

South Dakota State University

Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange

Electronic Theses and Dissertations

2018

Enabling Low Cost WIFI-Based Traffic Monitoring System Using Deep Learning

Sayan Sahu

South Dakota State University

Follow this and additional works at: <https://openprairie.sdstate.edu/etd>



Part of the [Digital Communications and Networking Commons](#), and the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Sahu, Sayan, "Enabling Low Cost WIFI-Based Traffic Monitoring System Using Deep Learning" (2018). *Electronic Theses and Dissertations*. 2688.
<https://openprairie.sdstate.edu/etd/2688>

This Thesis - Open Access is brought to you for free and open access by Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. For more information, please contact michael.biondo@sdstate.edu.

ENABLING LOW COST WIFI-BASED TRAFFIC MONITORING SYSTEM USING
DEEP LEARNING

BY
SAYAN SAHU

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Computer Science


South Dakota State University


2018


ENABLING LOW COST WIFI-BASED TRAFFIC MONITORING SYSTEM USING
DEEP LEARNING

SAYAN SAHU

This thesis is approved as a credible and independent investigation by a candidate for the Master of Science degree with major in Computer Science and is acceptable for meeting the thesis requirements for this degree. Acceptance of this thesis does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department.

 Myounggyu Won, Ph.D. _____ Date
Thesis Advisor

 George Hamer, Ph.D. _____ Date
Acting Head, EECS Department

 _____ Date
Dean, Graduate School

ACKNOWLEDGMENTS

I cannot express enough thanks to my advisor Dr. Myounggyu Won of Department of Computer Science at South Dakota State University. Dr. Won always encouraged me and gave me the advice to increase the quality of my thesis. He consistently allowed my research to be my own work while continuously steering me in the right direction.

I would also like to thank the committee members for my thesis especially Dr. Sung Shin and Dr. Alireza Salehnia who gave me valuable feedback during my preliminary presentation. Their constructive comments substantially helped me in improving the quality of my thesis. I am also grateful for Dr. Suvabrata Chakravarty for taking his time to review my thesis and provide valuable feedback during my final oral exam. Also, I would like to thank my parents. They always support me to complete learning to get Master degree.

Finally, I would like to thank the Department of Electrical Engineering and Computer Science and all the graduate faculty of the department for their continuous support and encouragement I have received during the course of my degree program. The courses I took from them helped me greatly to learn about fundamental materials that made my research activities possible.

TABLE OF CONTENTS

LIST OF TABLES	vi
LIST OF FIGURES	vii
ABBREVIATIONS	viii
ABSTRACT	x
Chapters	1
1 Introduction	1
1.1 Motivation	1
1.2 State of the Art	2
1.3 Proposed Work	3
1.4 Key Contributions	5
1.5 Thesis Organization	6
2 Related Work	7
3 Preliminaries and Problem Statement	12
3.1 WiFi Channel State Information	12
3.2 Problem Statement	14
4 Proposed System	16
4.1 System Overview	16
4.2 CSI Data Processing	17
4.2.1 Low Pass Filtering	17
4.2.2 PCA-Based Denoising	18
4.2.3 Phase Preprocessing	21

4.3	Vehicle Detection	23
4.4	Vehicle Classification	25
4.4.1	Convolutional Layer	26
4.4.2	Batch Normalization Layer	26
4.4.3	ReLU Layer	26
4.4.4	Max Pooling Layer	27
4.4.5	Dropout and Fully Connected Layer	27
4.4.6	Fully Connected Layer	27
5	Experimental Results	28
5.1	Experimental Setup	28
5.2	Detection Accuracy	31
5.3	Classification Accuracy	31
5.4	Classification Accuracy Per Lane	33
6	Conclusion	35
	Bibliography	36

LIST OF TABLES

2.1	Vehicle Classification Approaches	10
3.1	Notations	14
5.1	Vehicle types and number of samples	29
5.2	Hyper parameters for deep learning	30
5.3	Classification accuracy - SVM	31
5.4	Classification accuracy - kNN	32
5.5	Classification accuracy - CNN	32
5.6	Classification accuracy - Lane 1	33
5.7	Classification accuracy - Lane 2	33
5.8	Classification accuracy - Mixed Lane	34

LIST OF FIGURES

3.1	CSI amplitudes and phases of passing vehicles.	12
4.1	System architecture of DeepWiTraffic.	16
4.2	CSI amplitude values of passing car (raw vs filtered).	17
4.3	Filtered WiFi CSI stream for subcarriers #1, #2, and #3 showing high correlations.	19
4.4	PCA #1 that represents all 30 CSI streams vs. a CSI stream	21
4.5	Raw and processed phase data for a passing vehicle.	22
4.6	CNN architecture.	25
5.1	Experimental setting.	28

ABBREVIATIONS

CCS	Continuous Counting Station
CNN	Convolutional Neural Network
COT	Commonwealth Office of Technology
CSI	Channel State Information
DeepWiTraffic	Deep Learning
DOT	Department of Transportation
FHA	Federal Highway Administration
FHWA	United States Federal Highway Administration
ITS	Intelligent Transportation Systems
kNN	k-Nearest Neighbor
LIDAR	Laser Infrared Detection And Ranging
LoS	Line of Sight
MAD	Median Absolute Deviation
NIC	Network Interface Controller
OFDM	Orthogonal Frequency Division Multiplexing
PCA	Principal Component Analysis
ReLU	Rectified Linear Unit
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
RX	Receiver antenna

S,M,L	Small, Medium, Large
STD	Normalized Standard Deviation
SVM	Support Vector Machine
TMS	Traffic monitoring system
TX	Transmitter antenna
UAV	Unmanned Aerial Vehicle

ABSTRACT

ENABLING LOW COST WIFI-BASED TRAFFIC MONITORING SYSTEM USING
DEEP LEARNING

SAYAN SAHU

2018

A traffic monitoring system (TMS) is an integral part of Intelligent Transportation Systems (ITS) for traffic analysis and planning. However, covering huge miles of rural highways (119,247 miles in U.S.) with a large number of TMSs is a very challenging problem due to the cost issue. This paper aims to address the problem by developing a low-cost and portable TMS called *DeepWiTraffic* based on COTs WiFi devices. The proposed system enables accurate vehicle detection (counting) and classification by exploiting the unique WiFi Channel State Information (CSI) of passing vehicles. Spatial and temporal correlations of CSI amplitude and phase data are identified and analyzed using a deep learning technique to classify a vehicle into five different types: motorcycle, passenger vehicle, SUV, pickup truck, and large truck (a vehicle with more than three axles according to the FHWA classification). The principal component analysis (PCA) technique is exploited to reduce the dimension of the subcarriers and remove the device specific noise. The CSI phase data of a received signal are preprocessed by applying a linear transformation and the correlations of CSI phase information of multiple subcarriers are taken into account for effective vehicle classification. A convolutional neural network (CNN) is designed to extract optimal features of the preprocessed CSI

amplitude and phase data. A huge amount of CSI data of passing vehicles as well as ground truth video data are collected for about 120 hours to validate the performance of the proposed proof-of-concept system. The results show that the average detection accuracy of 99.4%, and the average classification accuracy of 91.1% (Motorcycle: 97.2%, Passenger Car: 91.1%, SUV:83.8%, Pickup Truck: 83.3%, and Large Truck: 99.7%) can be achieved with a very small cost of less than \$1,000.

Chapter 1

Introduction

1.1 Motivation

A traffic monitoring system (TMS) is an important component of Intelligent Transportation Systems (ITS) for improved safety and efficiency of transportation. TMSs are deployed to collect traffic data that characterize performance of a roadway system. Different traffic parameters are measured such as the number of vehicles, vehicle density, vehicle speed, and vehicle class [32]. These traffic parameters are essential information in analyzing transportation systems and estimating future transportation needs [33]. For example, TMSs have played a key role in supporting decision making process for road improvement plans, accessing the road network efficiency, and analyzing economic benefits, *etc.* [23].

The Department of Transportation (DOT) in each state is charged by the United States Federal Highway Administration (FHWA) to collect traffic information about vehicles traveling state and federal highways and roadways to improve the safety and efficiency [2]. As such, state highway and transportation agencies operate TMSs to perform vehicle counting, vehicle classification, and vehicle weight measurement. These traffic monitoring systems are either temporary or permanent [5]. Temporary stations typically operate less than a full year while permanent ones perform traffic monitoring on a continuous basis. There are 7,430 traffic monitoring stations under operation in the U.S. as of August 2015 [5].

One of the critical issues for DOTs is that they do not have enough TMSs to cover the huge land area of U.S. especially considering the huge miles of rural highways of about 119,247 miles. The main reason is the high cost [4]. According to the Georgia DOT, the minimum cost to install a continuous counting station (CCS) on a two-lane rural roadway is about \$25,000 [1]. 365 day vehicle classification on a two-lane rural roadway is more expensive costing about \$35,770 [4]. This paper aims at alleviating this cost problem by developing a significantly low-cost, portable, and innovative TMS based on WiFi channel state information (CSI) and deep learning.

