Pushing the Limit of CSI-based Activity Recognition: An Enhanced Approach via Packet Reconstruction

Youjing Lu†, Fan Wu†, Shaojie Tang§, Linghe Kong†, and Guihai Chen†
†Shanghai Jiao Tong University, China
§University of Texas at Dallas, USA
{luyoujing, linghe.kong}@sjtu.edu.cn, {fwu,gchen}@cs.sjtu.edu.cn, tangshaojie@gmail.com

***Abstract*—Fine-grained and complete Channel State Information (CSI) is essential for emerging CSI-based activity recognition applications. However, many probe packets collected for CSI measurements may be lost due to co-channel interferences and other malfunctions in practice, such as link interruptions, and thus limit the further applications of these CSI-based activity recognitions. To overcome this limitation, we propose an IMproved cOmpressive Sensing bAsed mIssing packet reCovery method, named IMOSAIC, to locate the lost probe packets and to reconstruct the missing CSIs, and thus improve the accuracy and the robustness of CSI-based activity recognitions. The key idea is to trace the probe packet flow to locate the positions of lost packets, derive the CSI Matrix from CSI measurements, and use improved compressive sensing technique to reconstruct the missing CSIs. We mainly address challenges in locating the lost packets, transforming CSI measurements into CSI Matrix, and digging up CSI measurement correlations and inherent low-rank properties to reconstruct the lost packets. Furthermore, experiment results show that IMOSAIC outperforms existing interpolation methods on reconstructing the lost packets, and can achieve an average recovery accuracy of 80.21%, when 90% of packets are lost, and the reconstructed CSI datasets can improve the activity recognition accuracy obviously.**

***Index Terms*—Channel State Information (CSI), activity recognition, compressive sensing**

I. Introduction

In recent years, studies on channel state information (CSI) have produced a large number of new applications, e.g., activity recognition, indoor positioning, cryptographic key extraction [1]–[21]. These applications are the foundations of virtual reality, augmented reality, secure communication, health care, smart homes, etc. In 2011, Halperin et al. modified the wireless driver of a network interface card (NIC) and released a CSI tool, which can record detailed measurements of the wireless channel along with received probe packet traces [22]. Since then, various CSI-based applications have experienced explosive growth. Activity recognition is one of the hottest research directions [6]–[9], [17]–[19], [21].

More specifically, we present an example to illustrate a general procedure of CSI-based activity recognition. Please refer to Fig. 1. The WiFi signal emitted by the transmitter is propagated via multi-path by scattering, reflection, and diffraction [23]. The movements of humans, as changing reflectors, affect the signal propagation paths, which are manifested and lead to the multi-path superimposed signal at the receiver side. Different activities and movements produce different multi-path superimposed signal effects. Thus, different activities and movements have different features, which can be expressed as different CSI waveform patterns. Existing activity recognition works exploit the waveform change patterns to profile and to recognize different activities, such as keystroke [6], arm gestures [7], gaits [8], and human presence [9]. The receiver (laptop), installed the CSI tool and equipped with a special WiFi NIC such as the Intel 5300, can collect the changing CSIs, which are used for activity recognitions.

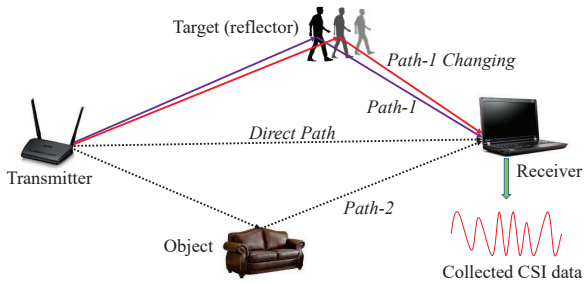


Fig. 1. A general procedure of CSI-based activity recognition.

However, collecting CSIs for activity recognition faces a significant concern, i.e., the loss of probe packets in practical environments. Some practical environments, such as office buildings, shopping malls, and residential areas, channel interferences are likely to happen from various kinds of signal sources. For instance, many deployed wireless access points heavily interfere with each other, and thus cause severe packet losses, which inevitably reduces the collection of CSI measurements and degrades the system performance of activity recognitions.

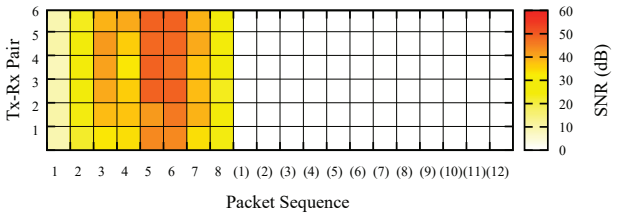
To tackle the problem caused by packet loss and to improve the robustness of CSI-based activity recognitions, we attempt to reconstruct the lost probe packets, which also record the CSI affected by human activities. There are mainly two challenges: the first is how to find the precise sequence number (position) of the lost packets, and the second is how to reconstruct the lost packets as accurately as possible after finding their positions.

