and refers to the process of performing detection, classification, and filtering directly on the compressive measurements obtained during the CS process without prior reconstruction. This concept has been further explored and analyzed in [22], [28], [29]; and while the goals and approaches differ, the fundamental idea of bypassing the reconstruction of x and instead directly leveraging y remains the same. In a similar manner, in our paper, we perform AVDI by extracting features directly from compressive measurements.

IV. PROPOSED SYSTEM

A. SYSTEM OVERVIEW

Our proposed system aims to obtain information from a specific frequency band within the audio signals of passing vehicles while sampling at sub-Nyquist rates. Features are extracted from the samples obtained in this manner and are used to detect and identify the vehicles.

An overview of our proposed system is shown in Figure 1 and is made up of three sections: *Random Demodulator*, *Spectral Shaper*, and *Feature Extraction and Classification*.

Input signals are simultaneously filtered and sampled at a sub-Nyquist rate in the *Random Demodulator* section using a Markov chain-generated spectrally shaped bipolar pseudorandom sequence $PRS(t)$ generated by the *Spectral Shaper* section. Features are then extracted from the acquired

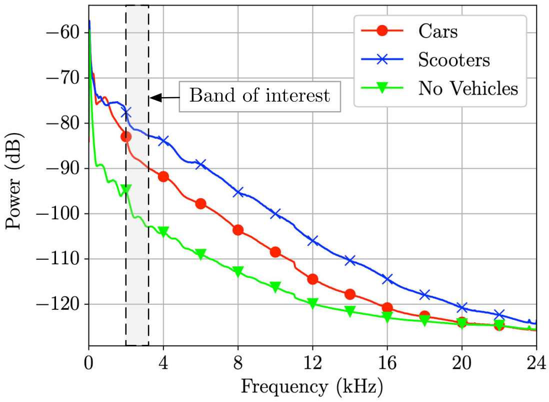


FIGURE 2. **Average audio signals for three vehicle classes.**

samples and used to detect and identify vehicles in the *Feature Extraction and Classification* section.

In our application, the specific frequency band of interest can be determined by examining the average frequency-domain plots of the vehicle classes under consideration. Figure 2 shows the average sound signals of passing cars, scooters, and periods without a passing vehicle, sampled for a duration $T_s = 2s$ at a rate $W = 48$ kHz. We can see that the signals show the clearest separation for the highest signal power over a narrow band of interest located just under 3 kHz.

As highlighted in Figure 1, in our proposed system, the processes that constitute the *Random Demodulator* and *Spectral Shaper* sections are performed on continuous-time analog signals, and the processes that constitute the *Feature Extraction and Classification* section are performed on discrete-time digital signals. These stages will be examined in more detail in the rest of Section IV.

B. RANDOM DEMODULATOR

At the heart of our proposed system is an architecture called the *Random Demodulator* (RD), which is composed of the following blocks: a signal generator, a mixer, a low-pass filter (LPF), and an analog-to-digital-converter (ADC). In our proposed C-AVDI system, we include an additional Markov chain block that is used in tandem with the signal generator to spectrally shape the *PRS*.

First presented in [30] and further expanded upon in [31] and most notably in [32], the RD enables us to perform CS on sparse or compressible continuous-time signals rather than the exclusively discrete signals described in initial theoretical work. Compared to more sophisticated architectures such as the modulated wideband converter (MWC) in [22] and quadrature analog-to-information converter (QAIC) in [33], the RD is more straightforward and cheaper to implement as it is single channel and only requires a single ADC. Furthermore, the RD is particularly suitable for our application as it is designed to acquire sparse single band multitone signals, unlike the MWC for instance, which is a multichannel architecture designed to acquire sparse multiband signals. More details on different CS acquisition strategies can be found in [27], and [34] provides an in-depth comparison of the RD and MWC systems in particular.

Intuitively, we can describe the RD's operation as follows: rather than acquiring a signal through traditional Nyquist sampling, the RD first demodulates the signal by multiplying it with a white noise-like pseudorandom sequence, spreading the signal's frequency content across the entirety of the spectrum. The resulting signal is then low-passed[1] before being sampled at a sub-Nyquist rate. If required, the original signal can be recovered from the sub-Nyquist samples through l_1 minimization as in Section III-A.

Let us describe the operation of the RD more formally. An analog signal as described in (1) is combined with a pseudorandom bipolar sequence of unitary amplitude defined as:

$$PRS(t) = \epsilon_n, \ \ t \in \left[\frac{n}{C}, \frac{n+1}{C}\right), \ \ n = 0, 1, \ldots, N-1 \quad (7)$$

where ϵ_n is a Rademacher sequence that switches between values $\{-1, 1\}$ at a rate $C = W$.

The combined signal $x(t)PRS(t)$ is passed through an anti-aliasing LPF $h(t)$ and sampled at a rate $R < W$ to obtain linear compressive samples $y[m]$. This procedure can be expressed as a multiplication followed by a convolution in the time domain:

$$y[m] = \int_{-\infty}^{\infty} x(\tau)PRS(\tau)h(t-\tau)\,d\tau \bigg|_{t=mR} \quad (8)$$

$$= \sum_{n=1}^{N} \alpha_n \int_{-\infty}^{\infty} \psi_n(\tau)PRS(\tau)h(mR-\tau)\,d\tau \quad (9)$$

From which we obtain an expression for Θ, whose entries are defined as $\theta_{m,n}$ for row m and column n:

$$\theta_{m,n} = \int_{-\infty}^{\infty} \psi_n(\tau)PRS(\tau)h(mR-\tau)\,d\tau \quad (10)$$

where Θ is a combination of the matrix Φ which represents the sequence of operations mapping the input signal x to the compressive measurements y, and of the orthonormal basis matrix Ψ. It is shown in [32] that Θ satisfies the previously outlined RIP conditions, and more importantly the dimensionality reduction properties outlined in Section III-B, as long as the number of measurements matches or exceeds M such as:

$$M = \mathcal{O}\left(K \log \frac{N}{K}\right) \quad (11)$$

Finally, it is necessary to ensure that in our proposed application, the input signals to the RD are sparse or compressible in the domain defined by Ψ. A key feature of our proposed system is the simultaneous sampling and filtering performed during the signal acquisition process using the

[1] In the traditional presentation of the RD architecture, the filtering is described as being performed by an integrator. Often, however, in practice this integrator is considered as, or replaced by an LPF [23], [32], [35]–[37]. In our implementation of the RD, the filtering is performed using an LPF, and will be referred to and represented as such in the remainder of the paper.

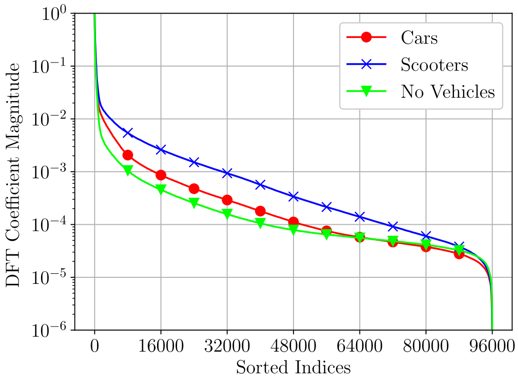


FIGURE 3. Sorted DFT coefficients $|\alpha_n|$ (rescaled).

Spectral Shaper to match the spectra of the bipolar sequence and input signal; an approach based on the results presented in [23] and [39]. We therefore consider the signals in this paper to be sparse or compressible in the frequency domain (Ψ = DFT matrix). Signals are only very rarely perfectly sparse and are much more likely to be compressible, that is, the magnitudes of the nonzero coefficients present in the signal decay following a power law distribution. This is defined in [38] as:

$$|\alpha_n| \leq Pn^{-q} \text{ for } n \in \{1, 2, \ldots, N\} \tag{12}$$

where P and q are constants.

Figure 3 shows the DFT coefficient distributions of the signals shown in Figure 2. We can see that the signals are compressible as their spectra are clearly dominated by a relatively small number K of high magnitude coefficients.

C. SPECTRAL SHAPER

The *Spectral Shaper* consists of a signal generator and Markov chain block. It is used to produce bipolar sequences with specifically tailored, rather than random, frequency distributions. Indeed, in more traditional RD architectures, the particular $PRS(t)$ used to demodulate the input signal switches polarity with equal probability; as a result, its frequency-domain representation resembles that of white noise. This ensures equal spreading of all K components contained within $x(t)$, which is optimal if we do not have prior knowledge of the distribution of the frequency information within the signal's bandlimit.

If, however, we do have prior knowledge of the locations of interest, then [23] shows that signal reconstruction accuracy can be improved by using bipolar sequences whose frequency-domain representation matches that of the input signal. The authors of [39] demonstrate that the bipolar sequence can be generated using a Markov chain with each state corresponding to an output symbol of ± 1. Thus, the polarity of $PRS(t)$ at a given time is determined by the corresponding Markov chain's state transition probability and chain length. The authors also establish that the Φ matrix obtained as a result of using a Markov-chain generated $PRS(t)$ satisfies the required RIP conditions outlined previously with very high probability. We base the design of our *Spectral Shaper* on these results and expand upon the single Markov chain sequence generation method proposed in [39] by designing a dual Markov chain sequence generation method, capable of creating bipolar sequences with more complex spectra.