1.2 State of the Art

Vehicle detection and classification techniques are largely categorized into three types: intrusive, non-intrusive, and off-roadway [8]. Intrusive solutions embed sensors such as magnetic detectors [49], vibration sensors [40], and inductive loops [20] in the pavement of roadway using a sawcut or hole. Non-intrusive approaches mount sensors like magnetic sensors [50], acoustic sensors [15], and LIDAR (Laser Infrared Detection And Ranging) [27] either on roadsides or over the road. Among these sensors, camera based solutions are very widely used [36]. Off-roadway solutions use mobile sensors that are equipped with UAVs [21] or satellite systems [24]. Detailed and comprehensive discussion of existing technologies is presented in Section 2.

Intrusive approaches are known to be the most expensive option mainly due to significantly high cost for installation and maintenance, especially for traffic disruption and lane closure to assure security of road workers. Furthermore, effectiveness of these

embedded sensors is easily affected by the condition of the pavement and often generates unreliable results. DOTs are increasingly adopting non-intrusive solutions. A widely adopted sensor in this category is a camera. However, it has been reported that the performance is degraded when vision obstructions are present and even more severely in adverse weather conditions. Furthermore, cameras are expensive especially because of the installation cost since they must be fixed at a certain mounting height for optimum performance. Other sensors for non-intrusive solutions such as magnetic sensors and acoustic sensors require precise calibration of sensor direction and placement, thus not appropriate for general and ad hoc deployment [37]. Although some sensors such as LIDAR guarantee very good performance, those sensors are extremely expensive. Thanks to recent advances in UAV technologies, off-road-based approaches are receiving greater attention. However, these solutions suffer from spatial and temporal limitations. Specifically, the operation time of a UAV is limited due to the limited flight time, and satellites are not always available.

1.3 Proposed Work

This paper proposes a portable, non-intrusive, yet inexpensive TMS called DeepWiTraffic. The proposed TMS hinges on unique wireless channel characteristics created by passing vehicles. It utilizes WiFi channel state information (CSI) of a received packet to extract rich information about the channel properties. A deep learning technique is used to effectively capture the unique features from CSI data, more specifically from CSI amplitude and phase data, to train a vehicle classification model that categorizes passing

vehicles into different car types. More precisely, given a WiFi transmitter and a receiver deployed on each side of a road respectively, spatial correlations of multiple subcarriers received via different receiver antennas (three receiver antennas in our system) are analyzed, and distinctive features are extracted. The time domain is considered as well, *i.e.*, temporal correlations of the time series of CSI amplitude and CSI phase data are characterized using a machine learning technique for effective vehicle classification.

Numerous techniques are applied to maximize the vehicle classification accuracy. Environment noise in CSI amplitude data caused by surrounding obstacles and low-speed moving objects, *e.g.*, people moving around is effectively removed. Principal component analysis (PCA) is then exploited to reduce the dimension of multiple subcarriers (in our experiments, 30 subcarriers for each TX and RX pair) down to one, significantly reducing the computational overhead. Random noise of CSI phase data is effectively cleared based on a linear transformation. These noise-removed and low-dimensional CSI data are then used for vehicle detection (counting) and classification. A novel vehicle detection algorithm is developed based on a simple threshold-based mechanism. This algorithm shows near 100% detection accuracy. Once a vehicle is detected, vehicle classification is performed. For vehicle classification, a convolutional neural network (CNN) is designed to identify, extract optimal features from CSI data, and to maximize the classification accuracy. CNN is selected due to the large input size. Specifically, the correlations of the time series of CSI amplitude and phase values are taken into account by aggregating them as a single input image of size $6 \times 2,500$.

We have collected huge amounts of CSI data for about 120 hours over a month.

The video data were obtained that are synchronized with the CSI data. The video data were used as ground truth, *i.e.*, the vehicle type of a passing vehicle was manually tagged with the corresponding CSI data. DeepWiTraffic detects and classifies vehicles into five different types: motorcycle, passenger vehicle, SUV, pickup truck, and large truck. Nontrivial efforts of repeating experiments with different combinations of hyper parameters were performed to find the best classification accuracy. The average vehicle classification accuracy was 91.1%.

1.4 Key Contributions

The contributions of this paper are summarized as follows:

- The first WiFi-based traffic monitoring system using deep learning is proposed.
- Spatial and temporal aspects of both CSI amplitude and phase data are taken into account to improve classification performance.
- An effective CNN model is trained for effective vehicle classification performance.
- Huge amounts of CSI data as well as ground truth video data are collected.
- Extensive experiments are conducted based on real world CSI data to compare the performance of DeepWiTraffic with other classifiers including support vector machine (SVM) and k-nearest neighbor (kNN).

1.5 Thesis Organization

This paper is organized as follows. In Chapter 2, we review the literature on intelligent traffic monitoring systems. Background of WiFi CSI is discussed and the problem is defined in Chapter 3. In Chapter 4, followed by the system overview, we provide the details of the system design. We then present the experimental setup and analyze the experimental results in Chapter 5. Finally, we conclude in Chapter 6.

Chapter 2

Related Work

Vehicle detection (counting) and vehicle classification are the key functionalities of traffic monitoring systems [32]. The literature shows that vehicle detection can be done with very high accuracy. However, the performance of vehicle classification techniques varies substantially. This section presents a comprehensive review on TMSs concentrating on vehicle classification mechanisms.

Vehicle classification methods are divided largely into three categories: intrusive, non-intrusive, and off-roadway approaches. Table 2 summarizes the properties of existing vehicle classification schemes including sensor types, vehicle types for classification, classification accuracy, and the cost. As can be seen, it is difficult to directly compare the performance of different approaches because they are designed to classify vehicles into different types. As such, this section is focused on drawing meaningful insights by covering the literature comprehensively and providing general guidelines to readers for selecting appropriate TMS.

A common property of intrusive solutions is that sensors (*e.g.*, piezoelectric sensors [38], magnetometers [10][49], vibration sensors [40], loop detectors [32]) are installed on a roadway. As Table I indicates, intrusive approaches are capable of classifying a large selection of vehicle types with high classification accuracy leveraging close contact with passing vehicles that allow for collecting precise sensor data. The main concern of these solutions, however, is the cost issue. Especially when sensors are installed under the pavement on roads, the cost increases prohibitively. The maintenance cost is nonnegli-

gible as it incurs extra cost for constructor safety assurance.

Due to the high cost of intrusive solutions, non-intrusive approaches have received a lot of attention. A typical characteristic of these solutions is that sensors are deployed on a roadside obviating the construction and maintenance cost for intrusive solutions. A most widely adopted sensor for non-intrusive solutions is a camera [11][9][19]. Significant advances in imaging technologies and image processing techniques based on machine learning algorithms gave a birth to precise camera-based TMSs [34]. As Table I shows, the classification accuracy of camera-based TMSs is very high. However, achieving high classification accuracy is still challenging at night, under severe weather conditions, and when there are obstacles that obstruct the clear view. There are other sensors such as magnetometers [41][12][22], accelerometers [31], and acoustic sensors [15][35][13] that have been used in non-intrusive TMSs. Table I shows that these sensors quite impressive classification accuracy. However, the low fidelity information that these sensors requires strategic positioning of multiple of those sensors. As such, minor errors in positioning or adjusting sensing directions may increase classification errors. To address the drawbacks of these sensors, more advanced sensors such as LIDAR (Laser Infrared Detection And Ranging), and infrared sensors can be considered [14][16][36]. While these advanced technologies allow for very high classification accuracy, it is possible only at an extremely high cost.

Off-roadway solutions utilize cameras mounted on UAVs [42] or satellites [7] for vehicle classification. As shown in Table I, the classification accuracy of off-roadway approaches is not quite impressive (note but Liu *et al.* achieved 98.2% accuracy, but

this accuracy is for only two vehicle types: cars and trucks). The low classification accuracy of off-roadway solutions is attributed to the small image size. However, off-roadway approaches are appropriate when the user needs to cover a large area.

Recently a fundamentally new non-intrusive approach using wireless signals has been proposed. Haferkamp *et al.* exploited multiple pairs of RF (radio frequency) transceivers to develop a TMS [17]. The key intuition of their system is that different types of vehicles, when passing the line of sight (LoS) between a pair of RF transceivers, result in unique received signal strength (RSSI) patterns. However, since RSSI represents only a single dimensional information (*i.e.*, signal strength for a single channel), it is challenging to correlate effectively the vehicle body shape with the resulting signal strength. To overcome this difficulty, multiple RF transceivers are necessary. In contrast, WiFi CSI data contain much richer information conveyed in 30 subcarriers for each pair of TX-RX antennas allowing us to perform vehicle classification more effectively.

