WiDet: Wi-Fi Based Device-Free Passive Person Detection with Deep Convolutional Neural Networks

Hua Huang
Stony Brook University
New York
hua.huang@stonybrook.edu

Shan Lin
Stony Brook University
New York
shan.x.lin@stonybrook.edu

ABSTRACT

To achieve device-free person detection, various types of signal features, such as moving statistics and wavelet representations, have been extracted from the Wi-Fi Received Signal Strength Index (RSSI), whose value fluctuates when human subjects move near the Wi-Fi transceivers. However, these features do not work effectively under different deployments of Wi-Fi transceivers because each transceiver has a unique RSSI fluctuation pattern that depends on its specific wireless channel and hardware characteristics. To address this problem, we present WiDet, a system that uses a deep Convolutional Neural Network (CNN) approach for person detection. The CNN achieves effective and robust detection feature extraction by exploring distinguishable patterns in Wi-Fi RSSI data. With a large number of internal parameters, the CNN can record and recognize the different RSSI fluctuation patterns from different transceivers. We further apply the data augmentation method to improve the algorithm robustness to wireless interferences and pedestrian speed changes. To take advantage of the wide availability of the existing Wi-Fi devices, we design a collaborative sensing technique that can recognize the subject moving directions. To validate the proposed design, we implement a prototype system that consists of three Wi-Fi packet transmitters and one receiver on low-cost off-the-shelf embedded development boards. In a multi-day experiment with a total of 163 walking events, WiDet achieves 94.5% of detection accuracy in detecting pedestrians, which outperforms the moving statistics and the wavelet representation based approaches by 22% and 8%, respectively.

KEYWORDS

device-free passive localization; convolutional neural network; person detection

ACM Reference Format:
Hua Huang and Shan Lin. 2018. WiDet: Wi-Fi Based Device-Free Passive Person Detection with Deep Convolutional Neural Networks. In *21st ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM '18), October 28-November 2, 2018, Montreal, QC, Canada.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3242102.3242119

MSWIM '18, October 28-November 2, 2018, Montreal, QC, Canada

ACM ISBN 978-1-4503-5960-3/18/10…$15.00
https://doi.org/10.1145/3242102.3242119

1 INTRODUCTION

Mainstream Wi-Fi based device-free person detection systems rely on either the Channel State Information (CSI) or the Received Signal Strength Index (RSSI) measurements. However, currently only a small number of Wi-Fi adapter models support access to CSI, which limits its ability for wide adoption. RSSI measurements are easily accessible in most Wi-Fi devices, but it has been challenging to extract effective and robust features for RSSI because different Wi-Fi adapters generate different RSSI fluctuation patterns due to their unique wireless channel and hardware characteristics. Specifically, the Wi-Fi signal experiences various types of degradations, including path loss, shadowing effect and multi-path effect, which are dependent on the Wi-Fi transceiver deployment location and the wireless channel of the environment. The RSSI measurements on Wi-Fi transceivers are hardware dependent and discrepancies exist even for devices from the same vendors [14]. As a result, as we have seen in experiments, RSSI recorded on each Wi-Fi transceiver fluctuates with a unique pattern. Many existing person detection techniques that rely on hand-crafted signal features, such as wavelet representation [21, 22] and moving statistics [31] based features. However, the detection performance of these features degrades when multiple different Wi-Fi links are used. It will be work intensive to design specialized detection features for each Wi-Fi link.

We apply the machine learning technique to address the challenge of effective feature extraction. Recently, deep learning techniques, Convolutional Network Networks (CNNs) in particular, have achieved remarkable success in time series classification problems [13, 25]. The advantage of the CNN is that it can learn detection features directly from the raw data samples. With a large number of internal parameters, the CNN can record and recognize the unique RSSI fluctuation patterns for all the different Wi-Fi transceivers. We designed a CNN architecture that consists of multiple convolutional layers, with each layer consisting of learnable filters that can detect the occurrence of some specific signal patterns. The parameters in the CNN are fine-tuned during the training phase using the back propagation algorithm. The outputs of the stacked convolutional layers are treated as features and are fed into a fully connected layer that conducts classification. We further plot the Class Activation Map (CAM) to interpret the specific patterns in the data that lead to the final detection results.

To adequately train a CNN model, a sufficient training dataset that includes all the common data variations is needed. In the scenario of device-free person detection, one of the most common types variations is the change of subject walking speeds. When a person walks at different rates, the durations of the RSSI fluctuations change accordingly. It will be time consuming to collect training data of all walking speeds. Instead, we apply the data augmentation technique to expand the size of the training dataset [13]. The basic idea is to warp the data samples with different ratios to mimic data changes caused by the variations of walking speeds. Since the warped augmentation data can be generated based on existing training dataset, we can improve the generability of the CNN model without increasing the data collection efforts. Another type common variations is the wireless noises that causes the RSSI to vary dramatically. It is known that many Wi-Fi connections are bursty: they shift between poor and good connection quality [19]. To reduce the impact on person detection accuracy, we also generate additional training data by adding random noises that resemble such connection changes. Utilizing these two types of augmented data, we further improve the performance of person detection.