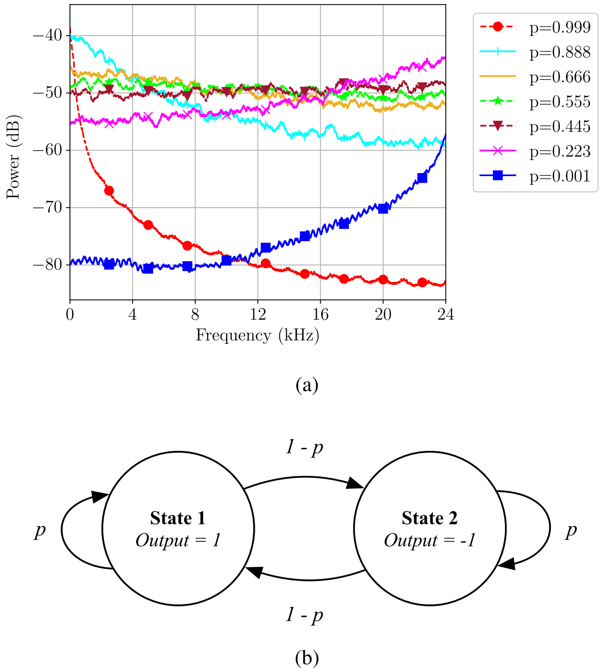
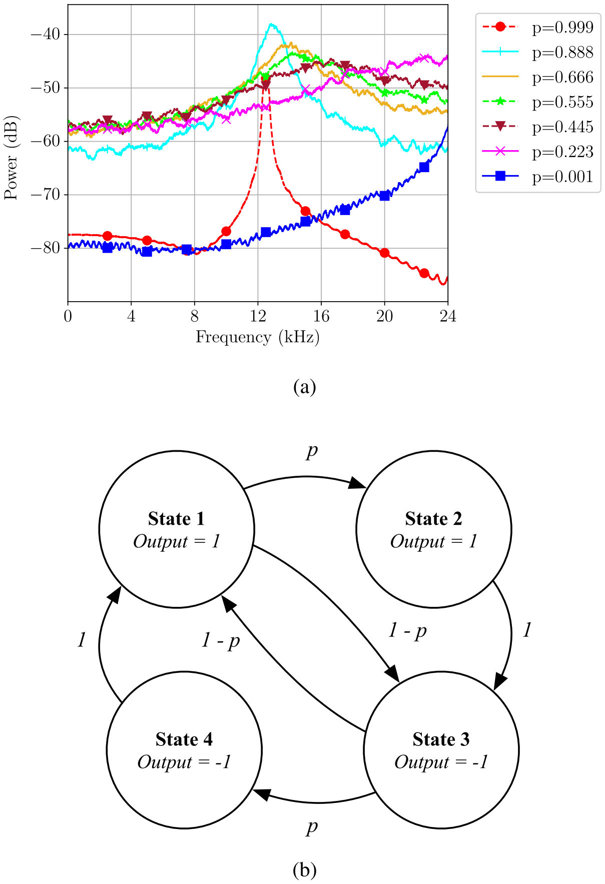


FIGURE 4. Spectra (a) and corresponding state transition diagram (b) of 2-state Markov chain-generated bipolar sequences. Each spectrum corresponds to a different transition probability p.

Figure 4 and Figure 5 show a selection of spectrally shaped bipolar sequence spectra and the diagrams of the corresponding Markov chains used to create them. The transition probability matrices corresponding to the 2-state and 4-state Markov chains are defined as $\mathbf{P_1}$ and $\mathbf{P_2}$ respectively:

$$\mathbf{P_1} = \begin{pmatrix} p & 1-p \\ 1-p & p \end{pmatrix}$$
$$\mathbf{P_2} = \begin{pmatrix} 0 & p & 1-p & 0 \\ 0 & 0 & 1 & 0 \\ 1-p & 0 & 0 & p \\ 1 & 0 & 0 & 0 \end{pmatrix} \tag{13}$$

We sweep the value of the transition probability p over the range $0 < p < 1$. The higher the value of p is, the more likely the 2-state chain is to stay in its current state, and the more likely the 4-state chain is to transition to a state with the same output. In the case of a 2-state chain, this results in more of the generated signal's energy being located towards the lower end of its bandlimit; and in the case of a 4-state chain this results in the generated signal's energy tending towards a narrow peak in the middle of its bandlimit. Conversely, a lower value of p means that both chains are more likely to transition to a state with a different output, leading to the energy of both of the generated signals being located towards the higher end of their respective bandlimits. It is important



(a)

(b)

FIGURE 5. Spectra (a) and corresponding state transition diagram (b) of 4-state Markov chain-generated bipolar sequences. Each spectrum corresponds to a different transition probability *p*.

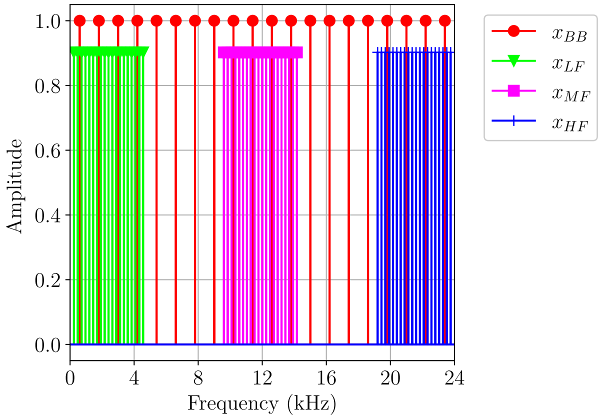


FIGURE 6. Spectra of test signals x_{LF}, x_{MF}, x_{HF}, and x_{BB}. Each signal has the same bandlimit but a different frequency composition (low-frequency, mid-frequency, high-frequency, and broadband).

to note that this relationship is due to the way $\mathbf{P_1}$ and $\mathbf{P_2}$ were designed: permuting the rows and columns of these matrices would change our state diagram and reverse the relationship between frequency distribution and transition probability.

D. CS WITH SPECTRALLY SHAPED BIPOLAR SEQUENCES

As stated previously, our final C-AVDI system does not reconstruct the original input signal at any point in its operation. During the system design process, however, it is important to quantify and visualize the effects of matching $PRS(t)$ and $x(t)$, which can be done most effectively by reconstructing test signals using different bipolar sequence generation parameters and examining the results.

This is done by performing CS using the RD on a set of four test signals whose spectra are shown in Figure 6, varying the Markov chain's p-value over each run for both a 2-state and a 4-state chain. The test signals each have the same bandlimit but different frequency compositions: a low-frequency x_{LF} signal, mid-frequency x_{MF} signal, high-frequency x_{HF} signal, and broadband x_{BB} signal. We denote the corresponding reconstructed versions of these signals as $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$ and $\hat{x}_{BB}$ (reconstruction is performed via basis pursuit using the SPGL1 toolbox available from [40]).

TABLE 2. Test signal reconstruction parameters.

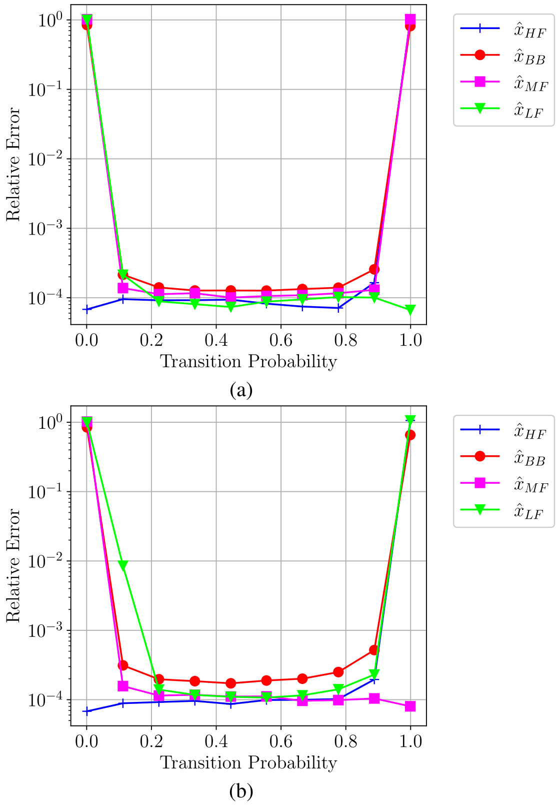
Nyquist Rate W	48 kHz
Switching Rate C	48 kHz
ADC Rate R	12 kHz
Bit Depth B	12 bits
Signal Length (Time) T_s	$1s$
Signal Length (Samples) N	48000
Signal Sparsity K	40
Compressive Measurements M	12000

The test signal reconstruction parameters are shown in Table 2. The optimal value of M is determined by first establishing a theoretical lower bound value using (11), from which an optimal value of M is obtained experimentally by repeatedly performing the CS process on our input signals while incrementally increasing M and noting the resulting relative error (RE). Repeating this process enables us to find the values of M after which the RE levels off for each of the four test signals. We use the largest of the four values, $M = 12000$.

Figure 7 plots the resulting RE of the original and reconstructed signals, and the values of p for the optimal reconstruction of each signal are summarized in Table 3. These results confirm that for a constant value of M, matching $PRS(t)$ and $x(t)$ by changing the value of p improves reconstruction performance.

E. INPUT SIGNAL FILTERING WITH SPECTRAL SHAPER

In addition to improving reconstruction accuracy, a tailored $PRS(t)$ can act as a filter to the input signal, attenuating or amplifying specific frequency content. Thus, we can equate



(a)

(b)

FIGURE 7. Relative error as a function of transition probability for a) a 2-state and b) a 4-state Markov chain-generated *PRS*(*t*).

the mixing of $x(t)$ and $PRS(t)$ to a simultaneous demodulating and filtering operation in which the spectrum of the bipolar sequence is likened to the frequency response of a filter. This property is a key feature of our proposed system, and again is most effectively quantified and visualized during the system design process by reconstructing the test signals in Figure 6.

1) SINGLE CHAIN-GENERATED SEQUENCES

We visualize the filtering effects of a spectrally tailored $PRS(t)$ on an input signal by plotting the spectra of test signals $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$ and $\hat{x}_{BB}$ reconstructed using different $PRS(t)$ with Markov chain transition probability values $p = 0.01$, $p = 0.555$, and $p = 0.999$.

Figure 8 and Figure 9 show the test signal spectra reconstructed using a 2-state chain and a 4-state chain respectively. We can see that the filtering effect a bipolar sequence has on a test signal depends on the length and p-value of the Markov chain used to it. This is best illustrated by x_{BB}: for a 2-state chain-generated $PRS(t)$, a value of $p = 0.001$ suppresses the low frequency content of $\hat{x}_{BB}$, and a value of $p = 0.999$ suppresses the high frequency content; and for a 4-state chain-generated $PRS(t)$, a value $p = 0.001$ again suppresses the low frequency content of $\hat{x}_{BB}$, and a value of $p = 0.999$ suppresses the high and low frequency content. The filtering effect is consistent with the shape of the $PRS(t)$ spectra shown in Figure 4 and Figure 5.