Table 2.1: Vehicle Classification Approaches

Classification	Approach	Publication	Vehicle Class	Cost	Accuracy	
Intrusive	Piezoelectric sensor	Rajab [38]	motorcycles, passenger vehicles, two axle for tire unit, buses, two axles six tire single units, three axles single units, four or more axles single unit, four axles single trailer, five axles single trailer, seven axles single trailer, seven or more axles multi-trailer	medium	86.9%	
	Magnetometer	Bottero [10]	car, van, truck	medium	88.0%	
		Xu [49]	hatchbacks, sedans, buses, and multi-purpose vehicles	medium	95.4%	
	Loop Detector		Meta [32]	car, van, truck, bus, motorcycle	high	94.2%
			Jeng [20]	motorcycles, passenger cars, other two-axle four-tire single unit vehicles, buses, two-axle, six-tire single-unit trucks, tree-axle single-unit trucks, four or more axle single-unit trucks, four or fewer axle single-trailer trucks, five-axle single-trailer trucks, six or more axle single-trailer trucks, five or fewer axle multi-trailer trucks, six-axle multi-trailer trucks, seven or more axle multi-trailer trucks	high	92.4%

Classification	Approach	Publication	Vehicle Class	Cost	Accuracy
Non-intrusive	Camera	Chen [11]	car, van, bus, motorcycle	medium/high	94.6%
		Bautista [9]	jeep, sedan, truck, bus, SUV, and van	medium/high	96.4%
	Infrared + ultrasonic sensors	Odat [36]	sedan, pickup truck, SUV, bus, two wheeler	low/medium	99.0%
	Magnetic Sensors	Wang [43]	bicycle, car, minibus	low/medium	93.0%
		Yang [50]	motorcycle, passenger car (two-box and saloon), SUV, bus	low/medium	93.6%
	Acoustic Sensors	George [15]	heavy (truck bus), medium (car, jeep, van), light (auto rickshaws, two wheelers)	low/medium	73.4%
	LIDAR	Lee [26][27]	motorcycle, passenger vehicle, passenger vehicle pulling a trailer, single-unit truck, single-unit truck pulling a trailer, and multi-unit truck	very high	99.5%
Off-roadway	UAV	Liu [30]	car, truck	medium	98.2%
		Tang [42]	seven vehicle types, such as car, truck, bus, etc. (specific type not specified)	medium	78.7%
	Satellites	Audebert [7]	pick up, van, truck, car	high	80.0%

Chapter 3

Preliminaries and Problem Statement

This section presents basic principles of WiFi CSI, followed by the problem statement.

3.1 WiFi Channel State Information

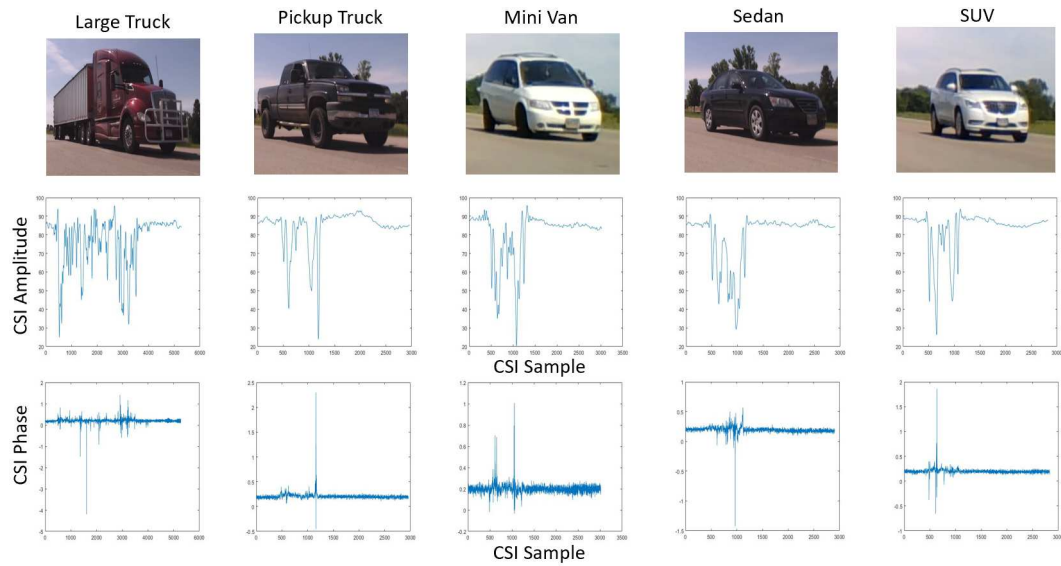


Figure 3.1: CSI amplitudes and phases of passing vehicles.

The orthogonal frequency division multiplexing (OFDM) modulation scheme is used to implement the physical layer of contemporary WiFi standards [18]. It is robust against the frequency selective fading since high data-rate stream is partitioned onto close-spaced subcarriers. The WiFi CSI represents the channel properties for these OFDM subcarriers, *e.g.*, a combined effect of fading, scattering, and power decay with distance. WiFi CSI has been successfully applied to numerous applications such as human activity recognition [6][52], traffic monitoring [47], and localization [45].

Formally, WiFi CSI represents the properties of the channel as follows [51].

$$y = H \cdot x + n. \quad (3.1)$$

Here x and y refer to the transmitted and received signal, respectively. n represents the channel noise. H is a $M \times N \times W$ matrix, where M , N , and W , are the number of receiver antennas, transmitter antennas, and subcarriers, respectively. Matrix H can be expressed as a vector of W subcarrier groups as follows.

$$H = [H_1, H_2, \dots, H_W]. \quad (3.2)$$

Here H_i is a $M \times N$ matrix that represents the channel state information values for the subcarrier group received via $M \times N$ different transmitter-receiver antenna pairs. A CSI value for the i -th subcarrier received via the transmitter n and the receiver m is denoted by CSI_{mn}^i , which is defined as follows.

$$CSI_{mn}^i = |h|e^{j\phi}. \quad (3.3)$$

This CSI value contains both the amplitude ($|h|$) and phase information (ϕ) of the i -th subcarrier signal received via the two antennas m and n . These are the two information that we utilize in this paper based on the observation that when a car passes, depending on the size and shape of the car, unique amplitude and phase patterns are generated (Figure 3.1).

Table 3.1: Notations

Notations	
N_C	the total number of passing cars
N	the total number of packets
M_{CSI}	a $N \times 30$ matrix that contains CSI values of 30 subcarriers for N packets
$A = \{a_1, \dots, a_N\}$	a set of CSI amplitude values extracted from M_{CSI}
$P = \{p_1, \dots, p_N\}$	a set of CSI phase values extracted from M_{CSI}
\hat{A}_i	a set of CSI amplitude values for i -th passing vehicle, $0 \leq i \leq N_C$, extracted from A
\hat{P}_i	a set of CSI phase values for i -th passing vehicle, $0 \leq i \leq N_C$, extracted from P
$A = \{\hat{A}_1, \dots, \hat{A}_{N_C}\}$	a collection of CSI amplitude sets for N_C passing vehicles
$P = \{\hat{P}_1, \dots, \hat{P}_{N_C}\}$	a collection of CSI phase sets for N_C passing vehicles
M	a convolutional neural network model for vehicle classification
A_d	vehicle detection algorithm
A_c	vehicle classification algorithm
Parameters	
δ_1	backward offset
δ_2	forward offset
ω	minimum inter vehicle distance for clear vehicle detection

3.2 Problem Statement

Let M_{CSI} denote a $N \times 30$ matrix that each element represents a CSI value for a certain TX-RX antenna pair. Specifically, M_{CSI} is a data structure that contains CSI values for N successively received packets (Note that each packet corresponds to 30 subcarriers). Also let N_C denote the total number of cars that passed while collecting the CSI data. We are tasked to classify N_C vehicles into five different types {bike, passenger car, SUV, pickup truck, large truck} given M_{CSI} . Note that if N_C is set to one, then it implements a real-time TMS.

The CSI amplitude and phase values are extracted from M_{CSI} . The CSI amplitude and phase values are denoted by a set $A = \{a_1, \dots, a_N\}$, and $P = \{p_1, \dots, p_N\}$, respectively. Note that each a_i (p_i) is an amplitude (phase) value that represents the amplitude (phase) values for 30 subcarriers. A technique to reduce the dimension of the CSI amplitude and phase is discussed in Section 4.2. We then extract from A (P) the CSI amplitude (phase) values denoted by \hat{A}_i (\hat{P}_i) that have been affected by a i th passing car using a vehicle detection algorithm (A_d). Now we can create the collections of \hat{A}_i and \hat{P}_i for all $0 \leq i \leq N_C$, which are denoted by $A = \{\hat{A}_1, \dots, \hat{A}_{N_C}\}$ and $P = \{\hat{P}_1, \dots, \hat{P}_{N_C}\}$, respectively. These A and P are provided as input to a convolutional neural network (CNN) either to train a model M (in training mode) or to classify an input instance consisting of \hat{A}_i and \hat{P}_i into five vehicle types {bike, passenger car, SUV, pickup truck, large truck} using the model. The algorithm used for this vehicle classification is denoted by (A_c).

Now the problem that we solve in this paper is concentrated on development of the two algorithms namely A_d and A_c . Specifically we target a typical two-lane rural highway. In subsequent sections, we will describe (1) the preprocessing methods to reduce the noise and dimension of raw CSI amplitude and phase data, (2) algorithms to extract the CSI amplitude and phase portions corresponding to a passing vehicle, (3) and design of a neural network for effective vehicle classification.