The wide deployment of Wi-Fi devices in the indoor environments provides us with opportunity to monitor the a walking event with multiple sender-receiver pairs. However, the RSSI data collected by different transceiver pairs have different fluctuation patterns, magnitudes and durations. We design a Dynamic Time Warping (DTW) based algorithm that can cope with the cross-device data heterogeneity, and can determine the subjects' walking directions. To facilitate system deployment, a low-cost, convenient implementation is needed. We build our system on Raspberry Pi development boards, with small USB Wi-Fi adapters attached. We use the open source libraries Aircrack to control the transmission and reception of the customized Wi-Fi packets. We deploy our system in our department building and conduct extensive testing. In a multi-day experiment with 163 walking instances, our deep convolutional neural network based approach is able to achieve 94.5% of detection accuracy. Our contributions are summarized as follows:

- We proposed a Wi-Fi based device-free person detection system that uses a deep Convolutional Neural Network (CNN) architecture. The CNN can automatically extract effective features from the Wi-Fi RSSI measurements to conduct person detection. Using a visualization technique called the Class Activation Map (CAM), we showed that the CNN is able to recognize RSSI fluctuations caused by human motions while remaining robust to wireless channel noises and interferences.
- To improve the system robustness to the wireless signal noises and the changes of subject moving speeds, we applied the data augmentation techniques that generates additional data to better train the CNN classifier.
- To take advantage of the ubiquitous deployment of Wi-Fi devices in the indoor environment, we designed a collaborative sensing method to determine the subjects' walking directions using RSSI data collected from multiple transceiver pairs.
- We implemented a prototype system with three transmitters and one receiver on low-cost embedded platforms. In a multi-day experiment with 163 walking events, WiDet achieved 94.5% of detection accuracy, outperforming the moving statistics and the wavelet representation based approaches by 22% and 8%, respectively.

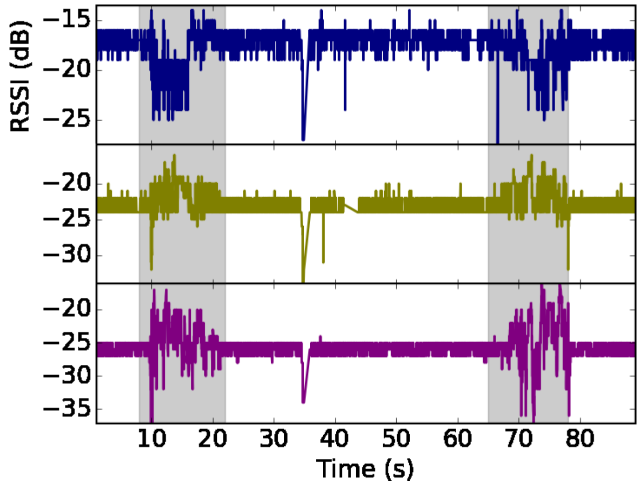

Figure 1: When a person moves (gray areas), three Wi-Fi receivers record different RSSI fluctuation patterns.

2 MOTIVATION

2.1 The Complexity of Wireless Signal Fluctuations

The primary challenge we face is the need to find effective features that can robustly detect people walking. It's challenging because the signal strength changes are determined by a multitude of factors that are unique to each Wi-Fi transceiver. The Wi-Fi signal experiences various types of fading, including path loss, the shadowing and the multipath effects, that are dependent on the wireless channel characteristics near the transceiver's deployment location. Furthermore, the Wi-Fi RSSI measurements are device-dependent. Generally lower RSSI values indicate weaker signal strengths, but there is no standardized relationship of any particular energy level to the RSSI reading. It has been shown that discrepancies in measurements exist even for transceivers built by the same vendors [4, 14]. As a result, the RSSI values fluctuate with different patterns on different Wi-Fi transceivers when a person moves in the nearby environment, and it's difficult to explicitly design a feature set that is effective for all the transceivers.

The challenge is illustrated in Figure 1. In this experiment, we have three Wi-Fi transmitter-receiver pairs deployed along a corridor. There is one person moving between the transmitter and the receivers between 10-20 and 65-80 seconds, illustrated in gray boxes. We can see that the RSSI values fluctuate differently when the person is moving. In the first row of the figure, the RSSI drops, while the RSSI increases in the second row. In the third row, the RSSI value fluctuates up and down, without significant changes in the mean value. Furthermore, we can see that the three links have different RSSI values when no one is walking. Link 1,2 and 3 have RSSI values of approximately -17, -23, and -26 dB, respectively when the environment is quiet.

To handle the different in the RSSI fluctuation patterns, we adopt the machine learning technique, convolutional neural network in particular, to learn the signal features directly from data. For time series data, the Convolutional Neural Networks (CNNs) have been shown to achieve state of the art performance [25]. One advantage of CNN is that it can learn the detection features that in traditional

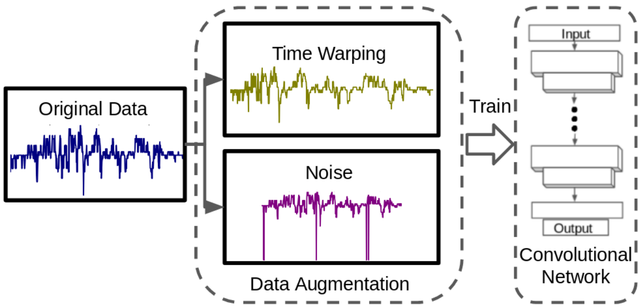

Figure 2: Algorithm Overview

algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage.

In the CNN architecture, each higher level convolutional layer captures data patterns of longer time scale. The outputs of the convolutional layers function as a detection feature set and are fed to a fully connected layer for person detection. Using the back-propagation algorithm, the parameters of the CNN are fine-tuned automatically in the training phase to optimize the detection accuracy. To understand the patterns captured by the CNN, we further apply the Class Activation Map (CAM) technique to visualize the activation values of the intermediate layers in Section 5.4. This way we can visually inspect what RSSI fluctuation patterns contribute most to the classification decision. We will present the CNN architecture in details in Section 3.2.