TABLE 3. Single chain parameters for optimal reconstruction.

Chain Length	Signal	p	RE [$\times 10^{-5}$]
2-state	$\hat{x}_{LF}$	0.999	6.67
	$\hat{x}_{MF}$	0.445	10.0
	$\hat{x}_{HF}$	0.001	6.80
	$\hat{x}_{BB}$	0.555	12.7
4-state	$\hat{x}_{LF}$	0.555	10.7
	$\hat{x}_{MF}$	0.999	8.00
	$\hat{x}_{HF}$	0.001	6.80
	$\hat{x}_{BB}$	0.445	17.2

2) DUAL CHAIN-GENERATED SEQUENCES

In order to create $PRS(t)$ with more complex spectra, we design combined dual Markov chain-generated sequences by mixing two different single chain sequences, which we define as $PRS_C(t)$, $PRS_1(t)$, and $PRS_2(t)$ respectively, where $PRS_C(t) = PRS_1(t)PRS_2(t)$.

To fully assess the effects of $PRS_C(t)$ on a given input signal, we perform three reconstruction tests using the x_{LF}, x_{MF}, x_{HF} and x_{BB} signals shown in Figure 6, while varying the p-values of $PRS_1(t)$, $PRS_2(t)$ (p_1 and p_2 respectively), and chain lengths. The purpose of these three reconstruction tests is to visualize different properties of our proposed dual chain-generated $PRS_C(t)$ approach:

- The purpose of the first reconstruction test is to gauge the improvements in signal reconstruction when using a dual chain-generated bipolar sequence. This is achieved by determining the optimal $PRS_C(t)$ that minimize the reconstruction RE for each of the four test signals and comparing their REs to those of their single chain-generated counterparts shown in Table 3. The spectra of the optimal dual chain-generated sequences and of the reconstructed test signals are shown in Figure 10, and the optimal dual chain reconstruction parameters are summarized in Table 4. These results show that using dual chain-generated sequences improves reconstruction performance compared to single chain-generated sequences, even if only marginally.
- The purpose of the second reconstruction test is to visualize the effects of any mismatches between the input signal and dual chain-generated sequence. This is achieved by reconstructing the four test signals using $PRS_C(t)$ designed to maximize reconstruction RE and comparing the spectra of the original and reconstructed signals. The spectra of the reconstructed signals and of their respective sequences are shown in Figure 11. These results show the impact of a mismatched $PRS_C(t)$ on signal reconstruction, underlining the importance of using an appropriately designed $PRS_C(t)$.
- The purpose of the third reconstruction test is to visualize the filtering properties of the $PRS_C(t)$. We reconstruct the broadband signal, x_{BB}, using four $PRS_C(t)$ sequences whose spectra can be likened to a low-pass, band-pass, and high-pass filter for the first three signals, and with the spectrum of the final signal resembling

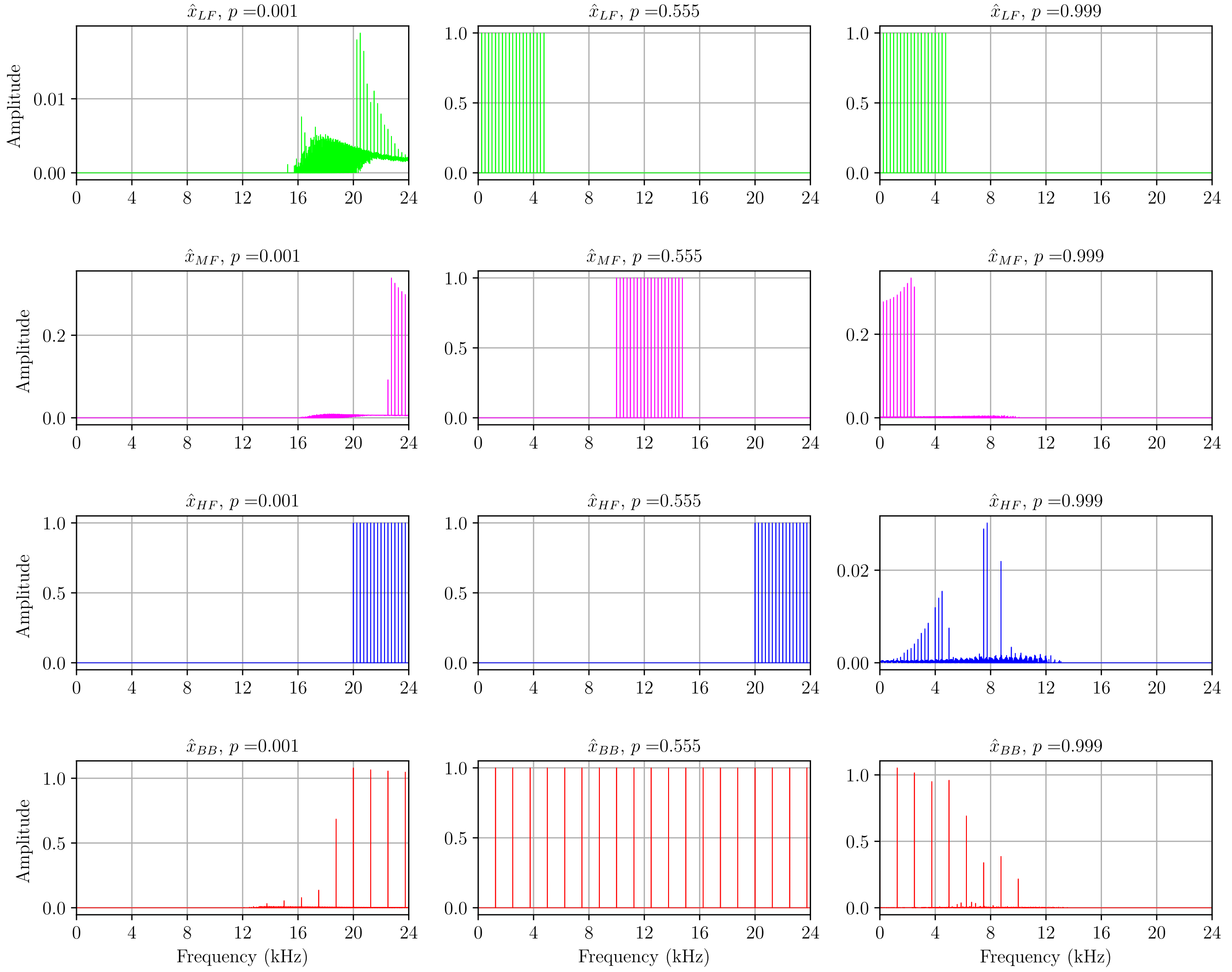


FIGURE 8. Reconstructed test signals $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$, and $\hat{x}_{BB}$ using 2-state Markov chain-generated *PRS*(t) with transition probability p. Each row represents one of the four test signals presented in Figure 6, and each column represents a different value of p.

white noise. The spectra of the bipolar sequences and of the reconstructed test signals are shown in Figure 12. These results show that in addition to having a direct impact on reconstruction performance, a tailored $PRS_C(t)$ can attenuate or amplify the frequency content of the input signal.

F. FEATURE EXTRACTION AND CLASSIFICATION

The *Feature Extraction and Classification* section is composed of a feature extraction block and a classifier block. In the feature extraction block, we extract a set of 5 features from the $y[m]$ measurements produced by the RD. We select the 5 most important features out of the 9 used in our preliminary work [1] to use in our C-AVDI system:

- *mean*
- *standard deviation*
- *median*
- *absolute max value*
- *interquartile range*

In the classifier block, these extracted features are used as inputs to a multiclass classifier to detect and identify passing vehicles. Classification is performed using an RF classifier, chosen for its robustness in the presence of outliers, inherent suitability for multiclass classification, and minimal preprocessing requirements (no input data rescaling required). The RF is implemented using the *scikit-learn* library [41].

V. EVALUATION

System evaluation is performed using a software-based implementation of the proposed system with audio data collected from vehicles traversing a university campus.

A. DATA ACQUISITION

The data acquisition setup is shown in Figure 13. A pair of Azden SGM-990 microphones are installed parallel to a one-lane two-way road and connected to a Sony HDR-MV1 video camera. The microphones are 1m from the ground, the intra-microphone distance is 50 cm, the distance between