Chapter 4

Proposed System

4.1 System Overview

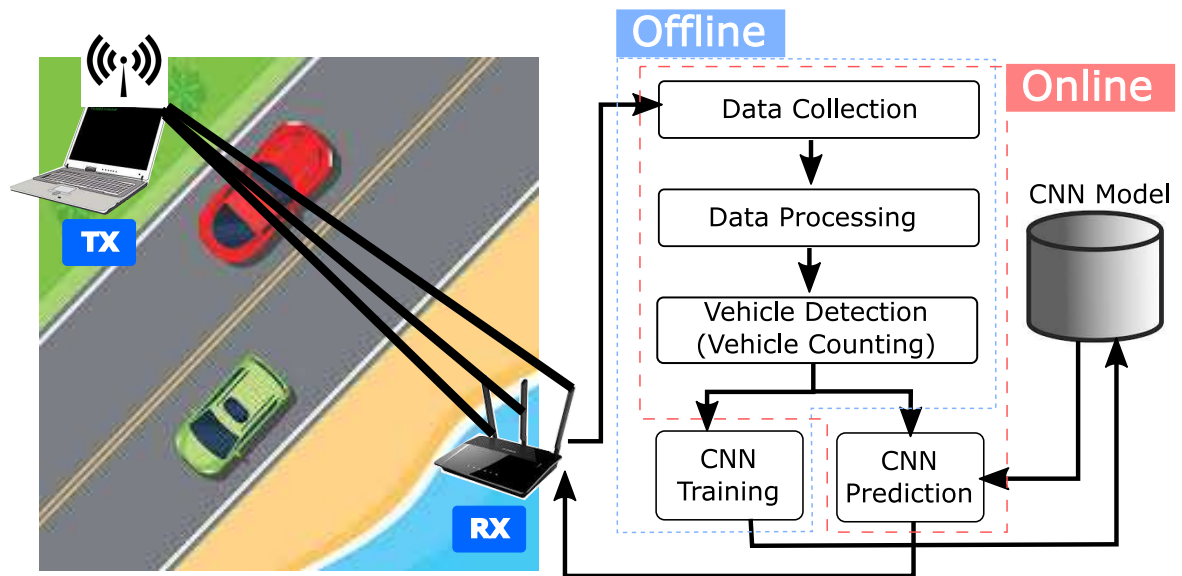


Figure 4.1: System architecture of DeepWiTraffic.

DeepWiTraffic consists of four system components, namely Data Collection, Data Processing, Vehicle Detection, Lane Detection, and Vehicle Classification. Figure 4.1 shows the system architecture of DeepWiTraffic. The data collection module receives N CSI packets and builds M_{CSI} . Note that in the training mode, N can be sufficiently large to collect CSI data for a large number of vehicles, while small N can be used for real-time vehicle classification in the online mode. Basically N controls the window size. The key roles of the data processing module are threefold: extraction of CSI amplitude values A and phase values P from M_{CSI} , noise reduction of A and P , and aggregation of CSI amplitude values for 30 subcarriers for faster processing. The lane detection

module implements A_d , *i.e.*, extracts corresponding CSI amplitude and phase values for a passing vehicle, generating A and P . The vehicle classification module consists of two parts: CNN Training and CNN Prediction. In the former part, the module trains a CNN model based on A and P , and the ground truth vehicle type typically input by the user. In the latter part, the module classifies the detected vehicle into five different types based on A and P .

4.2 CSI Data Processing

4.2.1 Low Pass Filtering

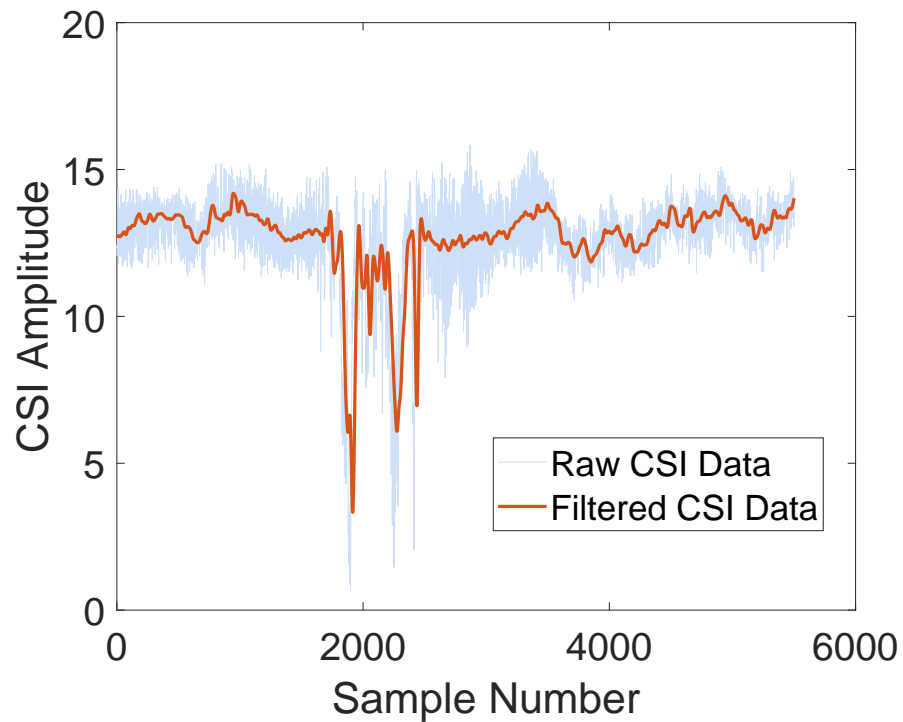


Figure 4.2: CSI amplitude values of passing car (raw vs filtered).

DeepWiTraffic is designed specifically to capture the CSI amplitudes of passing

vehicles. Thus, CSI amplitudes contributed by other slow moving objects, *e.g.*, minor human (mostly system operators) movements, are effectively cleared off. More precisely, we ensure that CSI amplitude fluctuations caused by any objects that move at a speed of less than 2m/s are excluded. The WiFi wavelength of our system that operates at 5.32GHz frequency bandwidth is 5.64cm [44]. With the wavelength of 5.64cm and the movement speed of 2m/s, the corresponding frequency component is calculated as 38Hz. Consequently, we apply a general low pass filter with a cutoff frequency of 38Hz to mitigate the impact of irrelevant objects on the received CSI amplitude data. Figure 4.2 shows an example of original CSI amplitude data of a passing car, and the results of low pass filtering.

4.2.2 PCA-Based Denoising

Environmental noise (*e.g.*, caused by slow moving objects) has been successfully mitigated by designing and applying a low pass filter. Another important source of performance degradation is noise caused by internal state transitions in a WiFi NIC which include changes in transmission power, adaptation of transmission rate, and CSI reference level changes [28]. Typically, burst noises in CSI data are caused by these internal state transitions. Ali *et al.* made an interesting observation that the effect of these burst noises is significantly correlated across CSI data streams of subcarriers [6].

The principal component analysis (PCA) is used to mitigate the burst noises by exploiting highly correlated CSI streams for different subcarriers. Figure 4.3 depicts an example showing that CSI streams for different subcarriers are highly correlated.