2.2 Environmental Variations

Wi-Fi connections shift between poor and good connection quality [19]. As a result, the RSSI values shift from time to time. For example, at around second 35 in Figure 1, we can see that the RSSI values for all the three links drop significantly. These types of noises, if not handled properly, can cause false detections results. Another type of variations is the change of human subject moving speeds. The change of moving speed will create warped versions of the RSSI fluctuation patterns. In small scale deployments, we may not be able to record all types of moving speeds in the training data, which can cause lower performance of the detection algorithm.

To mitigate these two problems, we apply the data augmentation techniques. We generate additional training data by adding noises and warping the RSSI time series. With the additional generated training data, we can better train the CNN model about the intra-class variations (noises and warping distortions). We will present the data augmentation techniques in Section 3.3.

3 ALGORITHM DESIGN

3.1 Overview

An overview of the system is shown in Figure 2. We use Wi-Fi devices to continuously record time-stamped RSSI values. After the training data is collected, we firstly conduct a data preprocessing by removing outliner values, and data resampling to ensure constant data rate. Then we generate augmenting training dataset by adding noises and introducing local time warping. Using both the augmented and original datasets, we train the CNN using the back-propagation algorithm. After training, we use CNN model to conduct person detection.

3.2 Deep Convolutional Neural Network Architecture

The CNN network architecture is shown in Figure 3. We use stacked convolutional layers as a feature extractor. Lower layer convolutional layers tend to capture localized detailed signal patterns, while the higher layers tend to reveal larger scale patterns. Specifically, given an input $\mathbf{x}$, a convolutional layer computes an output h using the following equation set:

$$\begin{aligned} y &= \mathbf{W} \otimes \mathbf{x} + \mathbf{b} \\ s &= BN(y) \\ h &= ReLU(s). \end{aligned} \tag{1}$$

In this equation set, $\otimes$ is the convolution operator. $\mathbf{b}$ represents constant offsets, and $\mathbf{W}$ are filters that can detect particular signal patterns in $\mathbf{x}$. Both $\mathbf{b}$ and $\mathbf{W}$ are learnable parameters that can be updated during the training phase. For the first level of the convolutional layer, $\mathbf{x}$ is the raw time series data input. For each upper-level convolutional layer, $\mathbf{x}$ represents the output of the immediately previous layer.

BN represents Batch Normalization, which normalizes the activations of the previous layer at batch, i.e. applies a transformation that maintains the mean activation close to 0 and the standard deviation close to 1. This operation has been shown to significantly speed up the training process [9].

ReLU represents Rectified Linear Unit. It is defined as $ReLU(s) = \max(0, s)$. The motivation for applying Rectified Linear Unit is three-folded: it's computationally efficient (only involves comparison, addition and multiplication), has sparse activation (only values greater than 0 are activated), and has few vanishing gradient problems when compared with the sigmoid activation function [8].

To reduce over-fitting and improve the network's ability to generalize, we apply dropout layers between convolutional layers. Basically during training, a certain percentage of neurons on a layer will be deactivated. This improve generalization because it forces the layer to learn with different neurons with the same "concept".

After the convolutional layers, the outputs are fed into a Global Average Pooling (GAP) layer. The GAP layer can minimize over-fitting by reducing the total number of parameters in the model. Specifically, let the output dimension of the convolutional network be $h \times d$. The GAP layer reduces each h dimensional feature map into a single number by taking average, and produce an intermediate output with dimension $1 \times d$. Greater details about GAP can be found in [34]. Then a sigmoid layer is used to compute a single detection value, which is compared with a threshold to decide whether or not there is an walking event.

In total, we have seven convolutional layers, and the number of filters used in each layer is marked in Figure 3. These are tunable network parameters and we arrive at these values after extensive tuning and testing.

3.3 Data Augmentation

Compared to fully connected deep neural networks, convolutional neural networks tend to suffer less from overfitting. Nevertheless,

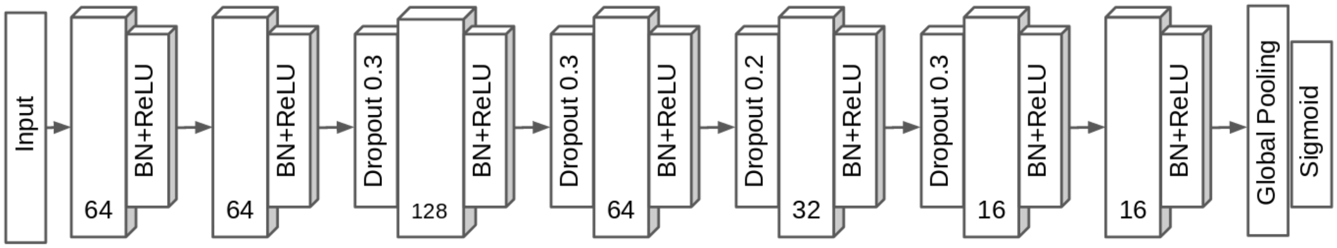

Figure 3: Deep Convolutional Neural Network Architecture

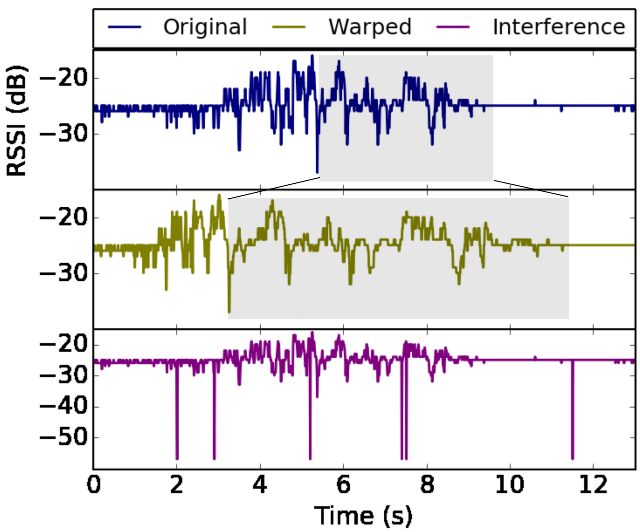

Figure 4: Synthesized Augmenting Data for Different Moving Speeds and Wireless Noises.