Locating Lost Packets: We illustrate the lost packet lo-

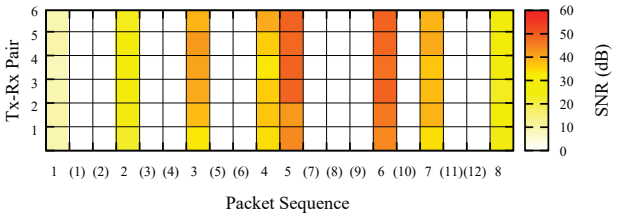
This work was supported in part by the National Key R&D Program of China 2018YFB1004703, in part by China NSF grant 61672348 and 61672353, in part by Supported by the Open Project Program of the State Key Laboratory of Mathematical Engineering and Advanced Computing 2018A09, and in part by Alibaba Group through Alibaba Innovation Research Program. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

F. Wu is the corresponding author.

978-1-7281-1207-7/19/$31.00 ©2019 IEEE



(a) Collected probe packets with interference



(b) Locations of lost probe packets

Fig. 2. The collected probe packets and the locations of the lost probe packets.

cations in Fig. 2. The X-axis denotes the sequence number of received packets, and the Y-axis denotes the 2×3 Tx-Rx pairs. Using Multiple-Input Multiple-Output (MIMO) technique, there are 2 transmitting antennas, 3 receiving antennas, and thus we have 2×3 Tx-Rx pairs. Each element represents the average CSI of a Tx-Rx pair in a probe packet. The heat of each element represents the Signal-to-noise ratio (SNR). As shown in Fig. 2(a), during a certain period of time, it is expected to collect 20 probe packets. However, only 8 packets are collected, thus, 12 packets are lost. The collected packets are stored in local and sorted by the time of arrival, i.e., the 8 collected packets are put together one by one as shown in Fig. 2(a). Other lost packets, e.g., the 12 lost packets shown on the right in Fig. 2(a), we do not know their locations at the time of arrival. To reconstruct these lost packets, we first have to determine their locations. For example, the actual locations of the two lost packets "(1)" and "(2)" in Fig. 2(a) are between the collected packets "1" and "2" in Fig. 2(b).

Reconstructing Lost Packets: After locating the lost probe packets, it is still highly non-trivial to reconstruct them. We notice that there are many interpolation methods to recover the missing data. However, those interpolation methods do not take the characteristic of the CSI variations into consideration. More specifically, there are many sudden changes, such as rises and falls in the CSI measurements, due to the multi-path interference of WiFi signals. Interpolation methods, such as K-nearest Neighbors (KNN) [24], Delaunay Triangulation (DT) [25], and Multi-channel Singular Spectrum Analysis (MSSA) [26], focus on the relations among adjacent points and overall change trends but not on abrupt changes. Therefore, they underperform in terms of accuracy in recovering lost packets. It is difficult to interpolate these values of abrupt changes. Thus, we must find an appropriate method to accurately reconstruct the lost packets.

To address the challenges of locating and reconstructing the lost packets, we propose a novel IMproved cOmpressive Sensing-bAsed lost packet reCovery method, namely IMOSAIC. The main idea is to trace the probe packet protocol flow to analyze the locations of lost packets, derive the CSI matrix from CSI measurements, and then use the improved compressive sensing technique to reconstruct them.

The main contributions of this paper are summarized as follows:

- We study the packet loss impact on the activity recognition performance. Next we derive the CSI matrix from CSI measurements, and then find the training sample correlations and inherent low-rank structures of the CSI measurements. Based on these observed characteristics, we develop an improved compressive sensing based algorithm, IMOSAIC, to estimate the lost packets.
- Our algorithm is evaluated using real-world datasets collected from several different activity recognitions, including keystroke recognition, arm gesture recognition, gait recognition, and human detection. The evaluation results show that IMOSAIC outperforms other interpolation methods on the lost packet reconstruction. In particular, when 90% of the packets are lost, IMOSAIC can still achieve as high as 80.21% reconstruction accuracy.

The remainder of this paper is organized as follows. Section II presents the related work. Section III introduces the preliminary works. Section IV analyzes the lost packet location and derives CSI Matrix from CSI measurements. Section V investigates and utilizes the low-rank structure and inherent correlations of CSI measurements to reconstruct the lost packets. Section VI shows the evaluation results. Section VII concludes the paper.