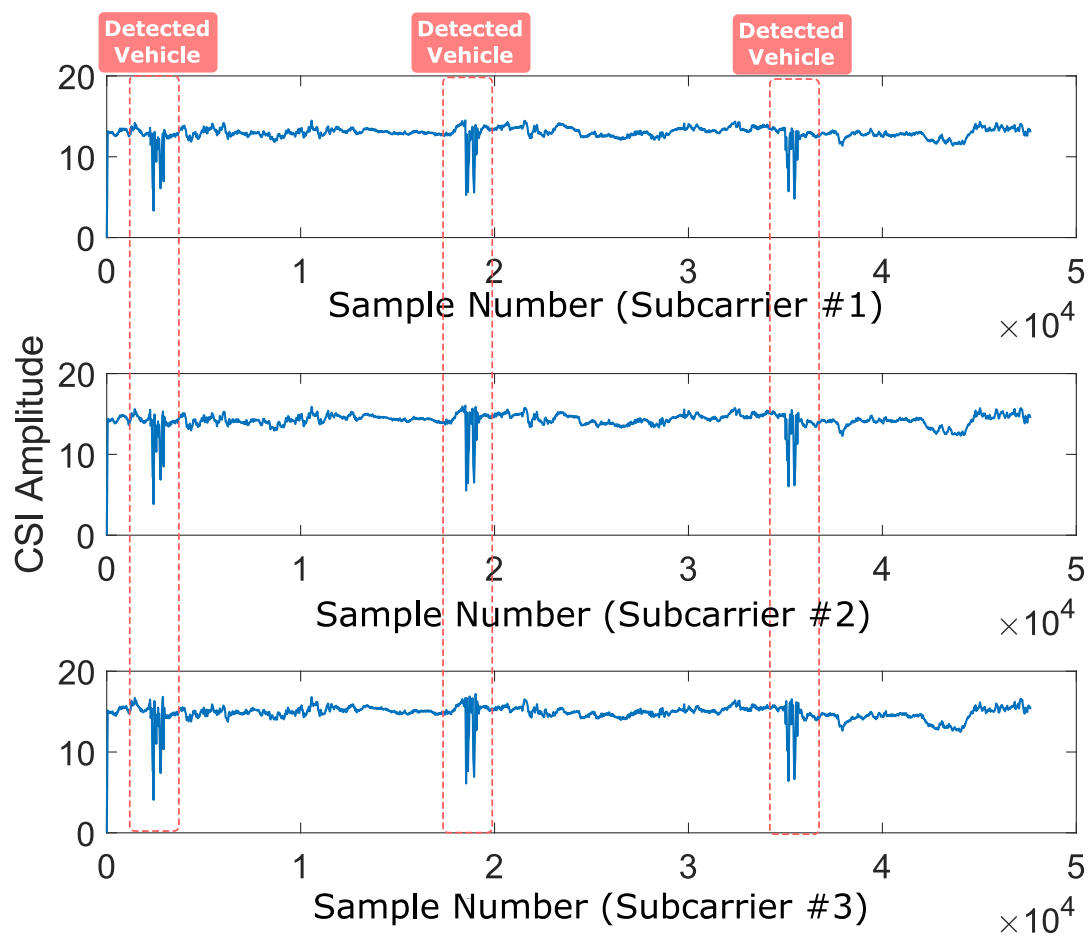


Figure 4.3: Filtered WiFi CSI stream for subcarriers #1, #2, and #3 showing high correlations.

The PCA is also used to reduce the dimension of CSI data for all 30 subcarriers down to one. More specifically, for each transmitted packet, we obtain 30 CSI amplitude values for 30 subcarriers. Using PCA, we analyze the correlations of these multi-dimensional CSI data, extract common features, and reduce the dimension to one. This noise and dimension reduction process is executed in four steps as follows.

Preprocessing of Sample: For each TX-RX antenna pair, define a $N \times 30$ matrix H that store CSI amplitude values for all 30 subcarriers and N received packets. A CSI stream (consisting of N CSI amplitude values) for each subcarrier is arranged in each column of matrix H . After construction of matrix H , the mean value of each column is calculated and subtracted from each column, which completes this step.

Computation of Covariance Matrix: the covariance matrix $H^T \times H$ is calculated in this step.

Computation of Eigenvalues and Eigenvectors of Covariance: Eigendecomposition of the covariance matrix $H^T \times H$ is performed to obtain the eigenvectors q ($30 \times k$).

Reconstruction of Signal: By projecting H onto the eigenvectors q ($30 \times k$), we obtain $h_i = H \times q_i$, where q_i is the i^{th} eigenvector and h_i is the i^{th} principal component.

Figure 4.4 shows the first PCA component compared with a filtered CSI stream. As shown, CSI amplitude values for passing vehicles are more clearly distinguished in the PCA component, which improves the performance of the proposed vehicle detection and classification algorithms.

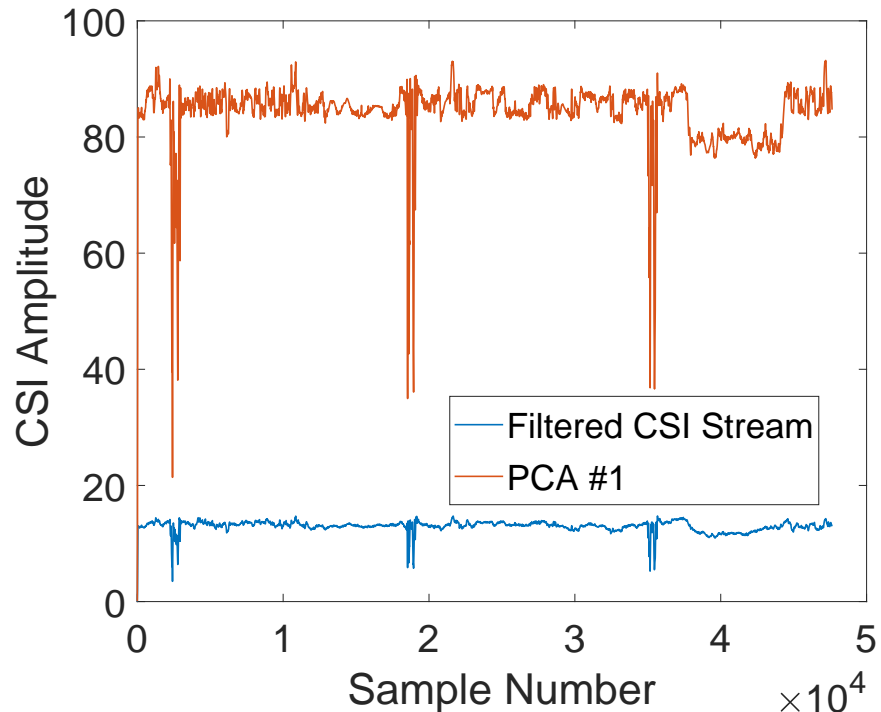


Figure 4.4: PCA #1 that represents all 30 CSI streams vs. a CSI stream

4.2.3 Phase Preprocessing

Since the phase information is one of the two primary features for vehicle classification, it is important to effectively mitigate the impact of random noises. This section presents a method to preprocess CSI phase data so that the random noises are reduced.

We can express the measured CSI phase of subcarrier c as the following [48].

$$\hat{\phi}_c = \phi_c - 2\pi \frac{k_c}{N} \alpha + \beta + Z. \quad (4.1)$$

Here ϕ_c is the original phase; k_c denotes the subcarrier index; N is the Fast Fourier Transform size (64 for IEEE 802.11 a/g/n); and Z is the measurement noise. Our objective is to remove α and β , which are the time lag and the phase offset at the re-

ceiver, respectively. We adopt a linear transformation to remove these noise factors [39].

Formally, we define the two variables e_1 and e_2 as follows.

$$e_1 = \frac{\hat{\phi}_F - \hat{\phi}_1}{2\pi F}, \quad (4.2)$$

$$e_2 = \frac{1}{F} \sum_{1 \leq c \leq F} \hat{\phi}_c, \quad (4.3)$$

Here F refers to the last subcarrier index. Note that $F = 30$ because we use the Intel 5300 NIC which exports 30 subcarriers. We then use a linear transformation: $\hat{\phi}_f - e_1 f - e_2$ to remove both the timing offset α and the phase offset β . We disregard the small measurement noise Z in this calculation.

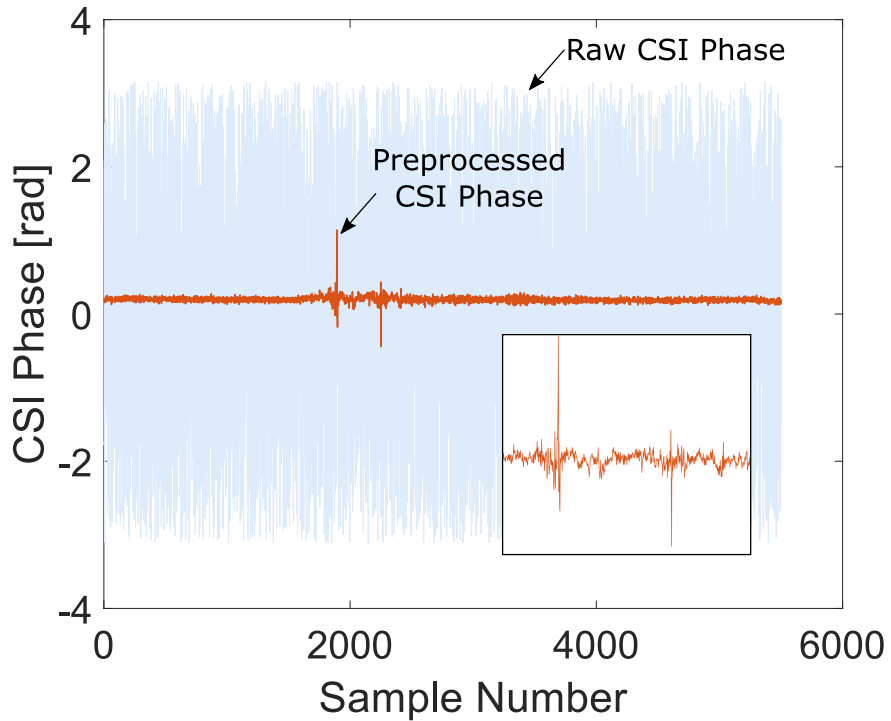


Figure 4.5: Raw and processed phase data for a passing vehicle.

Figure 4.5 shows both the raw and preprocessed CSI phase data (measured with sampling rates of 2,500 samples/sec). The result indicates that the algorithm successfully captures the time series of the CSI phase values by effectively reducing the random noises.