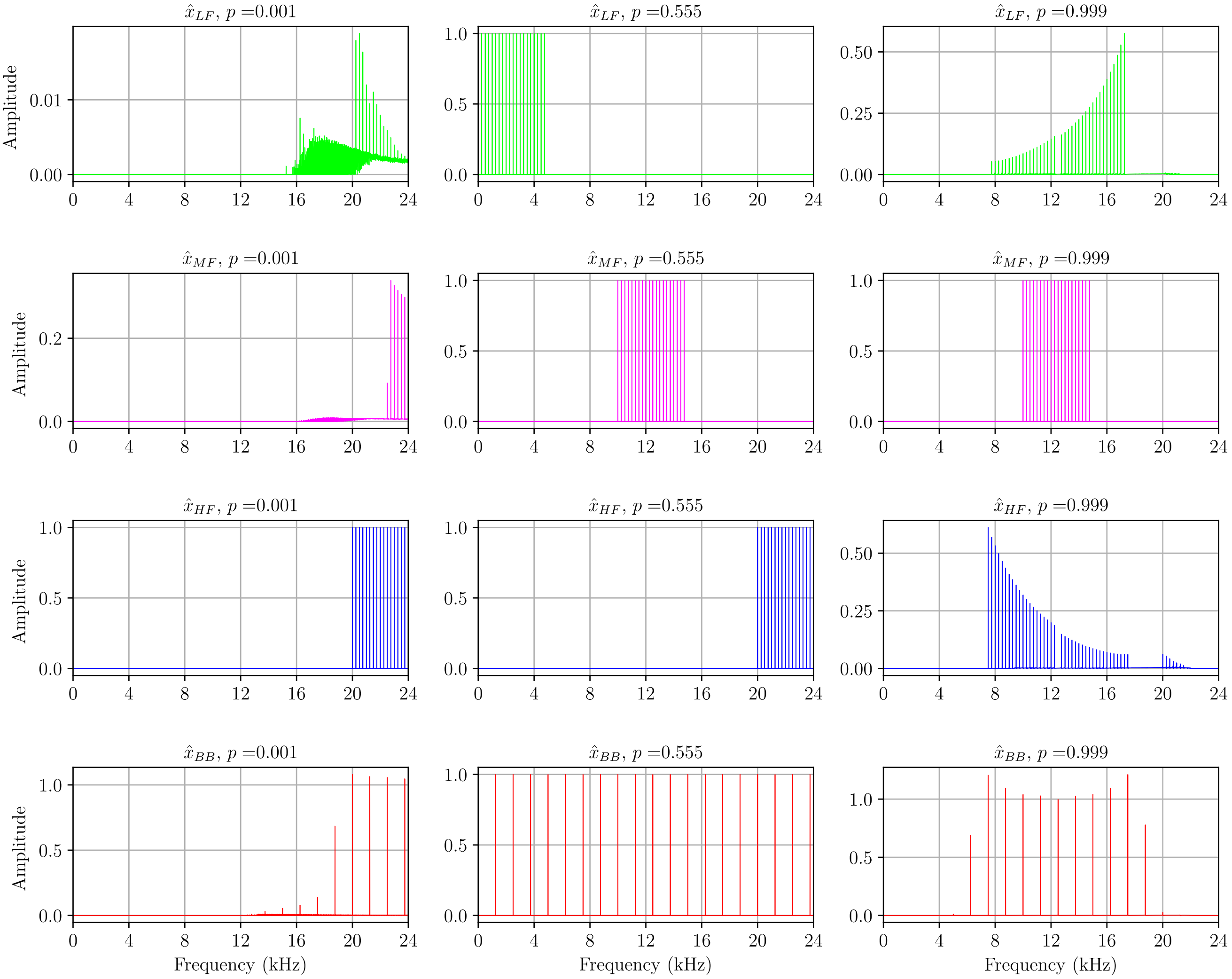


FIGURE 9. Reconstructed test signals $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$, and $\hat{x}_{BB}$ using 4-state Markov chain-generated $PRS(t)$ with transition probability p. Each row represents one of the four test signals presented in Figure 6, and each column represents a different value of p.

the microphones and the center of the front lane is 3 m, and the distance between the microphones and the center of the back lane is 6 m. The microphones' pickup pattern is set to *cardioid*, and they record the sound of passing vehicles for a 30-minute duration at a sample rate of 48 kHz and a bit depth of 16 bits. We average the signals obtained from the two microphones to create a mono signal used in subsequent analysis.

We obtain 14 different datasets by recording vehicle sounds at the same location with the same setup on 14 different days; at different times of day; on different days of the week; and under clear, windy, and rainy weather conditions. The average vehicle signal plots using data obtained under all weather conditions are shown in Figure 2, and the individual average vehicle signal plots for each weather condition are shown in Figure 14.

The three classes considered for classification are cars, scooters/motorbikes, and no vehicles. We refer to these as *Car*, *Scooter*, and *NoVeh*, respectively. The time at which a given vehicle passes in front of the middle of the microphone pair is defined as t_p, and the time window for the vehicle as $T_r = \left[t_p - \frac{T_s}{2}\ ;\ t_p + \frac{T_s}{2}\right]$, where $T_s = 2s$. As we are not seeking to detect successively or simultaneously passing vehicles in this evaluation, we retain only the vehicle signals whose T_r does not overlap with those of the preceding or following signals. The total number of vehicle sounds obtained for each class on each day can be seen in Table 5.

B. SYSTEM SIMULATION

In this section, we evaluate the system described in Section IV-A by simulating its operation in software. Figure 15 shows an overview of the software implementation of the system proposed in Figure 1. As previously mentioned, our proposed system operates by taking an audio signal emitted from a passing vehicle as input, sampling it at a sub-Nyquist rate, and outputting a predicted vehicle type. System performance is evaluated using the accuracy metric.

As seen in Section IV-B, the switching rate C of a bipolar sequence is defined as twice the highest frequency contained

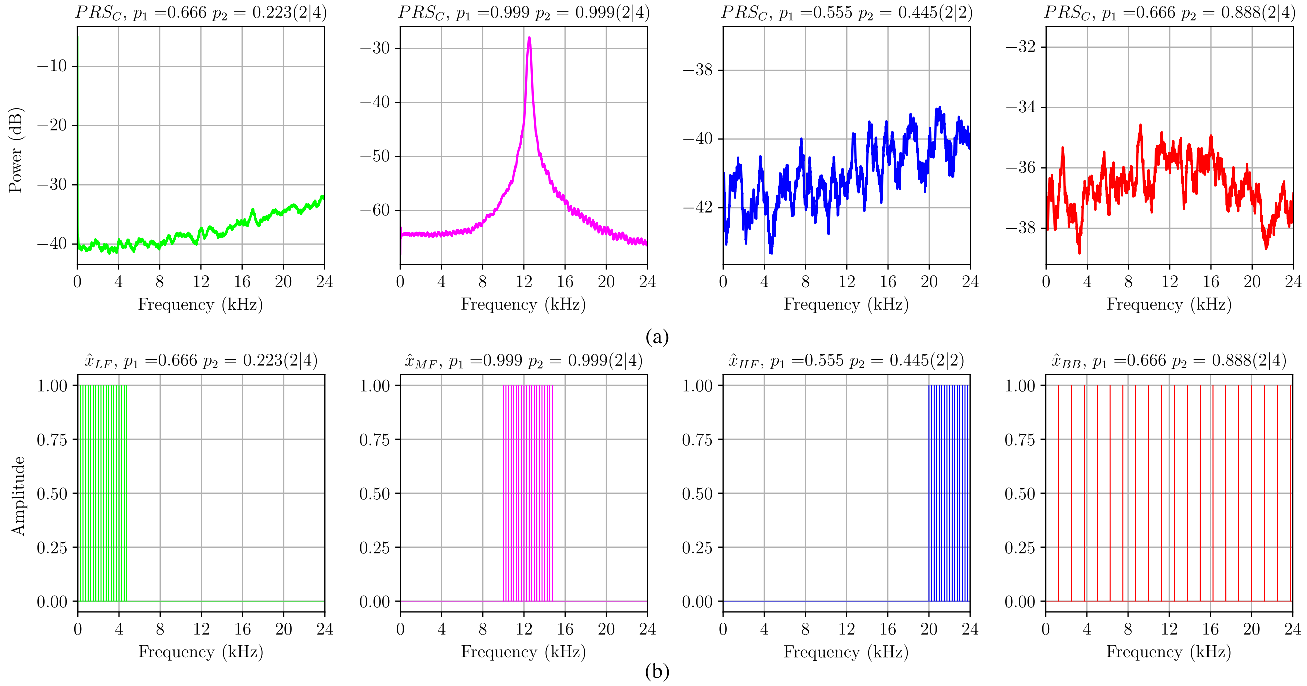


FIGURE 10. Spectra of a) PRS_C and (b) reconstructed test signals $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$, and $\hat{x}_{BB}$ obtained during the first reconstruction test, with the corresponding p_1, p_2, and respective chain lengths indicated in parentheses above each subplot.

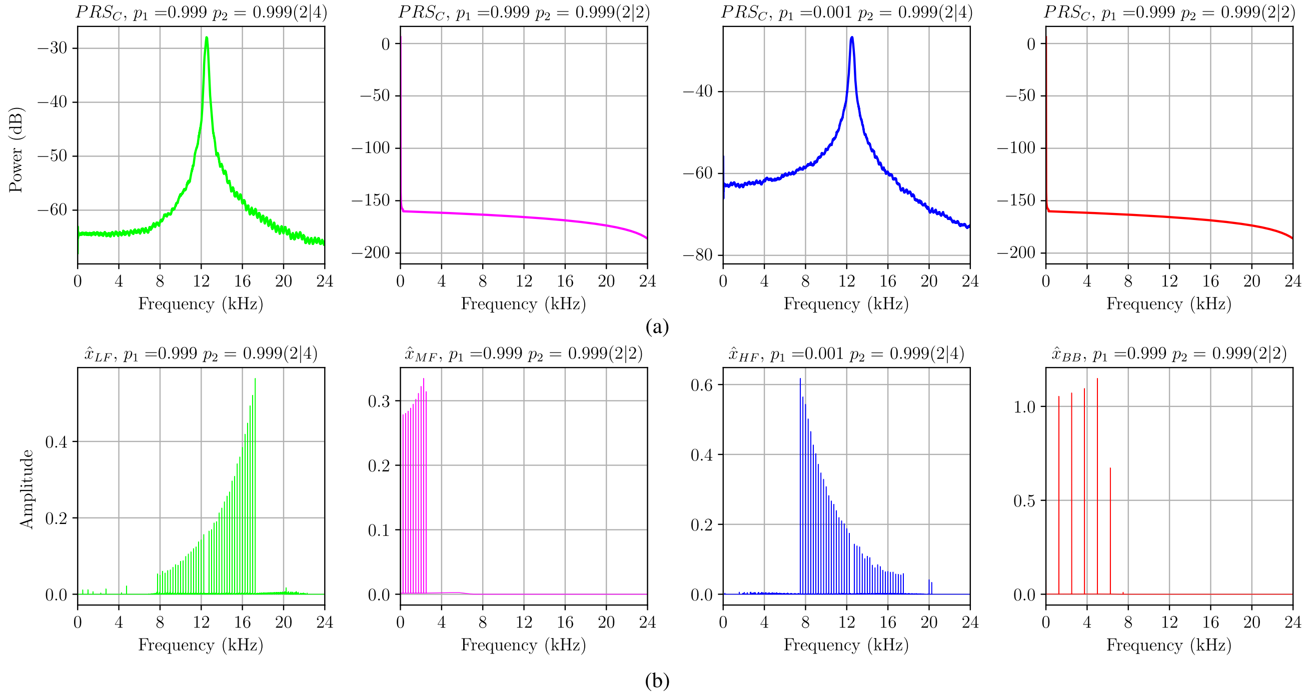


FIGURE 11. Spectra of a) PRS_C and (b) reconstructed test signals $\hat{x}_{LF}$, $\hat{x}_{MF}$, $\hat{x}_{HF}$, and $\hat{x}_{BB}$ obtained during the second reconstruction test, with the corresponding p_1, p_2, and respective chain lengths indicated in parentheses above each subplot.

in $x(t)$. Typically, C would be set to 48 kHz to match the sample rate of the microphone's ADC, ensuring that the entirety of the input signal's frequency content has at least a partial tone signature representation at baseband. We, however, are interested in the information contained in a narrow band whose upper limit is 3 kHz, so we set the $PRS(t)$ switching rate as $C = 6$ kHz accordingly. This limits the amount of high frequency content spread across the frequency spectrum during the demodulation process: in our case, we can expect all the frequency components below 3 kHz in the input signal to have at least a partial tone signature representation at baseband, but not the components located above 3 kHz.