II. Related Works

The most closely related works can be classified into three categories: improving the accuracy of CSI measurements, interpolating missing data, and using compressive sensing technique in wireless networks.

For the first category, a number of works have been proposed to improve the accuracy of CSI measurements. Some of them report phase rotation errors caused by Sampling Frequency Offset (SFO) [27], Carrier Frequency Offset (CFO) [28], Packet Detection Delay (PDD) [29], etc. Other methods propose to calibrate the CSI phase error by searching for linear or non-linear fits and subtracting a linear fit from the raw CSI phase [30], [31]. However, none of them can address the problem of packet loss.

Concerning the second category, there is a rich body of works on data interpolation. The K-Nearest-Neighbor (KNN) method is a typical local interpolation method [24]. It simply uses the focal value's k nearest neighbors to estimate the missing value but at low accuracy. Delaunay Triangulation (DT) is a distinctive interpolation method that takes collected data as vertices [25]. It utilizes these vertices and their global errors to construct virtual triangles for interpolating missing data. The multi-channel Singular Spectrum (MSSA) method is based on the embedded lag-covariance matrix with data adaption [26]. MSSA is frequently used in meteorological data and geographic data reconstruction. However, current methods do not take the characteristic of CSI data into consideration and underperform when the packets loss rate is high.

In the third category, the compressive sensing (CS) technique is used in different aspects of wireless networks. Kuo et al. used CS to reduce the burden of CSI feedback in MIMO

systems [32]. Gao et al. used structured CS to reduce pilot overhead [33]. Bajwa et al. used CS for channel estimation [34]. Yu et al. used CS to perform wideband spectrum sensing [35]. CS is also widely used to recover a low-rank matrix with a few data points [36].

III. Preliminaries

In this section, we introduce some preliminaries.

In a general CSI-based activity recognition system, the users collect ICMP packets for every type of activity repeatedly as training sets in advance, and extract features from these training sets for every type of activity respectively. Please note that the ICMP packet is the probe packet in activity recognitions. During training set building, there are some similar collected training samples corresponding to the same type of activity. For a type of activity, we suppose the number of corresponding training samples is n, and each training sample includes m probe packets. Subsequently, these training samples can be composed of an $n \times m$ matrix. Actually, a probe packet corresponds to 6 Tx-Rx pairs, and each pair includes 30 subcarriers because of Orthogonal Frequency Division Multiplexing (OFDM) and Multiple-Input Multiple-Output(MIMO) techniques. Thus, the size of the tensor is $6 \times 30 \times n \times m$. For ease of presentation, we simply use one of the subcarriers to present an entry of the tensor. Then, the size of the tensor returns to $n \times m$. Please note that we actually process 6×30 such tensors (matrices) for a type of activity in experiments.

For ease of presentation, we define the following matrices:

- **CSI Matrix (CSIM)**: The CSIM is defined as $P = (p(i,j))$, with n rows and m columns, where $i = 1, 2, \cdots, n$ and $j = 1, 2, \cdots, m$. Information about CSIM is often incomplete due to packet loss.
- **Binary Index Matrix (BIM)**: The BIM is an $n \times m$ matrix that indicates if an entry (packet data point) in CSIM is missing. The BIM is defined by

$$B = (b(i,j))_{n \times m} = \begin{cases} 0 & \text{if } p(i,j) \text{ is missing,} \\ 1 & \text{otherwise.} \end{cases} \quad (1)$$

- **Collected CSI Matrix (CCM)**: The CCM is an $n \times m$ matrix that stores the packets collected from practical scenarios. The CCM is denoted by I, which can be represented by the element-wise production of P and B:

$$I = P \cdot B. \quad (2)$$

Our goal is to reconstruct the CSIM based on available packets. To present and measure the effectiveness of interpolation methods in different scenarios, we first introduce some definitions as follows:

- **Reconstructed CSI Matrix (RCM)**: The RCM is generated by interpolating the missing values of CCM to approximate to the CSIM. The RCM is denoted by $\hat{P} = (\hat{p}(i,j))_{n \times m}$.
- **Correct Ratio (CR)**: The CR is a metric for measuring the reconstruction correctness after interpolation:

$$\theta = \frac{\sqrt{\sum_{i,j:b(i,j)=0} (\hat{p}(i,j))^2}}{\sqrt{\sum_{i,j:b(i,j)=0} (p(i,j))^2}}. \quad (3)$$