4.3 Vehicle Detection

Now we have noise-filtered CSI amplitude values A , and phase values P . The next task is to detect N_C passing vehicles and extract A and P from A and P (*i.e.*, extracting only the portions of CSI amplitude and phase values that are influenced by the passing vehicles). Detecting a passing vehicle is simple because it causes abrupt changes in CSI amplitude values. As such, we adopt a standard outlier detection technique based on scaled median absolute deviation (MAD): Scaled MAD = $c \cdot \text{median}(|A_i - \text{median}(A)|)$, $i = 1, 2, \dots, N$. $A = \{A_1, A_2, \dots, A_N\}$ is the set of collected CSI data samples, and $c = \frac{-1}{\sqrt{2} \cdot \zeta(3/2)}$, where ζ is the inverse complementary error function. A sample CSI amplitude value is considered as an outlier if it is more than three scaled MAD away from the mean, detecting a vehicle.

Once a vehicle(s) is detected, A and P are extracted. Since A and P are synchronized, outliers are found with A , but the result is applied to both A and P . More specifically, assume that an outlier is $a_i \in A$. Starting from a_i , the algorithm extracts the CSI amplitude samples in the range between $a_{i-\delta_1}$ and $a_{i+\delta_2}$. These δ_1 and δ_2 are system parameters. We use δ_2 to take into account the momentary CSI amplitude fluctuations after a vehicle passes through the line of sight (LoS) between the TX-RX antenna pair. δ_1 is used to capture the minor changes in CSI amplitude and phase values when the

Algorithm 1 CSI Data Extraction Algorithm

```

A and P A and P
O ← outlier(A).
i ← 1 |O|
O[i] && !r
s ← i.
f ← i.
r ← TRUE.
O[i] && r
f ← i.
!O[i] && r
i - f > ω
r ← FALSE.
(s - δ1) > 0 && (f + δ1) < |O|
A ← A[(s - δ1)...(f + δ1)].
P ← P[(s - δ1)...(f + δ1)].
continue.

```

vehicle is very close to the LoS but yet passed through it. In our experiments, we found that $\delta_1 = 500$ (0.25sec), and $\delta_2 = 500$ (0.25sec) gave the best results.

Algorithm 4.3 displays the pseudocode of the amplitude and phase extraction process. The function *outlier* finds the outlier samples and records the sample indices of the outliers in an array *O* (Line 2). The algorithm keeps track of the beginning *s* and end *f* of extracted amplitude and phase values, and a flag *r* is used to indicate that the extraction process is in progress so that the algorithm can finish when the extraction process is completed (Lines 4-7). In other words, the extraction process is continued as long as *r* is set to TRUE and the sample is considered as an outlier (Lines 8-9). If the sample is found to be a non-outlier, the interval between the current sample and the last valid outlier is calculated, and it is compared with the threshold ω (in our experiments, we used 1,250, *i.e.*, 0.5 second). Finally, if the interval is greater than ω , we reset the

flag r to FALSE to finish the extraction process and save the extracted CSI amplitudes and phases in A and P , respectively (Lines 10-15).

4.4 Vehicle Classification

We adopt the convolutional neural network (CNN) for vehicle classification. The correlations of the time series of CSI amplitude and phase values are taken into account by aggregating them as a single input image. Specifically, a $6 \times 2,500$ image is provided as input to CNN. The first three rows of the image represent the time series of extracted CSI amplitude values for three TX-RX antenna pairs (Note that there are 1 TX antenna, and 3 RX antennas). The subsequent three rows of the image are the time series of extracted CSI phase values. These 6 CSI data sequences are exactly aligned in the image to enable CNN extract the hidden correlations between the CSI data sequences. We ensure that all images have the same size by padding with 0s.

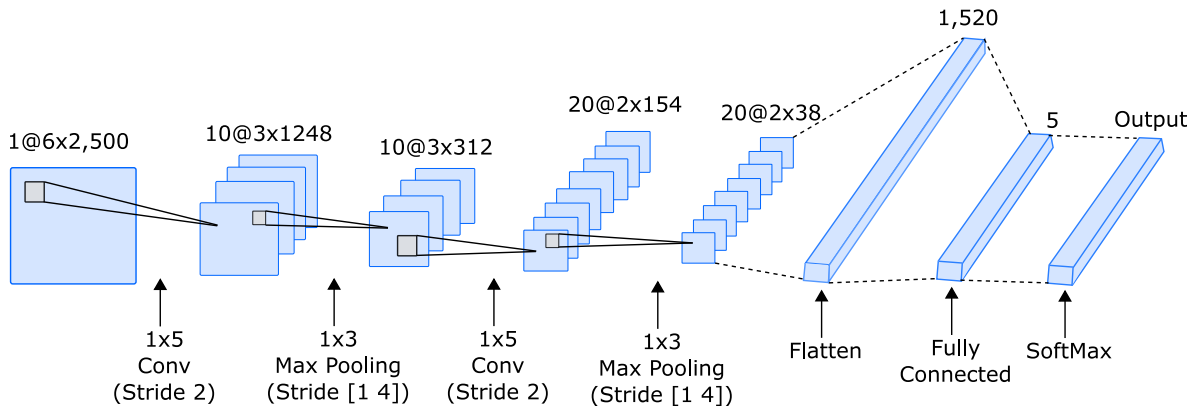


Figure 4.6: CNN architecture.

Figure 4.6 shows the design of the proposed CNN. As shown, it consists of two layers of alternating Convolution, (Batch Normalization, ReLu), and Pooling sublayers

such that the lower layer extracts basic features while the higher layer extracts more complex features [25]. In the following section, we describe the detailed roles of the sublayers.

4.4.1 Convolutional Layer

The convolutional layer basically convolves the input images by sliding the kernels (also called as filters) vertically and horizontally and calculates the dot product of the input and the weights of the kernels.

4.4.2 Batch Normalization Layer

Before providing the result of the convolutional layer as an input to the next layer, the result goes through the normalization layer. The normalization layer is used to speed up the training process and reduce the sensitivity to the initial network configuration.

4.4.3 ReLu Layer

After the convolutional layer and batch normalization layer, a nonlinear activation function σ is executed, for which we adopt the rectified linear unit (ReLU) function. It basically performs a threshold operation to each element we obtain after the convolutional and batch normalization layer. Specifically ReLu layer is selected to avoid the vanishing gradients problem and to achieve much faster training speed.

4.4.4 Max Pooling Layer

In the max pooling layer, the resolution of the feature maps is decreased in order to prevent overfitting using the max pooling.

4.4.5 Dropout and Fully Connected Layer

While training the CNN model, we observed significant overfitting and decided to deploy the dropout layer to reduce the impact of overfitting. Basically, this layer randomly drops out an element of the results of the Max Pooling Layer with a fixed probability p_{drop} . In our experiments, we found that a drop out rate p_{drop} of 0.6 gave the good results.

4.4.6 Fully Connected Layer

Followed by the two layers of alternating Convolution, Batch Normalization, ReLu, and Pooling sublayers is the Fully Connected Layer. This layer is basically the same as the regular neural network which maps the flattened feature into the output classes (*i.e.*, five vehicle types) generating the scores for each output class. Finally, the output scores of the Fully Connected Layer is provided as input to the SoftMax layer in which the scores are converted into values in the range between 0 and 1 such that the sum is 1. This way the SoftMax layer represents the output as a true probability distribution.

Chapter 5

Experimental Results

5.1 Experimental Setup



Figure 5.1: Experimental setting.

We deployed a prototype of DeepWiTraffic in a two-lane rural highway (Figure 5.1). Two laptops (HP Elite 8730w model) were used to develop the prototype. One was a WiFi transmitter, and the other one was a WiFi receiver. These laptops were equipped with 2.53GHz Intel Core Extreme CPU Q9300 processor, 4GB of RAM, and Intel 5300 NIC. DeepWiTraffic was executed on Ubuntu 14.04.04 (kernel version of 4.2.0-27). We deployed another two laptops to record the ground-truth video data on

each side of the road. Separate laptops were used to avoid interfering with the WiFi communication and WiFi CSI data processing. These separate laptops were synchronized with the WiFi transmitter to ensure that video recording is started at the same time WiFi communication is triggered.

Table 5.1: Vehicle types and number of samples

Vehicle Classification			# of Samples
Car-like	Small	Bike	22
		Passenger Car	238
	Medium	SUV	253
		Pickup Truck	252
Truck-like	Large	Large Truck	18

CSI data were collected for about 120 hours over a month. Extracted CSI amplitude and phase set for each passing vehicle was manually tagged based on the recorded video. Consequently, we collected CSI data for a total of 783 vehicles. Table III shows detailed vehicle types and number of samples collected. We referred to Federal Highway Administration (FHA) vehicle classification [3] to determine the vehicle types. Classifying vehicles with more than two axles is known to be quite effective due to the large vehicle body. Mostly the challenge exists in classifying vehicles with two axles due to the very similar body size. As such, we concentrate on classifying vehicles with two axles, *i.e.*, class 1 (motorcycle) class 2 (passenger car) class 3 (SUVs) class 4 (buses) class 5 (trucks), and other class (large trucks) according to the Federal Highway Administration (FHA) classification [3]. Here the large truck means a single unit with the axle count greater than three. Note that we excluded the class 4 (buses) since we spot-

ted only 2 buses in the rural highway during the period of data collection. As Table III shows, we adopted two other typical classification methods namely ‘car-like vs truck-like’ classification [3], and ‘small, medium, large’ classification [29].