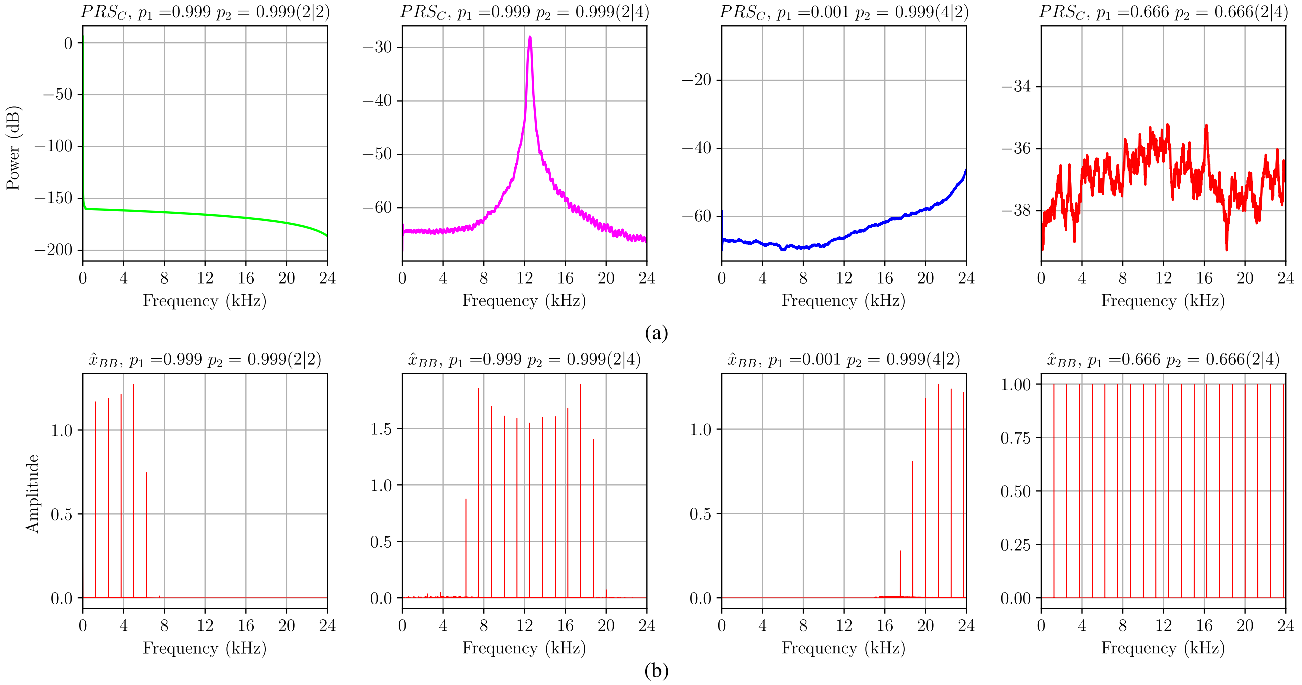


FIGURE 12. Spectra of a) PRS_C and (b) reconstructed test signal $\hat{x}_{BB}$ obtained during the third reconstruction test, with the corresponding p_1, p_2, and respective chain lengths indicated in parentheses above each subplot.

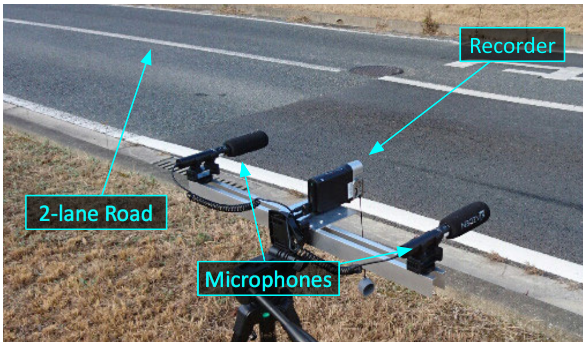


FIGURE 13. Experimental setup.

We are looking to design a $PRS(t)$ that, when used to demodulate an input signal, will strongly emphasize the frequency content present in the band of interest outlined in Figure 2 and Figure 14. This same sequence will be used when acquiring data under all three weather conditions, as the location of the frequency band of interest is the same in all three cases.

We create a $PRS_C[n] \in \mathbb{R}^N$ with switching rate C, from $PRS_1[n] \in \mathbb{R}^N$ a 4-state chain with $p = 0.001$ and $PRS_2[n] \in \mathbb{R}^N$ a 2-state chain with $p = 0.999$. The spectrum of our $PRS_C[n]$ is shown in Figure 16.

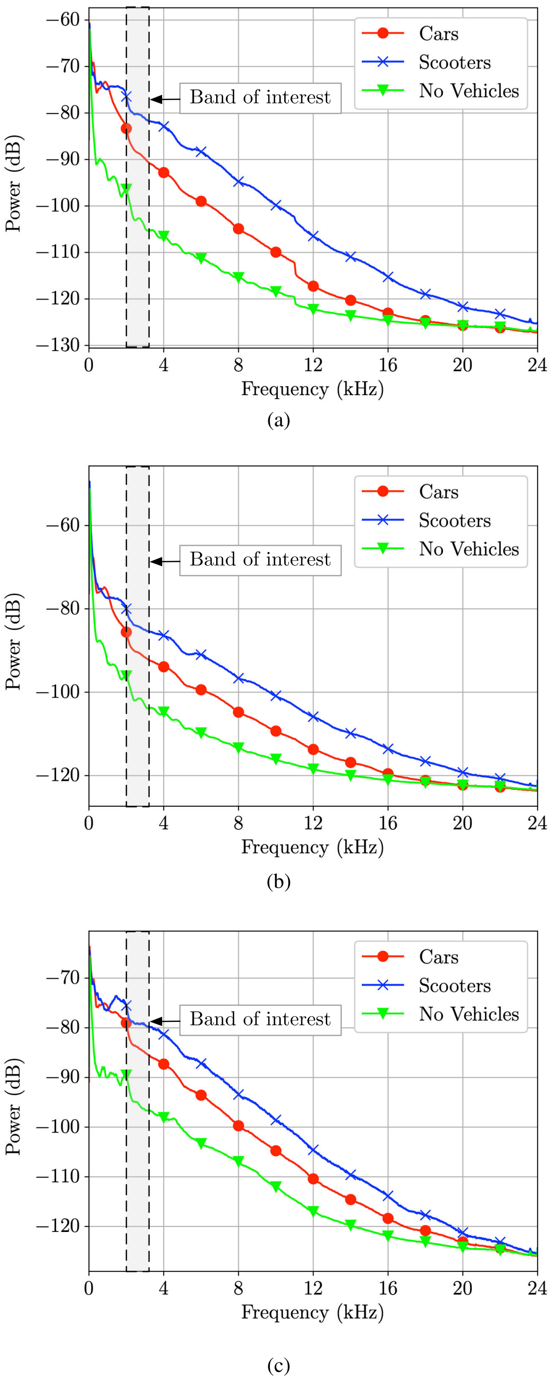
Our input signal is represented as a discrete version of the analog input signal $x(t)$, defined as $x[n] \in \mathbb{R}^N$ sampled at rate $W = 48\,\text{kHz}$ during the data acquisition process. We approximate the sparsity level of $x(t)$ as $K = 400$ by determining the number of DFT coefficient magnitudes shown in Figure 3 such that $|\alpha_n| \geq 10^{-1}$, and rounding to a single significant figure.

TABLE 4. Dual chain parameters for optimal reconstruction.

Chain		Signal	p_1	p_2	RE
1st	2nd				$[\times 10^{-5}]$
2-state	2-state	$\hat{x}_{LF}$	0.666	0.001	6.64
		$\hat{x}_{MF}$	0.888	0.555	9.20
		$\hat{x}_{HF}$	0.555	0.445	6.64
		$\hat{x}_{BB}$	0.666	0.445	12.2
2-state	4-state	$\hat{x}_{LF}$	0.666	0.233	6.52
		$\hat{x}_{MF}$	0.999	0.999	7.98
		$\hat{x}_{HF}$	0.999	0.001	6.81
		$\hat{x}_{BB}$	0.666	0.888	11.2
4-state	2-state	$\hat{x}_{LF}$	0.223	0.666	6.52
		$\hat{x}_{MF}$	0.999	0.999	7.98
		$\hat{x}_{HF}$	0.001	0.999	6.81
		$\hat{x}_{BB}$	0.888	0.666	11.2
4-state	4-state	$\hat{x}_{LF}$	0.999	0.999	6.68
		$\hat{x}_{MF}$	0.999	0.001	8.12
		$\hat{x}_{HF}$	0.555	0.445	7.00
		$\hat{x}_{BB}$	0.888	0.666	11.3

The anti-aliasing LPF preceding the ADC is a 2nd order Butterworth filter, and the measurements $y[m] \in \mathbb{R}^{\mathbb{M}}$ are obtained by sampling and quantizing the combined $x[n] \odot PRS_C[n]$ signal (where $\odot$ signifies elementwise multiplication) at rate $R = 3\,\text{kHz}$ and bit depth $B = 12$ Bits. This corresponds to a reduction in sample rate by a factor of $(\frac{48\,\text{kHz}}{6\,\text{kHz}})(\frac{6\,\text{kHz}}{3\,\text{kHz}}) = 16$.