CNNs can still benefit from data augmentation techniques [7, 12, 13]. The basic idea of data augmentation techniques is to synthesize new data by transforming existing labeled training data samples, so that the neural network model can learn a wider range of intra-class variations. By observing the collected RSSI data traces, we find two common types of variations. The first type is the change of the signal fluctuation durations due to different walking speeds. The second type is signal strength drops during interferences, i.e., RSSI drops suddenly from time to time. Based on these observations, we design two data augmentation methods.

Augmenting Data with Different Moving Speeds. To synthesize augmentation data with different moving speeds, we warp a randomly selected slice of a time series by re-sampling it up or down, as shown in the second row in Figure 4. In particular, given a time stamped RSSI series s of length n, we randomly select a sub-series of length αn, where $0 < \alpha < 1$. Then we re-sample this selected sub-series to the length of βn. A larger value of β corresponds to simulating slower walking speeds. Study shows that the comfortable walking speed for pedestrians ranges from 1.3 to 2.5 m/s [6]. Therefore, we don't need to only consider very slow or fast walking speeds, and the range of β is set to be $(0.5\alpha, \min(2\alpha, 1))$. Finally we re-sample the remaining section of the data so that the total length of the generated data remains n, as illustrated in the second row in Figure 4.

Augmenting Data with Added Noise. Adding noise to the training data is a common technique for data augmentation. In our system, we generate augmented data by adding a noise value t with probability p. The range t is empirically determined to be $[0, -30]$, and p is a tunable parameter with a value around one thousandth. An example is shown in the third row in Figure 4.

After the augmenting data are generated, the deep convolutional neural network is trained on the original data together with the augmenting data. The CNN is ready for person detection after training.

3.4 Collaborative Moving Direction Detection Based On DTW

Wi-Fi devices are ubiquitous in modern office or home environments. After walking events are detected using the CNN networks, we can utilize sensing results from multiple Wi-Fi transceiver pairs to determine the walking directions. Intuitively, we can do so by comparing the time of occurrences of the RSSI fluctuations. However, it's difficult to conduct comparison between RSSI measurements from multiple devices directly, because the RSSI fluctuation in each transceiver pair has a unique pattern and duration. For example, as discussed in Section 2, the RSSI decreases in some devices but increases in others when a pedestrian is walking.

To alleviate the problem of different fluctuation patterns, we use the moving variance of the RSSI values instead of the raw RSSI measurements. Then we design a Dynamic Time Warping (DTW) based algorithm to address the discrepancies in fluctuation time durations. The basic idea is that DTW can be used to find the warped versions of the two time series so that they optimally match with each other. This way we can eliminate the impacts of duration discrepancies and estimate the differences in the fluctuation time. Intuitively, if the data points in A consistently matches with points in B with later (earlier) time stamps, then it indicates that events happens earlier (later) in A.

DTW is a classical algorithm and a brief description is provided as follows. Let $A = \{a_1, a_2, ..., a_n\}$ and $B = \{b_1, b_2, ..., b_n\}$ represent two sequences of length n, respectively. The best match between the two sequences is defined as the one with the lowest distance path after aligning one to the other. The optimal matching is represented by a matrix *path* such that $a_{path[i,1]}$ is aligned with $b_{path[i,2]}$. We compute the matrix *path* using the efficient algorithm described in [18].

The algorithm is illustrated in Figure 5, we plot the dynamic time warping between two moving variance series, shown in the left and in the bottom of the figure. The color map represents the matching

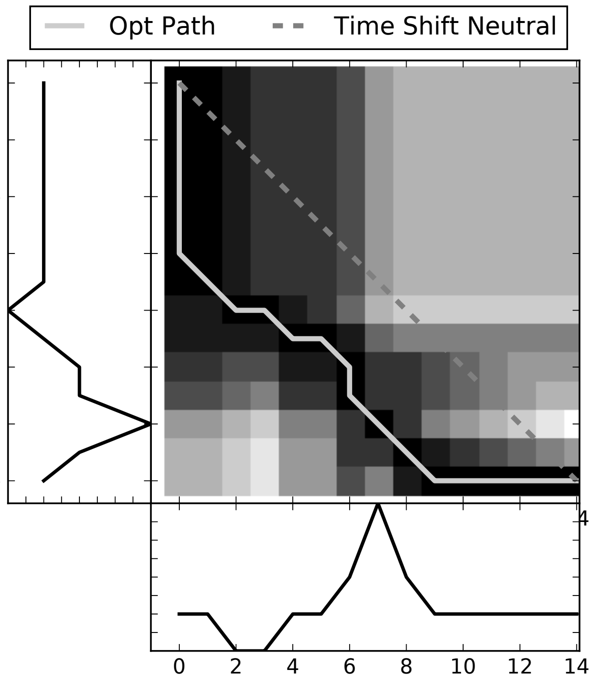

Figure 5: The Dynamic Time Warping Between Two Moving Variance Series

cost (darker colors means lower costs), and the thick gray line represents the optimal matching path, which is represented by the matrix *path*. Intuitively, if there is no time delay of the fluctuation events within two series, then *path* will be a purely diagonal line, as shown in the gray dashed line. On the other hand, if time delay exists between two fluctuation events, the optimal path will deviate from the diagonal line. This is illustrated in Figure 5. The moving variance series on the left side reaches peak at time step 12, while the moving variance series on the bottom reaches peak earlier at time step 7. Since the time series on the bottom fluctuate earlier than the one on the left, the Opt Path travels below the diagonal line. Based on this intuition, we can use the area between the optimal path and the diagonal line as an indicator about the relative time difference between fluctuation times in the two time series.