4532.023159000	10.181.86.181	10.181.86.33	ICMP	seq=17801
4532.023278000	10.181.86.181	10.181.86.33	ICMP	seq=17802
4532.023386000	10.181.86.181	10.181.86.33	ICMP	seq=17804
4532.023453000	10.181.86.181	10.181.86.33	ICMP	seq=17805
4532.023524000	10.181.86.181	10.181.86.33	ICMP	seq=17806
4532.023617000	10.181.86.181	10.181.86.33	ICMP	seq=17807
4532.023865000	10.181.86.181	10.181.86.33	ICMP	seq=17809
4532.023923000	10.181.86.181	10.181.86.33	ICMP	seq=17810
4532.024106000	10.181.86.181	10.181.86.33	ICMP	seq=17811
4532.024348000	10.181.86.181	10.181.86.33	ICMP	seq=17814

Fig. 3. The locations of lost packets in terms of ICMP sequence number.

Please note that the condition $b(i,j) = 0$ in Eqn. (3) indicates the correctness in terms of the counted lost packets.

- **Accuracy Ratio (AR)**: The AR measures the effectiveness of interpolation methods on activity recognitions. It is defined by $\sigma = \frac{\alpha}{\beta}$, where α is the activity recognition accuracy with reconstructed CSI measurements, and β is the activity recognition accuracy with complete CSI measurements.

Problem: Lost Packet Reconstruction (LPR): Given a CCM I, the LPR problem is to find an optimal RCM $\hat{P}$ that can best approximate to CSIM P. The problem can be formulated as follow,

$$\begin{aligned} \text{Objective} &: \min \parallel P - \hat{P} \parallel_F, \\ \text{Subject to} &: I. \end{aligned} \quad (4)$$

where $\parallel \cdot \parallel_F$ is the Frobenius norm used to measure the error between the matrix P and $\hat{P}$. The objective of the LPR problem is to minimize the absolute error.

IV. Tracing Packets and Deriving CSI Matrix

In this section, we present the methods of tracing the protocol flow to locate the lost packets, and deriving the CSI Matrix from the CSI measurements.

A. Tracing Lost Packets

To locate the lost packets, the basic idea is to trace the ICMP protocol flow to analyze the locations of lost packets in time series. CSI measurements are recorded instantaneously in probe packets, which are transmitted by the ICMP protocol. Thereby, we can analyze the sequence of the collected ICMP packet series to find the packets that do not arrive successfully. To find the lost packets, we filter the ICMP packets with Wireshark. In the process of collecting packets, the receiver is continuously pinging the transmitter, and the command is similar to "sudo ping -i r 10.181.XX.XX", where "r" is the rate of pinging and "10.181.XX.XX" is the IP address of receiver.

As shown in Fig. 3, a packet is discarded between "seq= 17802" and "seq= 17804", as well as between "seq= 17807" and "seq= 17809". Then, we can locate the sequence numbers "17803", "17808", etc., which will be marked as lost packet locations.

B. Deriving CSI Matrix

After locating the lost packets, we need to process the CSI measurements to derive the CSI Matrix. To achieve accurate activity recognition, users must collect a number of training samples for feature extraction. These training samples are the foundation for the next profiling and matching steps.

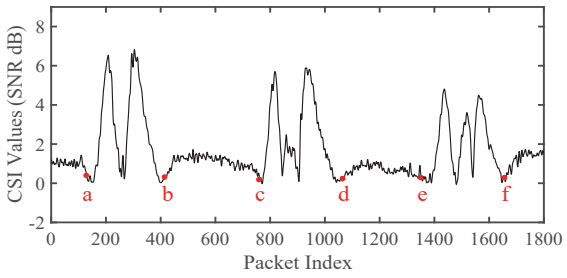


Fig. 4. The CSI waveform for typing letter "k" repeatedly.

To recover those lost packets, we transform the training samples into an $n \times m$ CSI Matrix (CSIM) at first. More specifically, we use an example to illustrate the procedure of deriving the CSIM from the training samples. For keystroke recognition, users collect CSI training samples by pressing the letter on the keyboard. The training samples associated with a certain letter are all composed of probe packets, which are collected when the user presses the particular letter n times. Thus, we can get n training samples for each letter. We know that pressing a letter once generates a training sample and thus a row in the CSIM. Thereby, pressing the same letter n times generates n rows in the CSIM. Suppose that the length of each training sample is m, then we can get an $n \times m$ CSIM for every type of activities.

Before deriving the CSIM, we need to get the training samples for the activities. For example, after we press letter "k" repeatedly for several times, we get a series of probe packets. The recorded CSI can be shown in Fig. 4. The intervals, such as $a \sim b$, $c \sim d$, and $e \sim f$ are the corresponding training samples to letter "k". To get these training samples, we have to properly intercept the packet series.