Similarly to Section IV-D, the optimal value of M for our system is determined by first establishing a theoretical lower bound value that is incrementally increased until we obtain the optimal balance between the smallest number of $y[m]$ samples and maximum system prediction accuracy. We find



(a)

(b)

(c)

FIGURE 14. Average audio signals for three vehicle classes recorded in a) clear, b) windy, and c) rainy conditions.

this value to be $M = 6000$ and list it, along with the rest of the parameters used in our system simulation, in Table 6.

The discrete-time and frequency-domain representations of the average $y[m]$ measurements for each class obtained by our simulated system are shown in Figure 17.

C. CLASSIFICATION

Classification is performed on 14 feature sets, which are obtained by extracting the 5 features outlined in Section IV-F from the $y[m]$ measurements of each of the 14 datasets described in Section V-A.

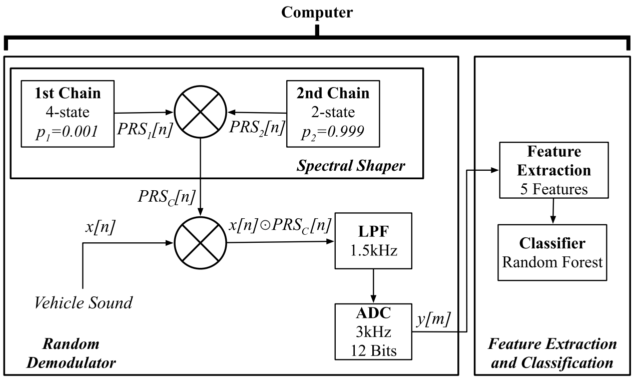


FIGURE 15. Overview of our proposed system's software implementation.

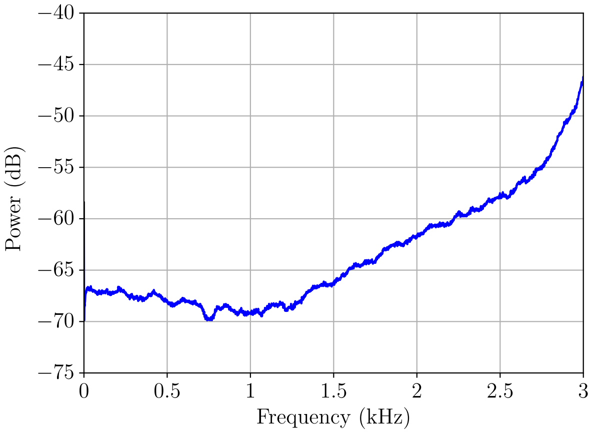


FIGURE 16. Spectrum of tailored $PRS_C[n]$.

TABLE 5. Weather conditions and number of vehicles for each day.

Day	Weather	Car	Scooter	NoVeh	Total
1	Clear	60	8	104	172
2	Clear	21	10	45	76
3	Wind	21	40	64	125
4	Clear	40	57	115	212
5	Clear	52	50	112	214
6	Rain	43	26	77	146
7	Rain	77	30	152	259
8	Clear	177	95	305	577
9	Wind	174	103	267	544
10	Clear	143	99	239	481
11	Wind	129	84	234	447
12	Clear	130	59	197	386
13	Wind	119	51	186	356
14	Rain	228	65	317	610

We measure the performance of our system using *leave-one-day-out* cross-validation: we set each one of the 14 feature sets (where each set corresponds to a different day) as the testing set, and the combined 13 remaining feature sets act as the training set. This process is performed 14 times in total, with each of the feature sets acting as the testing set in turn. We set the number of trees and the minimum number of samples per leaf parameters of the RF classifier as n_estimators = 1000 and min_samples_leaf = 3 respectively, with the other

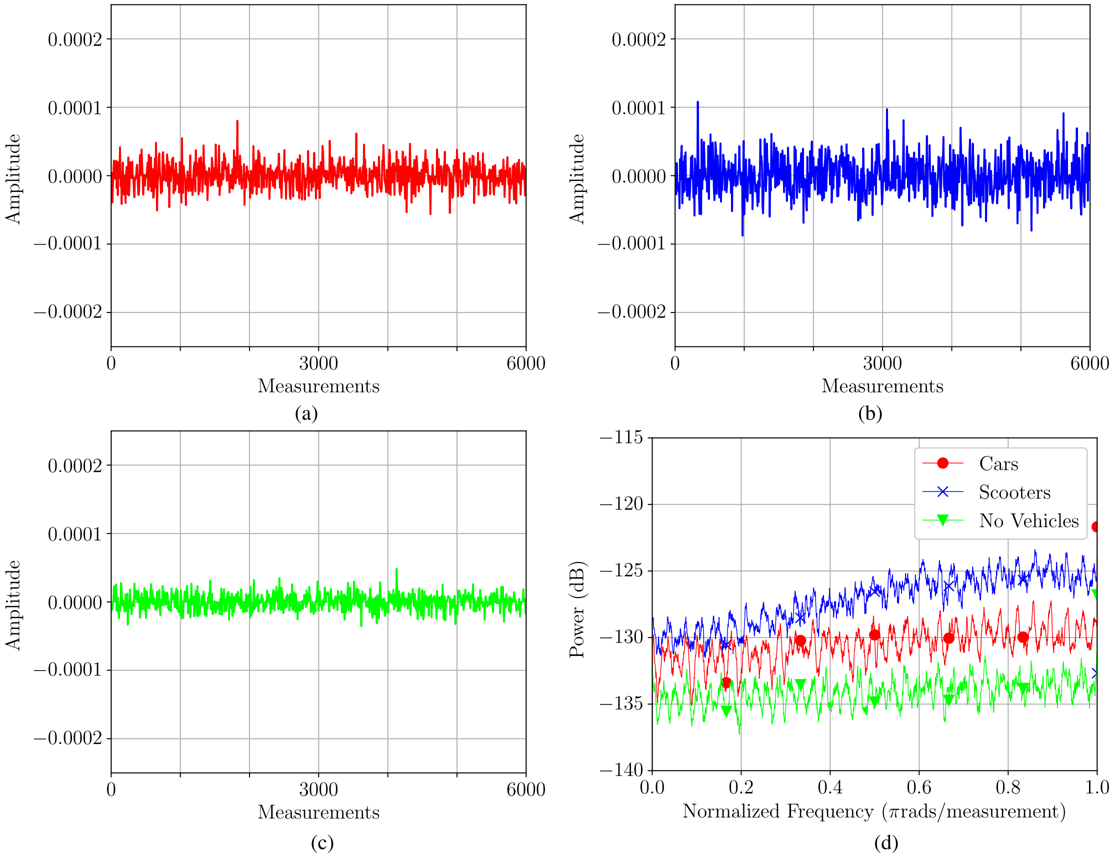


FIGURE 17. Compressive measurements $y[m]$ drawn from the average audio signals of a) *Cars*, b) *Scooters*, and c) *No Vehicles*. Subplot d) shows the respective normalized frequency representations of the $y[m]$ measurements.

TABLE 6. Simulated C-AVDI system parameters.

Nyquist Rate W	48 kHz
Switching Rate C	6 kHz
ADC Rate R	3 kHz
Bit Depth B	12 bits
Signal Length (Time) T_s	$2s$
Signal Length (Samples) N	96000
Signal Sparsity K	400
Compressive Measurements M	6000
1st Chain	4-state
1st Chain Transition Probability p_1	0.001
2nd Chain	2-state
2nd Chain Transition Probability p_2	0.999

TABLE 7. System accuracy by weather condition.

		All Conditions	Clear	Wind	Rain
Number of Vehicles	Car	1414	623	443	348
	Scooter	777	378	278	121
	NoVeh	2414	1117	751	546
Metric [%]	Accuracy	80	85	55	79

parameters left as default. Both the training and testing sets are balanced using random undersampling prior to classification, and the results obtained from each of the 14 runs are averaged to obtain the system accuracy. To account for any potential discrepancies caused by inherent randomness in the classification process, we perform the full classification process 10 times and average the obtained results, resulting in our final confusion matrix shown in Figure 18 and the system accuracy scores by weather condition shown in Table 7.

The confusion matrix in Figure 18 shows that the most prominent misclassification is that of *Car* as *Scooter*. This can be explained by the similar variance and amplitude of their $y[m]$ samples, which translates to similar *standard deviation* and *interquartile range* features in particular.

This information, along with the results presented in Table 7, gives us insight into how to modify the system to improve performance in future evaluations. Most notably, the breakdown of metrics by weather condition sheds light on one of the principal causes of misclassification: adverse weather conditions. The design of a system capable of more effectively mitigating the effects of wind and rain noise on classification accuracy will be the focus of future work.

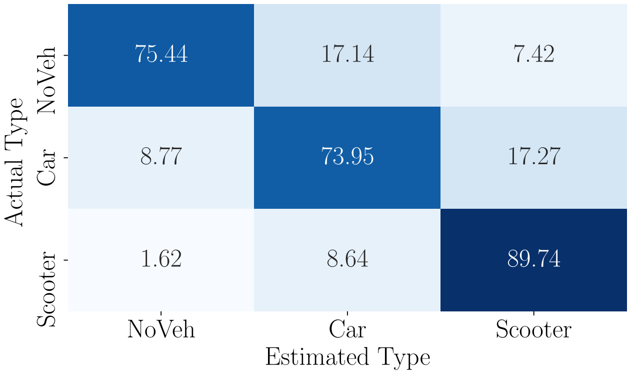


FIGURE 18. Confusion matrix of the proposed system's software implementation.

TABLE 8. Benchmarking results.

		C-AVDI	Mod-SMBAS
Time [s]	Extract Features	1.33	3.65
	Classify	0.09	0.28
	Total	1.42	3.93
Metric [%]	Accuracy	80	87

D. COMPUTATIONAL EVALUATION

We can gauge the computational performance of our proposed system by benchmarking it against our previous work. Benchmarking is performed by running the systems under test on the same computer under identical conditions and comparing their runtimes. We use a computer equipped with an Intel i9-9900K 16-core CPU @ 3.60 GHz with 64GB memory running Ubuntu 18.04. As in the previous section, the full process is run 10 times, from which we calculate the respective average runtimes.

The systems under test are the following:

1) The C-AVDI method presented in this paper.
2) A simplified single-microphone version of SMBAS, our previous system presented in [2] (Mod-SMBAS). For benchmarking purposes, we changed the classifier from SVM to RF, and removed the signal emphasis process.