Table 5.2: Hyper parameters for deep learning

Parameter Type	Value
Solver	Stochastic Gradient Descent with Momentum (SGDM) Optimizer
Dropout Rate	60%
Shuffle Frequency	Every Epoch
Validation Data	30%
Input Image Size	$6 \times \text{WINDOW_SIZE}$
WINDOW_SIZE	2,500
L2 Regularization	None

Table IV summarizes the hyper parameters we selected to train the CNN model. As shown, we used 70% of the collected CSI data to train the CNN model, and the rest of the 30% for testing purpose. We compared the performance of DeepWiTraffic with that of support vector machine (SVM) and k nearest neighbor (kNN). In particular we used the following five features in training the SVM and kNN models: (1) the normalized standard deviation (STD) of CSI, (2) the offset of signal strength, (3) the period of the vehicle motion, (4) the median absolute deviation (MAD), (5) interquartile range (IR) according to [46] which exploited WiFi CSI for fall detection. Specifically, we select $k=5$ as we found that it gave the best results.

5.2 Detection Accuracy

The detection accuracy was 99.4% (778 out of 783). This high vehicle detection accuracy is attributed to the PCA analysis that achieves sharp differentiation of the CSI amplitude values for passing vehicles by effectively extracting the common features of the CSI amplitude values of 30 subcarriers and representing the CSI amplitude values with a single dimension. The result coincides with the literature that most recent TMSs have very high vehicle detection accuracy. A total of 24 false positives were observed.

5.3 Classification Accuracy

We measured the classification accuracy of DeepWiTraffic. We used the common definition of the classification accuracy, *i.e.*, it is defined as the total number of correctly detected vehicles divided by the total number of detected vehicles. The classification accuracy of DeepWiTraffic is compared with SVM and kNN-based approaches. In this experiment, we randomly selected 30% of the passing vehicles as the validation set for SVM, kNN, and Deep Learning (DeepWiTraffic). We then calculated the average classification accuracy for 1,000 randomly selected validation sets.

Table 5.3: Classification accuracy - SVM

Classification			SVM		
Car-like	Small	Bike	99.3%	85.7%	81.3%
		Passenger Car			75.9%
	Medium	SUV		85.8%	50.6%
		Pickup Truck			75.5%
Truck-like	Large	Large Truck	98.0%	96.2%	95.5%
Average			98.7%	89.2%	75.8%

Table 5.4: Classification accuracy - kNN

Classification			kNN		
Car-like	Small	Bike	99.2%	46.5%	77.2%
		Passenger Car			54.5%
	Medium	SUV	92.8%	92.8%	47.8%
		Pickup Truck			42.5%
Truck-like	Large	Large Truck	92.8%	91.1%	90.4%
Average			96.0%	76.8%	62.5%

Table 5.5: Classification accuracy - CNN

Classification			CNN		
Car-like	Small	Bike	100.0%	91.1%	97.2%
		Passenger Car			91.1%
	Medium	SUV	100.0%	94.1%	83.8%
		Pickup Truck			83.3%
Truck-like	Large	Large Truck	100.0%	100.0%	99.7%
Average			100.0%	95.1%	91.1%

The results are summarized in Table V. All classifiers did a good job in classifying vehicles into car-like and truck-like types. However, the performance decreased as the number of classes increased, especially the classification accuracy for SVM and kNN sharply dropped. In contrast, the average classification accuracy of DeepWiTraffic remained high as 91.1% for individual vehicle types. However, it seemed still challenging for Deep Learning to classify similar sized vehicles, *i.e.*, SUV and pickup trucks, with the accuracy of 83.8% for SUV and 83.8% for pickup trucks. Finding more effective features of CSI data to improve this accuracy is left as an open problem. Overall, DeepWiTraffic shows very promising performance comparable to some camera-based solutions [11][9], and magnetic sensor-based approaches [43][50].

5.4 Classification Accuracy Per Lane

Another interesting research question that we answer here is: how does the lane affect the performance of DeepWiTraffic. To answer this question, we created CNN models separately for each lane. The results for different CNN models for lane 1, lane 2, and aggregated lanes are summarized in Table VI. We found that the effect of lane was negligible when vehicles are classified into car-like and truck-like types. However, the accuracy of the CNN model for aggregated lanes significantly degraded by 8.9% and 8.7% for ‘S,M,L’ classification and individual vehicle classification, respectively. The results indicate that the CNN models should be trained separately for each lane; input CSI data may be tested with each CNN model; and the output with a higher probability should be used.

Table 5.6: Classification accuracy - Lane 1

Classification			Lane 1		
Car-like	Small	Bike	100.0%	91.1%	97.2%
		Passenger Car			91.1%
	Medium	SUV		94.1%	83.8%
		Pickup Truck			83.3%
Truck-like	Large	Large Truck	100.0%	100.0%	99.7%
Average			100.0%	95.1%	91.1%

Table 5.7: Classification accuracy - Lane 2

Classification			Lane 2		
Car-like	Small	Bike	100.0%	90.3%	97.0%
		Passenger Car			87.5%
	Medium	SUV		93.7%	83.1%
		Pickup Truck			80.0%
Truck-like	Large	Large Truck	100.0%	100.0%	99.1%
Average			100.0%	94.7%	89.3%

Table 5.8: Classification accuracy - Mixed Lane

Classification			Mixed Lane		
Car-like	Small	Bike	99.8%	79.6%	95.5%
		Passenger Car			81.6%
	Medium	SUV		80.0%	76.3%
		Pickup Truck			66.5%
Truck-like	Large	Large Truck	99.5%	99.0%	92.1%
Average			99.7%	86.2%	82.4%

Another interesting observation is that the accuracy for Lane 1 is slightly higher than that for Lane 2. The reason is, as we illustrated in Figure 4.1, when a passing vehicle is close to the receiver (being located on Lane 1), WiFi signals for different TX-RX antenna pairs are spaced more widely allowing for capturing more information about the vehicle body.

Chapter 6

Conclusion

We have presented the design, implementation, and evaluation of DeepWiTraffic, a low-cost and portable TMS based on WiFi CSI. With the large amounts of CSI data and ground truth video data that we collected over a month, we performed extensive real-world experiments and successfully validated the effectiveness of DeepWiTraffic. Despite the low cost, the average classification accuracy of 91.1% for five different vehicle types is comparable to most recent non-intrusive vehicle classification solutions. We expect DeepWiTraffic to contribute to solving the endemic cost of issue of deploying a large number of TMSs to cover the huge miles of rural highways.

A possible extension of this work is to develop a WiFi based traffic monitoring system for congested traffic environments. The current system does not classify vehicles effectively under high traffic scenarios as it is primarily designed for low traffic rural highways.

Bibliography

- [1] Challenges of the day-today operation of a traffic monitoring program (georgia dot). <http://onlinepubs.trb.org/onlinepubs/conferences/2016/NATMEC/Wiegand-TannerPPT.pdf>. Accessed: 2018-06-01.
- [2] Fhwa transportation infrastructure management. <https://www.fhwa.dot.gov/legregs/directives/fapg/cfr050b.htm>. Accessed: 2018-06-01.
- [3] Fwha vehicle classification. https://www.fhwa.dot.gov/policyinformation/tmguidetmg_2013/vehicle-types.cfm. Accessed: 2018-06-01.
- [4] Traffic monitoring guide. https://www.fhwa.dot.gov/policyinformation/tmguidetmg_fhwa_pl_17_003.pdf. Accessed: 2018-06-01.
- [5] U.s. traffic monitoring location data. <https://www.fhwa.dot.gov/policyinformation/tables/trafficmonitoring/>. Accessed: 2018-06-01.
- [6] Kamran Ali, Alex X Liu, Wei Wang, and Muhammad Shahzad. Keystroke recognition using wifi signals. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 90–102. ACM, 2015.
- [7] Nicolas Audebert, Bertrand Le Saux, and Sébastien Lefèvre. Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. *Remote Sensing*, 9(4):368, 2017.
- [8] Walid Balid, Hasan Tafish, and Hazem H Refai. Intelligent vehicle counting and

- classification sensor for real-time traffic surveillance. *IEEE Transactions on Intelligent Transportation Systems*, 19(6):1784–1794, 2018.
- [9] Carlo Migel Bautista, Clifford Austin Dy, Miguel Iñigo Mañalac, Raphael Angelo Orbe, and Macario Cordel. Convolutional neural network for vehicle detection in low resolution traffic videos. In *Region 10 Symposium (TENSYMP), 2016 IEEE*, pages 277–281. IEEE, 2016.
- [10] Marco Bottero, Bruno Dalla Chiara, and Francesco Paolo Deflorio. Wireless sensor networks for traffic monitoring in a logistic centre. *Transportation Research Part C: Emerging Technologies*, 26:99–124, 2013.
- [11] Zezhi Chen, Tim Ellis, and Sergio A Velastin. Vehicle detection, tracking and classification in urban traffic. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 951–956. IEEE, 2012.
- [12] Sing Cheung, Sinem Coleri, Baris Dundar, Sumitra Ganesh, Chin-Woo Tan, and Pravin Varaiya. Traffic measurement and vehicle classification with single magnetic sensor. *Transportation research record: journal of the transportation research board*, (1917):173–181, 2005.
- [13] Marco F Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2004.
- [14] Fernando Garcia, Pietro Cerri, Alberto Broggi, Jose Maria Armingol, and Arturo