In particular, the walking direction detection algorithm works as follows. For each receiver i, we extract the moving variance of the RSSI, represented by $S_i = \{s^i(1), s^i(2), ...s^i(n)\}$. Then we apply the DTW algorithm to find their matching path *path* between S_i and S_j from two transmitter-receiver pairs. To quantify the time difference between the walking events, we define the Sequence Mis-match Index (SMI), which is the area encircled by the optimal path and the diagonal line, divided by the area of the entire rectangle. We further define SMI to be negative when the path is below the diagonal and positive otherwise. The SMI between two sequences S_i and S_j is computed using the following equation:

$$SMI = (\sum_i (path[i,1] - path[i,2]))/n^2. \tag{2}$$

Finally, we compare SMI with a threshold T_m. If $SMI > T_m$ or $SMI < -T_m$, then the algorithm output that the pedestrian is moving along the direction from pair i to pair j, or the reverse direction, respectively.

4 IMPLEMENTATION

The system is running on the Linux kernel version 2.6 on a Raspberry Pi development board. We use the Alfa Network Wi-Fi adapters (AWUS036H) to transmit and receive Wi-Fi packets. To control Wi-Fi, we use the Aircrack-ng library [1]. The library enables us to custom-build and broadcast 802.11 frames in the transmitter, and capture 802.11 frames in the receivers.

For each packet transmitter, we construct customized beacon frames. We use the beacon frame format defined in RadioTap [3] as a template, and assign a unique value to the field Source Address (DA) field as the transmitter ID. To enable packet loss detection, we assign a packet ID number to the Sequence Control field (Seq-ctl) field. After the beacon frame is built, we broadcast it using socket building and transmission API in the Aircrack-ng library, at an interval of 10ms.

We configure the receivers into the Monitoring Mode so that they can receive any Wi-Fi packets available. After the packets are received, we use the RadioTap parser software to analyze the information [3]. We extract the Source Destination to identify the transmitters, and abandon unrelated packets. Then we use the RadioTap parser to extract the packet RSSI, time stamp and the sequence number for walking detection.

The training of the CNN architecture is computational intensive. We transfer the collected training dataset to a cloud-based server equibed with a 2.2Ghz CUP, 12Gb memory and a Tesla K80 GPU. After the CNN model is trained, the person detection can be computed efficiently on devices with lower computing power.

5 EVALUATION

5.1 Deployment

We have conducted walking detection experiments to evaluate the performance of the proposed system. The layout of the office environment of the experiment is shown in Figure 6. Our testbed consists of three Wi-Fi transmitters and one Wi-Fi receiver, whose locations are depicted using black dots (APs) and triangles (MP), respectively. There are no direct line of sight channels between the transmitters and the receiver and the walls are made of wood.

The pedestrians are walking in the corridor, as encircled by the red lines in Figure 6. The experiments have been repeated at different times across multiple days, and 163 moving events are recorded. The ground truth of human moving times are recorded manually by another operator. We slice the collected data into 15-second-long segments. If an motion event occurs during this time interval, the segment of data has a positive label. The data segment has a negative label otherwise. We randomly split the entire dataset into training and testing data. For the training data, we use the data augmentation techniques to generate additional data to improve the algorithm training. The trained machine learning model is then tested in the test dataset to find detection accuracy. We use the detection accuracy as the evaluation metric, which is defined as the percentage of correct detections versus the total number of test samples.

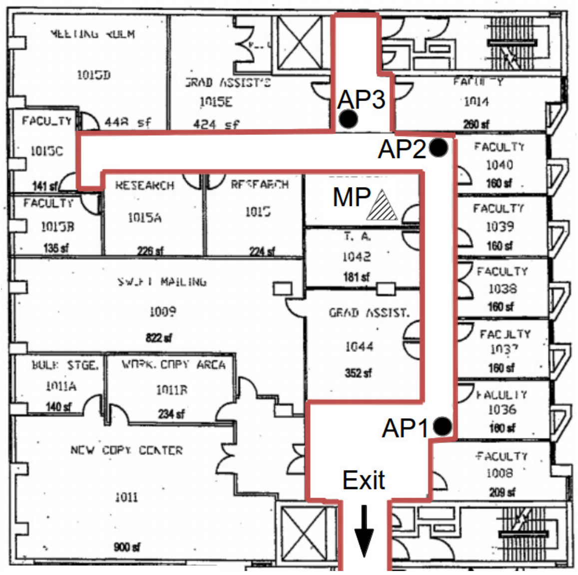

Figure 6: Deployment Floor Plan

5.2 Baseline Algorithms

5.2.1 Moving Statistics Based Algorithm. The first baseline algorithm we implemented is the widely used moving statistics based algorithms[31]. This algorithm computes the average and variance of the data within a moving detection window w, and compares these statistical values with a threshold T to determine whether or not an event has occurred.

5.2.2 Wavelet Transform Based Algorithm. Next we designed another baseline algorithm, which is based on wavelet transform. From the data we observe that human motions mainly caused low-frequency RSSI fluctuations, while other noises have a much broader distribution of energy in the frequency domain. The wavelet transform is an effective tool that can localize an event in both time and frequency domain. Using the wavelet transform as a feature extractor, we can achieve more accurate detection accuracy than statistical features.

The basic idea of the wavelet transform based approach is to analyze the time series data in a multi-scale perspective. Using small scale wavelets, the transformed signal will have large peaks during the event process, which can be used to locate the motion events in time domain. When large scale wavelets are applied, the transformed signal will generate two peaks that represent the start and over of each event.