The benchmarking results in Table 8 show that there is a link between computation time and performance: Mod-SMBAS shows the best accuracy but is more computationally intensive than C-AVDI by a factor of ≈ 2.8.

E. MICROCONTROLLER IMPLEMENTATION

We also seek to create an MCU implementation of the *Feature Extraction and Classification* section of the simulated system shown in Figure 15.

Table 9 shows the specifications of some commonly used MCUs. We choose the Teensy 4.1 MCU board for our MCU implementation as it has the most total available memory. The compressive measurements $y[m]$ are generated in the same manner as in our simulated system in Section V-B before being saved on an SD card from which they are sequentially loaded into the MCU for feature extraction. The previously trained classifiers are ported from their software implementations onto the MCU using the *emlearn* library [42].

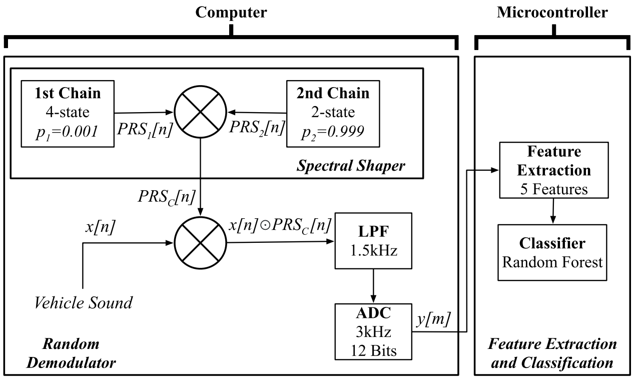


FIGURE 19. Overview of our proposed system's MCU implementation.

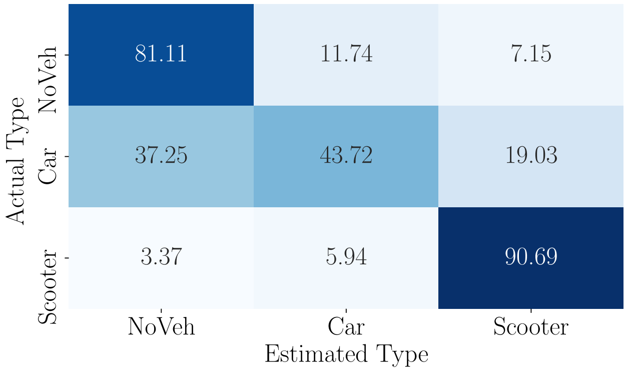


FIGURE 20. Confusion matrix of the proposed system's MCU implementation.

Figure 19 shows the MCU implementation of the proposed system: the *Random Demodulator*, and *Spectral Shaper* sections are implemented in software, and the *Feature Extraction and Classification* section is implemented on the MCU.

Figure 20 shows the resulting confusion matrix. We obtain an accuracy of 72%, which is lower than the accuracy obtained by the simulated system in Section V-B. Upon inspection, we find that the features extracted from $y[m]$ in the feature extraction block in both the software and MCU implementations of our system are identical. From this, we can infer that the difference in accuracy between both implementations is due to the difference in size and complexity between the ported and original versions of the classifier models. Indeed,the limited memory of the MCU restricts the number of trees in the RF classifiers to 100, compared to the 1000 used in the software implementation, while also truncating the size of the various coefficients, weights and parameters used in MCU implementation of the RF classifier, leading to a drop in classification accuracy.

VI. DISCUSSION

We can assess the performance and viability of our C-AVDI system by comparing it to SMBAS, our previous system presented in [2]. While SMBAS presents an accuracy of 95 %, which is higher than the 80 % obtained by our C-AVDI

TABLE 9. Specifications of commonly used microcontrollers and development boards.

Board (MCU)	Volatile Memory	Non-Volatile Memory	ADC Resolution	Power Consumption
Teensy 4.1[b] (iMX RT1062[c])	1024 kB	7936 kB	12 bits	10.0 mW at 24 MHz
LaunchPad[d] (MSP-EXP430FR599[e])	8 kB	256 kB	12 bits	5.9 mW at 16 MHz
Arduino Uno[f] (ATMega328P[g])	2 kB	32 kB	10 bits	26.0 mW at 8 MHz
PIC18F452[h]	8 kB	32 kB	10 bits	8.0 mW at 4 MHz

[b] https://www.pjrc.com/store/teensy41.html
[c] https://www.embeddedartists.com/products/imx-rt1062-oem/ based on iMX RT1060 https://www.nxp.com/docs/en/application-note/AN12245.pdf
[d] https://www.ti.com/lit/pdf/slau678
[e] https://www.ti.com/lit/ds/slase54d/slase54d.pdf?ts=1626677655975
[f] https://www.farnell.com/datasheets/1682209.pdf
[g] http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf
[h] https://ww1.microchip.com/downloads/en/DeviceDoc/39564c.pdf

system, there are two key differences to consider when comparing these results. The first is the weather conditions in which the systems were tested: SMBAS used only vehicle sounds obtained under clear conditions, and was therefore not tested for robustness to adverse conditions, whereas the C-AVDI system has been tested under different weather conditions as outlined in Table 5 and Table 7. The second is the computational cost: as discussed in Section V-D our C-AVDI system runs approximately 2.8 times faster than Mod-SMBAS, the simplified version of our previous system.

Taking these factors into consideration makes the difference in performance between these two systems much less marked, as in clear conditions the C-AVDI system obtains an accuracy of 85 % while running approximately 2.8 times faster than Mod-SMBAS which presents an only marginally higher accuracy of 87 %. The similar accuracy score combined with a faster computation time when compared to a system of similar complexity (mono input signal, limited or no post-sampling processing, use of RF classifier), serves to demonstrate the viability of our proposed C-AVDI system, particularly in applications where low-cost, low-complexity sensing is required. On the other hand, the full implementation of SMBAS is better suited to applications where a high-cost, high-complexity system is required due to its superior accuracy performance.

There are two notable limitations in our proposed system that need to be addressed in any future work. The first limitation is the performance of our proposed C-AVDI system in adverse weather conditions (windy conditions in particular). This limitation can be mitigated by incorporating an additional passive or active noise cancellation stage inspired by our previous work [14] [15], by implementing CSP-based interference removal techniques as presented in [21], or by designing more complex bipolar sequences for more precise filtering during signal acquisition. The second limitation is the performance of the MCU implementation of our C-AVDI system, which, as discussed in Section V-E, is limited by the available memory of the Teensy device. The design and use of a custom board with a suitable amount of memory could theoretically bring the accuracy of the MCU implementation of the system in line with that of the simulated system. The creation of an updated hardware implementation of our C-AVDI system containing all the changes outlined above is also a subject of future work.

The obtained M values listed in Table 2 and Table 6 would suggest that in our particular C-AVDI application, in which we perform classification without reconstruction, the requirements on the lower bound value of M can be relaxed. Indeed, we can observe an inconsistency between the two values of M with regard to the rest of their respective system parameters. According to (11), we would expect the value of M listed in Table 2 to be smaller than the value listed in Table 6; however, in our case, the opposite is true. Given that the purpose of the system described by the parameters in Table 2 is to minimize the RE, and that the purpose of the system described by the parameters in Table 6 is to maximize the classification accuracy, their respective values of M would seem to indicate that a smaller number of $y[m]$ measurements is required for optimal classification than for optimal reconstruction. A more rigorous investigation of this phenomenon is left as a topic for future research.

Finally, the concepts underpinning C-AVDI presented in this paper can be used to extend the benefits of sub-Nyquist sampling to a wide variety of different acoustic (smart device wake-up-word detection) and non-acoustic (human activity recognition using devices such as smartwatches) sensing applications, further contributing to the democratization of ITS technologies.

VII. CONCLUSION

This paper presented C-AVDI, a method to detect and identify vehicles from their audio signatures at sub-Nyquist rates using features extracted from the compressive measurements obtained by a modified random demodulator. Our system uses Markov chain-generated spectrally shaped bipolar sequences to target a specific frequency band in the input signal during the sampling process itself.

The experimental evaluation of a simulated version of our system under a range of weather conditions produced a classification accuracy of 80 % for a back-end ADC sample rate of 3 kHz, with a runtime approximately 2.8 times quicker than the frequency-domain feature-based method proposed in our previous paper [2]. Evaluation of the system's MCU implementation produced an accuracy of 72 %.

Future work includes improving system performance by reducing the misclassification errors caused by adverse weather conditions, as well as improving the performance of the system's MCU implementation by increasing both the number and complexity of the classifiers used in the detection and identification process, and working towards a full hardware implementation of the proposed C-AVDI system.

ACKNOWLEDGMENT

An earlier version of this paper was presented at IEEE VTC2020-Fall [1] [DOI: 10.1109/VTC2020-Fall49728.2020.9348569].

REFERENCES

[1] B. Dawton, S. Ishida, Y. Hori, M. Uchino, and Y. Arakawa, ''Proposal for a compressive measurement-based acoustic vehicle detection and identification system,'' in *Proc. IEEE 92nd Veh. Technol. Conf. (VTC-Fall)*, Nov. 2020, pp. 1–6.

[2] B. Dawton, S. Ishida, Y. Hori, M. Uchino, Y. Arakawa, S. Tagashira, and A. Fukuda, ''Initial evaluation of vehicle type identification using roadside stereo microphones,'' in *Proc. IEEE Sensors Appl. Symp. (SAS)*, Mar. 2020, pp. 1–6.

[3] H. Göksu, ''Vehicle speed measurement by on-board acoustic signal processing,'' *Meas. Control*, vol. 51, nos. 5–6, pp. 138–149, 2018.

[4] H. Göksu, ''Engine speed–independent acoustic signature for vehicles,'' *Meas. Control*, vol. 51, nos. 3–4, pp. 94–103, Apr. 2018, doi: 10.1177/0020294018769080.

[5] J. George, L. Mary, and K. S. Riyas, ''Vehicle detection and classification from acoustic signal using ANN and KNN,'' in *Proc. Int. Conf. Control Commun. Comput. (ICCC)*, Dec. 2013, pp. 436–439.