- De La Escalera. Vehicle detection based on laser radar. In *International Conference on Computer Aided Systems Theory*, pages 391–397. Springer, 2009.
- [15] Jobin George, Leena Mary, and KS Riyas. Vehicle detection and classification from acoustic signal using ann and knn. In *Control Communication and Computing (ICCC), 2013 International Conference on*, pages 436–439. IEEE, 2013.
- [16] Lin Gu, Dong Jia, Pascal Vicaire, Ting Yan, Liqian Luo, Ajay Tirumala, Qing Cao, Tian He, John A Stankovic, Tarek Abdelzaher, et al. Lightweight detection and classification for wireless sensor networks in realistic environments. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 205–217. ACM, 2005.
- [17] Marcus Haferkamp, Manar Al-Askary, Dennis Dorn, Benjamin Sliwa, Lars Habel, Michael Schreckenberg, and Christian Wietfeld. Radio-based traffic flow detection and vehicle classification for future smart cities. In *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th*, pages 1–5. IEEE, 2017.
- [18] Lajos Hanzo, Yosef Akhtman, Jos Akhtman, Li Wang, and Ming Jiang. *MIMO-OFDM for LTE, WiFi and WiMAX: Coherent versus non-coherent and cooperative turbo transceivers*. John Wiley & Sons, 2010.
- [19] Jun-Wei Hsieh, Shih-Hao Yu, Yung-Sheng Chen, and Wen-Fong Hu. Automatic traffic surveillance system for vehicle tracking and classification. *IEEE Transactions on Intelligent Transportation Systems*, 7(2):175–187, 2006.

- [20] Shin-Ting Jeng and Lianyu Chu. A high-definition traffic performance monitoring system with the inductive loop detector signature technology. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 1820–1825. IEEE, 2014.
- [21] Konstantinos Kanistras, Goncalo Martins, Matthew J Rutherford, and Kimon P Valavanis. A survey of unmanned aerial vehicles (uavs) for traffic monitoring. In *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*, pages 221–234. IEEE, 2013.
- [22] Saowaluck Keawkamnerd, Jatuporn Chinrungrueng, and Chaipat Jaruchart. Vehicle classification with low computation magnetic sensor. In *ITS Telecommunications, 2008. ITST 2008. 8th International Conference on*, pages 164–169. IEEE, 2008.
- [23] Lawrence A Klein, Milton K Mills, and David RP Gibson. Traffic detector handbook. Technical report, Federal Highway Administration, 2006.
- [24] Siri Øyen Larsen, Hans Koren, and Rune Solberg. Traffic monitoring using very high resolution satellite imagery. *Photogrammetric Engineering & Remote Sensing*, 75(7):859–869, 2009.
- [25] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
- [26] Ho Lee and Benjamin Coifman. Side-fire lidar-based vehicle classification.

- Transportation Research Record: Journal of the Transportation Research Board*, (2308):173–183, 2012.
- [27] Ho Lee and Benjamin Coifman. Using lidar to validate the performance of vehicle classification stations. *Journal of Intelligent Transportation Systems*, 19(4):355–369, 2015.
- [28] Mengyuan Li, Yan Meng, Junyi Liu, Haojin Zhu, Xiaohui Liang, Yao Liu, and Na Ruan. When csi meets public wifi: Inferring your mobile phone password via wifi signals. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1068–1079. ACM, 2016.
- [29] Mingpei Liang, Xinyu Huang, Chung-Hao Chen, Xin Chen, and Alade Tokuta. Counting and classification of highway vehicles by regression analysis. *IEEE Transactions on Intelligent Transportation Systems*, 16(5):2878–2888, 2015.
- [30] Kang Liu and Gellert Mattyus. Fast multiclass vehicle detection on aerial images. *IEEE Geosci. Remote Sensing Lett.*, 12(9):1938–1942, 2015.
- [31] Wenteng Ma, Daniel Xing, Adam McKee, Ravneet Bajwa, Christopher Flores, Brian Fuller, and Pravin Varaiya. A wireless accelerometer-based automatic vehicle classification prototype system. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):104–111, 2014.
- [32] Soner Meta and Muhammed G Cinsdikici. Vehicle-classification algorithm based

- on component analysis for single-loop inductive detector. *IEEE Transactions on Vehicular Technology*, 59(6):2795–2805, 2010.
- [33] Luz Elena Y Mimbela and Lawrence A Klein. Summary of vehicle detection and surveillance technologies used in intelligent transportation systems. Technical report, Federal Highway Administration’s Intelligent Transportation Systems Joint Program Office, 2007.
- [34] R Niessner, H Schilling, and B Jutzi. Investigations on the potential of convolutional neural networks for vehicle classification based on rgb and lidar data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:115, 2017.
- [35] AY Nooralahiyan, Howard R Kirby, and D McKeown. Vehicle classification by acoustic signature. *Mathematical and Computer Modelling*, 27(9-11):205–214, 1998.
- [36] Enas Odat, Jeff S Shamma, and Christian Claudel. Vehicle classification and speed estimation using combined passive infrared/ultrasonic sensors. *IEEE Transactions on Intelligent Transportation Systems*, 2017.
- [37] Enas Odat, Jeff S Shamma, and Christian Claudel. Vehicle classification and speed estimation using combined passive infrared/ultrasonic sensors. *IEEE Transactions on Intelligent Transportation Systems*, 19(5):1593–1606, 2018.
- [38] Samer A Rajab, Ahmad Mayeli, and Hazem H Refai. Vehicle classification and

- accurate speed calculation using multi-element piezoelectric sensor. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 894–899. IEEE, 2014.
- [39] Souvik Sen, Božidar Radunovic, Romit Roy Choudhury, and Tom Minka. You are facing the mona lisa: spot localization using phy layer information. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 183–196. ACM, 2012.
- [40] Markus Stocker, Mauno Rönkkö, and Mikko Kolehmainen. Situational knowledge representation for traffic observed by a pavement vibration sensor network. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1441–1450, 2014.
- [41] Saber Taghvaeeyan and Rajesh Rajamani. Portable roadside sensors for vehicle counting, classification, and speed measurement. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):73–83, 2014.
- [42] Tianyu Tang, Shilin Zhou, Zhipeng Deng, Lin Lei, and Huanxin Zou. Arbitrary-oriented vehicle detection in aerial imagery with single convolutional neural networks. *Remote Sensing*, 9(11):1170, 2017.
- [43] Rui Wang, Lei Zhang, Kejiang Xiao, Rongli Sun, and Li Cui. Easisee: Real-time vehicle classification and counting via low-cost collaborative sensing. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):414–424, 2014.
- [44] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. Understanding and modeling of wifi signal based human activity recognition. In *Proceed-*

- ings of the 21st annual international conference on mobile computing and networking*, pages 65–76. ACM, 2015.
- [45] Yan Wang, Jian Liu, Yingying Chen, Marco Gruteser, Jie Yang, and Hongbo Liu. E-eyes: device-free location-oriented activity identification using fine-grained wifi signatures. In *Proc. of Mobicom*, 2014.
- [46] Yuxi Wang, Kaishun Wu, and Lionel M Ni. Wifall: Device-free fall detection by wireless networks. *IEEE Transactions on Mobile Computing*, 16(2):581–594, 2017.
- [47] Myounggyu Won, Shaohu Zhang, and Sang H Son. Witraffic: Low-cost and non-intrusive traffic monitoring system using wifi. In *Proc. of ICCCN*, 2017.
- [48] Chenshu Wu, Zheng Yang, Zimu Zhou, Kun Qian, Yunhao Liu, and Mingyan Liu. Phaseu: Real-time los identification with wifi. In *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pages 2038–2046. IEEE, 2015.
- [49] Chang Xu, Yingguan Wang, Xinghe Bao, and Fengrong Li. Vehicle classification using an imbalanced dataset based on a single magnetic sensor. *Sensors*, 18(6):1690, 2018.
- [50] Bo Yang and Yiqun Lei. Vehicle detection and classification for low-speed congested traffic with anisotropic magnetoresistive sensor. *IEEE Sensors Journal*, 15(2):1132–1138, 2015.
- [51] Yunze Zeng, Parth H Pathak, Chao Xu, and Prasant Mohapatra. Your ap knows

how you move: fine-grained device motion recognition through wifi. In *Proc. of HotWireless*, 2014.

- [52] Jin Zhang, Bo Wei, Wen Hu, and Salil S Kanhere. Wifi-id: Human identification using wifi signal. In *Proc. of DCOSS*, 2016.