Specifically, to extract the fine-grained wavelet features, we conduct continuous wavelet transform [5] using a small scale s_f wavelet on the input RSSI data. Then we apply the local maximal algorithm to find the tallest three peaks. The average height h_f of these three peaks is used as the fine-scale wavelet feature. To extract the coarse-scale wavelet features, we conduct continuous wavelet transform using a wavelet of a large scale s_c on the input RSSI data. Then we apply the local maximal algorithm to find the most prominent two peaks. The average height h_c of these two peaks and the distance d between them are used as the coarse-scale wavelet features.

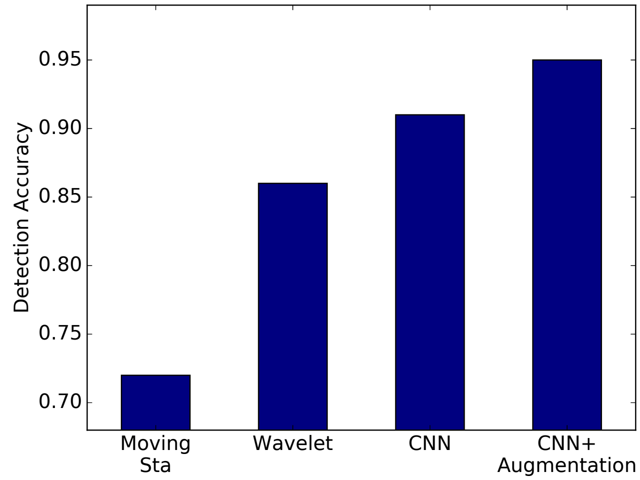

Figure 7: Detection Accuracy

After the features h_f, h_c and d are collected, we use a Bayes classifier to detect whether there is a pedestrian. The wavelet scales s_f and s_c are tunable parameters and we set them to be 0.2 s and 2 s, respectively.

5.3 Walking Detection Accuracy

We firstly evaluate the accuracy for the system to detect walking of pedestrians. The results are shown in Figure 7. We can see that using the basic moving statistics based algorithm (Moving Sta), the detection accuracy is around 72%. This is because these features cannot handle the disturbances caused by interferences and noises. Using the wavelet transform to extract motion features, we can improve the detection accuracy to 86%. This shows the improvement of using frequency domain features. The wavelet transform based feature is more effective in coping with random noises.

When we apply the Convolutional Neural Network (CNN), we have achieved 91.9% of detection accuracy. This is because compared with hand-craft features, the deep machine learning architecture can learn more subtle patterns in the time series data. By using the data augmentation technique in the training phase, we can improve the performance of the CNN to 94.5%. This is because the added noises and time-warped training samples can help the network gain robustness and improve generability to the testing dataset. As a result, the test detection accuracy is improved.

5.4 Localize the Contributing Regions with Class Activation Map

One drawback of the CNN is that it can not provided explicit information about the signal patterns it recognizes. One way to partially mitigate this problem is to plot the Class Activation Maps (CAM) of data samples. A CAM can visualize the contributing regions in each data sample. This can help us highlight the discriminative subsequences that contribute most to the detection result. Furthermore, CAM provides a way to find a possible explanation on why the CNN works for certain classification settings.

The basic idea of the CAM is that each node in the Global Average Pooling (GAP) layer corresponds to a different activation map, and that the weights connecting the GAP layer to the final sigmoid layer

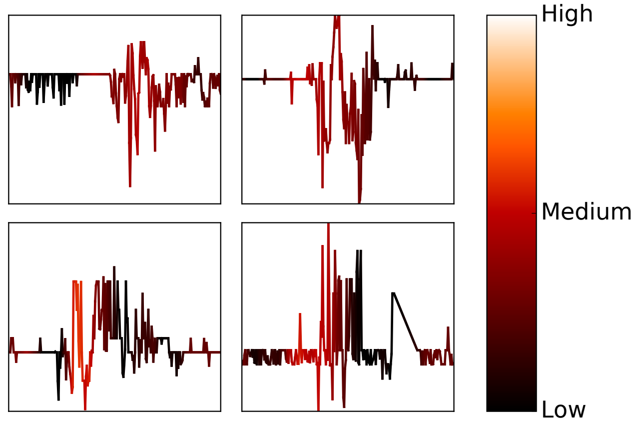

Figure 8: Sample Class Activation Maps

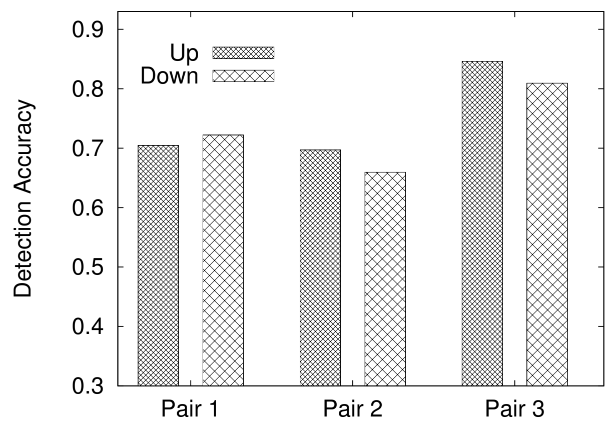

Figure 9: Moving Direction Detection Accuracy

encode each activation map's contribution to the detection result. To obtain the class activation map, we sum the contributions of each of the detected patterns in the activation maps, where detected patterns that are more important to the predicted object class are given more weight. Using the approach described in [34], we plot the CAMs for a few sample data in Figure 8.