[6] A. Aljaafreh and L. Dong, ''An evaluation of feature extraction methods for vehicle classification based on acoustic signals,'' in *Proc. Int. Conf. Netw., Sens. Control (ICNSC)*, Apr. 2010, pp. 570–575.

[7] Y. Schulz, A. K. Mattar, T. M. Hehn, and J. F. P. Kooij, ''Hearing what you cannot see: Acoustic vehicle detection around corners,'' *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 2587–2594, Apr. 2021.

[8] E. Alexandre, L. Cuadra, S. Salcedo-Sanz, A. Pastor-Sánchez, and C. Casanova-Mateo, ''Hybridizing extreme learning machines and genetic algorithms to select acoustic features in vehicle classification applications,'' *Neurocomputing*, vol. 152, pp. 58–68, Mar. 2015, doi: 10.1016/j.neucom.2014.11.019.

[9] H. Chen and Z. Zhang, ''Hybrid neural network based on novel audio feature for vehicle type identification,'' *Sci. Rep.*, vol. 11, no. 1, pp. 1–10, Dec. 2021.

[10] L. Becker, A. Nelus, J. Gauer, L. Rudolph, and R. Martin, ''Audio feature extraction for vehicle engine noise classification,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 711–715.

[11] S. Ishida, J. Kajimura, M. Uchino, S. Tagashira, and A. Fukuda, ''SAVeD: Acoustic vehicle detector with speed estimation capable of sequential vehicle detection,'' in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 906–912.

[12] M. Uchino, B. Dawton, Y. Hori, S. Ishida, S. Tagashira, Y. Arakawa, and A. Fukuda, ''Initial design of two-stage acoustic vehicle detection system for high traffic roads,'' in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2020, pp. 1–6, doi: 10.1109/PerComWorkshops48775.2020.9156248.

[13] K. Kubo, C. Li, S. Ishida, S. Tagashira, and A. Fukuda, ''Design of ultra low power vehicle detector utilizing discrete wavelet transform,'' in *Proc. ITS AP Forum*, May 2018, pp. 1052–1063.

[14] M. Uchino, S. Ishida, K. Kubo, S. Tagashira, and A. Fukuda, ''Initial design of acoustic vehicle detector with wind noise suppressor,'' in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2019, pp. 814–819.

[15] S. Ishida, M. Uchino, C. Li, S. Tagashira, and A. Fukuda, ''Design of acoustic vehicle detector with steady-noise suppression,'' in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 2848–2853.

[16] E. J. Candès, J. Romberg, and T. Tao, ''Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,'' *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[17] D. L. Donoho, ''Compressed sensing,'' *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[18] P. Prince, A. Hill, E. Piña Covarrubias, P. Doncaster, J. Snaddon, and A. Rogers, ''Deploying acoustic detection algorithms on low-cost, open-source acoustic sensors for environmental monitoring,'' *Sensors*, vol. 19, no. 3, p. 553, Jan. 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/3/553

[19] A. Fukuda, K. Hisazumi, T. Mine, S. Ishida, T. Ando, S. Ishibashi, S. Tagashira, K. Kaneko, Y. Arakawa, W. Kong, and G. Li, ''Toward sustainable smart mobility information infrastructure platform: Project overview,'' in *New Trends in E-Service and Smart Computing, Studies in Computational Intelligence*, vol. 742, T. Matsuo, T. Mine, and S. Hirokawa, Eds. Cham, Switzerland: Springer, Jan. 2018, pp. 35–46.

[20] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, ''Green AI,'' *Commun. (ACM)*, vol. 63, no. 12, pp. 54–63, 2020, doi: 10.1145/3381831.

[21] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, ''Signal processing with compressive measurements,'' *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 445–460, Apr. 2010.

[22] M. Mishali, Y. C. Eldar, O. Dounaevsky, and E. Shoshan, ''Xampling: Analog to digital at sub-Nyquist rates,'' *IET Circuits, Devices Syst.*, vol. 5, no. 1, pp. 8–20, Jan. 2011, doi: 10.1049/IET-CDS.2010.0147.

[23] A. Harms, W. U. Bajwa, and R. Calderbank, ''A constrained random demodulator for sub-Nyquist sampling,'' *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 707–723, Feb. 2013, doi: 10.1109/TSP.2012.2231077.

[24] D. Mo and M. F. Duarte, ''Design of spectrally shaped binary sequences via randomized convex relaxations,'' in *Proc. 49th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2015, pp. 164–168.

[25] E. J. Candès and T. Tao, ''Decoding by linear programming,'' *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[26] E. Candes and M. Wakin, ''Introduction to compressive sampling,'' *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[27] M. Rani, S. Dhok, and R. Deshmukh, ''A systematic review of compressive sensing: Concepts, implementations and applications,'' *IEEE Access*, vol. 6, pp. 4875–4894, 2018.

[28] T. Wimalajeewa and P. K. Varshney, ''Compressive sensing based classification in the presence of intra-and inter-signal correlation,'' *IEEE Signal Process. Lett.*, vol. 25, no. 9, pp. 1398–1402, Sep. 2018.

[29] X. Liu, E. Gonultas, and C. Studer, ''Analog-to-feature (A2F) conversion for audio-event classification,'' in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 2275–2279.

[30] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk, ''Analog-to-information conversion via random demodulation,'' in *Proc. IEEE Dallas/CAS Workshop Design, Appl., Integr. Softw.*, Oct. 2006, pp. 71–74.

[31] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, ''Theory and implementation of an analog-to-information converter using random demodulation,'' in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 1959–1962.

[32] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, ''Beyond Nyquist: Efficient sampling of sparse bandlimited signals,'' *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.

[33] T. Haque, R. T. Yazicigil, K. J.-L. Pan, J. Wright, and P. R. Kinget, ''Theory and design of a quadrature analog-to-information converter for energy-efficient wideband spectrum sensing,'' *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 2, pp. 527–535, Feb. 2015.

[34] M. A. Lexa, M. E. Davies, and J. S. Thompson, ''Reconciling compressive sampling systems for spectrally sparse continuous-time signals,'' *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 155–171, Jan. 2012.

[35] P. J. Pankiewicz, T. Arildsen, and T. Larsen, ''Sensitivity of the random demodulation framework to filter tolerances,'' in *Proc. 19th Eur. Signal Process. Conf.*, Aug. 2011, pp. 534–538.

[36] J. Zhang, N. Fu, W. Yu, and X. Peng, "Analysis of non-idealities of low-pass filter in random demodulator," in *Proc. 8th Int. Symp. Precis. Eng. Meas. Instrum.*, Jan. 2013, pp. 428–435, doi: 10.1117/12.2015006.

[37] W. Xu, Y. Cui, Y. Wang, S. Wang, and J. Lin, "A hardware implementation of random demodulation analog-to-information converter," *IEICE Electron. Exp.*, vol. 13, no. 16, 2016, Art. no. 20160465.

[38] R. Baraniuk, M. A. Davenport, M. F. Duarte, and C. Hegde, *An Introduction to Compressive Sensing*. Houston, TX, USA: OpenStax, Apr. 2011. [Online]. Available: https://legacy.cnx.org/content/col11133/latest/

[39] A. Harms, W. U. Bajway, and R. Calderbankz, "Shaping the power spectra of bipolar sequences with application to sub-nyquist sampling," in *Proc. 5th IEEE Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2013, pp. 236–239, doi: 10.1109/CAMSAP.2013.6714051.

[40] E. van den Berg and M. P. Friedlander. (Dec. 2019). *SPGL1: A Solver for Large-Scale Sparse Reconstruction*. [Online]. Available: https://friedlander.io/spgl1

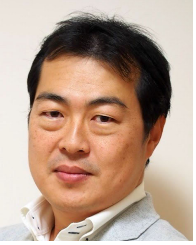
[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[42] J. Nordby, "Emlearn: Machine learning inference engine for microcontrollers and embedded devices (0.8.2)," Zenodo, 2019. [Online]. Available: https://zenodo.org/record/2589394/export/hx, doi: 10.5281/zenodo.2589394.

BILLY DAWTON (Graduate Student Member, IEEE) received the B.E. degree in electrical and electronic engineering from the University of Sussex, Brighton, U.K., in 2014, and the M.Sc. degree in wireless and optical communications from University College London, London, U.K., in 2015. He is currently pursuing the Ph.D. degree with the Department of Information Science and Electrical Engineering, Kyushu University, Fukuoka, Japan. His current research interests include new sensing technologies for intelligent transportation system applications, in particular low-cost, low-complexity acoustic vehicle detection, and identification systems.

SHIGEMI ISHIDA (Member, IEEE) received the B.E. degree in electronic engineering from the Shibaura Institute of Technology, Tokyo, Japan, in 2006, and the M.Sc. degree in information science and the Ph.D. degree in electrical engineering from The University of Tokyo, Japan, in 2008 and 2012, respectively.

From 2008 to 2009, he worked as a programmer in industry. From 2011 to 2013, he was a Japan Society for the Promotion of Science (JSPS) Research Fellow. In 2013, he was a Visiting Scholar with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis. From 2013 to 2021, he was an Assistant Professor with the Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka, Japan. He is currently an Associate Professor with the Department of Media Architecture, School of Systems Information Science, Future University Hakodate, Hokkaido, Japan. His research interests include wireless sensor networks, low-power wireless communications, localization systems, cross-technology communications, and intelligent transportation systems. He is currently focused on localization systems and sensing technologies for better understanding of the world around us.



YUTAKA ARAKAWA (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Keio University, Japan, in 2001, 2003, and 2006, respectively. From 2006 to 2013, he was an Assistant Professor at Keio University and Kyushu University, and from 2013 to 2019, he was an Associate Professor at the Nara Institute of Science and Technology. He also worked as a Visiting Researcher at ENSEEIHT, France, in 2011; DFKI, Germany, from 2012 to 2013; and UCLA, USA, from 2017 to 2018. Since 2019, he has been a Professor at the Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University. His current research interests include human activity recognition, behavior change support systems, and location-based information systems. He is a member of ACM, IPSJ, and IEICE. He won second place in the App Contest of ACM MobiCom 2014, the Best Demo Award both in ACM Ubicomp/ISWC2016 and IEEE PerCom2019, and IPSJ/IEEE-CS Young Computer Researcher Award in 2019.