In Figure 8, the more discriminative regions in the input data are highlighted using lighter red color. We can see that the fluctuations caused by walking have brighter colors, while the random noises have darker colors. These indicate that the CNN is able to distinguish human motions from random heat noises. Furthermore, in the right lower figure, we can see that the abrupt change of RSSI, which is caused by wireless interference, has dark color. This shows that the CNN is able to recognize that wireless interferences are not associated with human walking events.

5.5 Moving Direction Detection

Next we evaluate our collaborative moving direction detection algorithm. As depicted in Figure 6, we have three transceiver links, i.e., l1 between AP1 and MP, l2 between AP2 and MP, and l3 between AP3 and MP. To determine the walking direction of the subjects, RSSI data from two links are needed. We group the three transceiver links into three pairs: l1-l2, l1-l3, and l2-l3, and execute the moving direction detection algorithm on each pair. We define the up/down direction as walking north/south (the Exit arrow in Figure 6 points to the south). The ground-true moving directions are recorded manually. We calculate the detection accuracy for both directions. The results are shown in Figure 9.

From Figure 9, we can see that the detection performances for both walking directions are similar. This shows that our algorithm is able to handle both moving directions equally. We can also see that the detection accuracy in Link Pair 3, (l1 and l3), is more than 82%, which is higher than the other two links (around 70%). This is because the distance between the two links in Pair 3 is larger than the other two pairs. As a result, the time differences recorded by these two links are larger, which facilitate the detection of moving directions. This indicates that increasing the distance between the two Wi-Fi links tends to improve performance of direction detection, given that the subjects will still traverse both.

6 RELATED WORK

Device-free localization has attracted much attention due to its unique advantages: it doesn't require active user participation, can achieve through the wall detection, and is privacy preserving. Early works such as [15, 27, 30–32] use moving average or variance of RSSI levels as features to detect human motions and locations. To improve performance, more advanced features based on wavelet transform are designed [21, 22]. However, these type of features are not effective in recognizing the different RSSI fluctuation patterns recorded on different Wi-Fi transceivers. To address this problem, we apply the Convolutional Neural Network (CNN) to extract all the different RSSI fluctuation patterns recorded on different Wi-Fi transceivers automatically. Using the data augmentation technique, we further improve the robustness of the algorithm to the wireless noises.

With development of the Orthogonal Frequency Devision Multiplexing (OFDM) technologies in the Wi-Fi devices, the Channel State Information (CSI) has been used in device free localization. By exploiting the subcarrier signal strengths in the CSI , rich multi-path information about the environment can be extracted [17, 23, 28, 29]. The CSI technology enables many applications, including fine-grained localization [20], emotion sensing [33], and vital sign monitoring [24]. One drawback of the CSI based system is that there are limited Wi-Fi devices that can support providing CSI information. Currently many CSI based systems are implemented on the Intel's IWL 5300 NIC [2], since most other Wi-Fi NICs don't provide CSI information to developers. This limits the adoption of the CSI based localization systems. On the contrary, in many existing Wi-Fi devices, as well as other wireless protocols such as bluetooth and zigbee, signal strength can be easily retrieved by accessing the RSSI in the MAC layer. This ability to utilize wireless signals in the existing wireless infrastructure facilitates wider deployment.

Due to the wide availability of the Wi-Fi infrastructure, much work has been devoted to explore collaborative RF sensing using multiple wireless devices. One representative technique is called Radio Tomographic Imaging (RTI)[10, 11, 16, 26]. When wireless links are obstructed by objects moving in the radio tomographic network, they will experience signal degradation due to shadowing effects. The RTI systems utilize this phenomenon to image the attenuation of objects within the network area. However, these systems require dense deployment of homogeneous wireless transceivers, and all of them require direct line of sight to each other. These requirements limit the application of this technology. In this work, our goal is to achieve walking direction detection with low-cost deployment and without the line-of-sight deployment requirement. By utilizing the Dynamic Time Warping (DTW) technique, our system can achieve walking direction detection with small numbers of Wi-Fi devices (As few as two transmitters and one receiver), and can achieve through the wall walking direction detection.

7 CONCLUSION AND FUTURE WORK

In this work, we have designed a deep convolutional neural network to conduct device-free walking detection with high accuracy. We show that the CNN architecture can distinguish signal variations caused by human motions from random noises and wireless interferences. To further improve the network generability to variations including walking speed changes and additional noises, we generate augmentation data in the training dataset, which helps the system better learn the intra-class invariance in the data. Utilizing wireless signals from multiple sender-receiver pairs, we design a dynamic time warping based algorithm to detect the pedestrian's walking directions. We implement our system on a low-cost embedded platform. In an experiment with 163 walking events, we show that our system can detect walking events with over 94.5% of accuracy. In the future work, we plan to explore extending the current system for the multi-person detection. By differentiating the Wi-Fi RSSI signal perturbation caused by different number of pedestrians, many new applications, such as traffic flow monitoring and crowd density estimation can be achieved.

ACKNOWLEDGMENT

This work is funded by NSF CNS 1553273 and CNS 1463722.

REFERENCES

[1] [n. d.]. Aircrack-ng. https://www.aircrack-ng.org/index.html. Accessed: 2017-07-09.

[2] [n. d.]. Linux 802.11n CSI Tool. https://dhalperi.github.io/linux-80211n-csitool/. Accessed: 2017-07-09.

[3] [n. d.]. Radiotap. https://www.radiotap.org. Accessed: 2017-07-09.

[4] [n. d.]. RSSI in Wi-Fi. https://en.wikipedia.org/wiki/Received_signal_strength_indication. Accessed: 2018-02-09.

[5] [n. d.]. Wavelet Transform. https://en.wikipedia.org/wiki/Continuous_wavelet_transform. Accessed: 2018-02-09.

[6] Richard W Bohannon. 1997. Comfortable and maximum walking speed of adults aged 20 to 79 years: reference values and determinants. *Age and ageing* 26, 1 (1997), 15–19.

[7] Xiaodong Cui, Vaibhava Goel, and Brian Kingsbury. 2015. Data augmentation for deep neural network acoustic modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23, 9 (2015), 1469–1477.

[8] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. PMLR.

[9] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

[10] Ossi Kaltiokallio, Maurizio Bocca, and Neal Patwari. 2012. Enhancing the accuracy of radio tomographic imaging using channel diversity. In *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*. IEEE, 254–262.

[11] Mohammad A Kanso and Michael G Rabbat. 2009. Compressed RF tomography for wireless sensor networks: Centralized and decentralized approaches. In *Distributed Computing in Sensor Systems*. Springer, 173–186.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[13] Arthur Le Guennec, Simon Malinowski, and Romain Tavenard. 2016. Data augmentation for time series classification using convolutional neural networks. In *ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data*.

[14] G. Lui, T. Gallagher, B. Li, A. G. Dempster, and C. Rizos. 2011. Differences in RSSI readings made by different Wi-Fi chipsets: A limitation of WLAN localization. In *2011 International Conference on Localization and GNSS (ICL-GNSS)*. 53–57. https://doi.org/10.1109/ICL-GNSS.2011.5955283

[15] May Moussa and Moustafa Youssef. 2009. Smart cevices for smart environments: Device-free passive detection in real environments. In *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*. IEEE, 1–6.

[16] Neal Patwari and Piyush Agrawal. 2008. Effects of correlated shadowing: Connectivity, localization, and RF tomography. In *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on*. IEEE, 82–93.

[17] Muneeba Raja and Stephan Sigg. 2016. Applicability of RF-based methods for emotion recognition: A survey. In *Pervasive Computing and Communication Workshops (PerCom Workshops), 2016 IEEE International Conference on*. IEEE, 1–6.

[18] Pavel Senin. 2008. Dynamic time warping algorithm review. *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA* 855 (2008), 1–23.

[19] Kannan Srinivasan, Maria A Kazandjieva, Saatvik Agarwal, and Philip Levis. 2008. The β-factor: measuring wireless link burstiness. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 29–42.

[20] Ju Wang, Hongbo Jiang, Jie Xiong, Kyle Jamieson, Xiaojiang Chen, Dingyi Fang, and Binbin Xie. 2016. LiFS: low human-effort, device-free localization with fine-grained subcarrier information. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 243–256.

[21] Jie Wang, Xiao Zhang, Qinghua Gao, Xiaorui Ma, Xueyan Feng, and Hongyu Wang. 2017. Device-free simultaneous wireless localization and activity recognition with wavelet feature. *IEEE Transactions on Vehicular Technology* 66, 2 (2017), 1659–1669.

[22] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2017. Device-free human activity recognition using commercial WiFi devices. *IEEE Journal on Selected Areas in Communications* 35, 5 (2017), 1118–1131.

[23] Xuyu Wang, Lingjun Gao, Shiwen Mao, and Santosh Pandey. 2015. DeepFi: Deep learning for indoor fingerprinting using channel state information. In *Wireless Communications and Networking Conference (WCNC), 2015 IEEE*. IEEE, 1666–1671.

[24] Xuyu Wang, Chao Yang, and Shiwen Mao. 2017. PhaseBeat: Exploiting CSI phase data for vital sign monitoring with commodity WiFi devices. In *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*. IEEE, 1230–1239.

[25] Zhiguang Wang, Weizhong Yan, and Tim Oates. 2017. Time series classification from scratch with deep neural networks: A strong baseline. In *Neural Networks (IJCNN), 2017 International Joint Conference on*. IEEE, 1578–1585.

[26] Joey Wilson and Neal Patwari. 2010. Radio tomographic imaging with wireless networks. *Mobile Computing, IEEE Transactions on* 9, 5 (2010), 621–632.

[27] Kristen Woyach, Daniele Puccinelli, and Martin Haenggi. 2006. Sensorless sensing in wireless networks: Implementation and measurements. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, 2006 4th International Symposium on*. IEEE, 1–8.

[28] Kaishun Wu, Jiang Xiao, Youwen Yi, Dihu Chen, Xiaonan Luo, and Lionel M Ni. 2013. CSI-based indoor localization. *IEEE Transactions on Parallel and Distributed Systems* 24, 7 (2013), 1300–1309.

[29] Jiang Xiao, Kaishun Wu, Youwen Yi, and Lionel M Ni. 2012. FIFS: Fine-grained indoor fingerprinting system. In *Computer Communications and Networks (ICCCN), 2012 21st International Conference on*. IEEE, 1–7.

[30] Jie Yang, Yong Ge, Hui Xiong, Yingying Chen, and Hongbo Liu. 2010. Performing joint learning for passive intrusion detection in pervasive wireless environments. In *INFOCOM, 2010 Proceedings IEEE*. IEEE, 1–9.

[31] Moustafa Youssef, Matthew Mah, and Ashok Agrawala. 2007. Challenges: device-free passive localization for wireless environments. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*. ACM, 222–229.

[32] Dian Zhang, Jian Ma, Quanbin Chen, and Lionel M Ni. 2007. An RF-based system for tracking transceiver-free objects. In *Pervasive Computing and Communications, 2007. PerCom'07. Fifth Annual IEEE International Conference on*. IEEE, 135–144.

[33] Mingmin Zhao, Fadel Adib, and Dina Katabi. 2016. Emotion recognition using wireless signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 95–108.

[34] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2921–2929.