To get the accurate training samples, we need to find the starting and ending points of every training sample generated by the same activity. We know that the starting and ending points of a training sample have obvious increasing and decreasing trends, e.g., point "a" and point "b", as shown in Fig. 4. Referring to [6], we use a moving window approach to detect the increasing and decreasing trends to find accurate starting and ending points. In this approach, we calculate the mean absolute deviation for each window of size W at y-th iteration, i.e.,

$$\Delta M_y = \frac{\sum_{x=y}^{y+W} \left| Z(x) - \bar{Z}(y : y+W) \right|}{W}, \tag{5}$$

where y is the serial number of a packet; $Z(x)$ is the SNR of packet x; $\bar{Z}(y : y+W)$ is the means of SNR in the y-th window. Then we set a threshold to decide whether the current window contains significant variations in CSI amplitudes. By doing so, we are able to decide the starting and ending points of the CSI wave generated from the same activity.

However, a challenge in building CSIMs is that the lengths of training samples could be different. This is mainly because the durations for every time of activity are different, e.g., when a user presses the same letter for several times, the time of duration for every pressing can vary to some extent. To overcome this challenge and ensure that all training samples have identical lengths, we use the average length of available training samples to represent the number of columns in CSIM. To get as possible as available training samples, the first 5 training samples are discarded. For example, the average length of training samples is m. Assume that the original length of samples is l, the last $l-m$ packets will be discarded when $l \geq m$; the last $m-l$ packets will be marked as lost packets when $l < m$. These operations can provide a tradeoff between retaining more CSI and reducing the redundancy in CSI measurements.

V. Reconstruction of Lost Packets

In this section, we analyze the low rank structure and the correlations of CSI measurements. Then, we design an improved compressive sensing algorithm to reconstruct these lost packets in the training set.

A. Inherent Low Rank Structure

By analyzing the features of collected packets for training samples, we find that CSIMs have inherent low rank property and training sample correlations. As mentioned above, we know that there are training samples corresponding to a certain type of activity and that they have similar CSI waveforms. Naturally, we wonder wheter there exists rank property or redundancy in the training samples. To validate the inherent low rank property and redundancy in these training samples, we utilize the Principal Component Analysis (PCA) technique, which is an effective method for investigating the rank and inherent structures in a dataset [37].

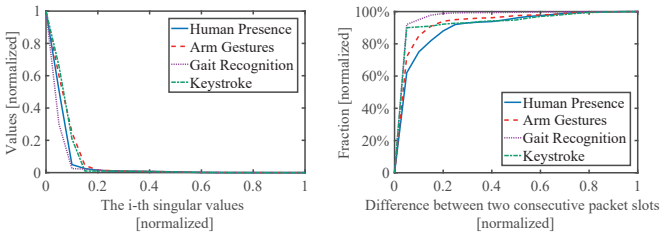
According to the Singular Value Decomposition (SVD) method, an $n \times m$ matrix P can be decomposed into three parts:

$$P = U\Gamma V^T = \sum_{i=1}^{\min(n,m)} \sigma_i u_i v_i^T, \tag{6}$$

where U is an $n \times n$ unitary matrix ($i.e.$, $UU^T = U^TU$) and V^T is the transpose of V, which is an $m \times m$ unitary matrix ($i.e.$, $VV^T = V^TV$). Γ is an $n \times m$ diagonal matrix, containing the singular values σ_i of P. Generally, the singular values along the diagonal are sorted in descending order, $i.e.$, $\sigma_{i+1} < \sigma_i$, $i = 1, 2, \ldots \min(n,m)$, where $\min(n,m)$ is the number of singular values in Γ. The number of non-zero singular values is the rank r of a matrix, and the matrix has a low-rank structure when $r << \min(n,m)$.

In Eqn. (6), the energy of the i-th principal component is presented by the singular value σ_i. The sum of all singular values is equal to $\sum_{i=1}^{\min(n,m)} \sigma_i$. According to PCA, the first r singular values in a low-rank matrix contain all or almost all the energy $\sum_{i=1}^{r} \sigma_i = \sum_{i=1}^{\min(n,m)} \sigma_i$.

In Fig. 5 (a), we depict the distribution of the singular values of 4 different CSIMs from 4 different types of activity recognitions. They are human presence detection, arm gesture recognition, gait recognition, and keystroke recognitions [6]–[9]. The X-axis denotes the i-th singular values. We normalize the X-axis because the scales of the 4 CSIMs are different. Then, $\min(n,m)$ of each CSIM is normalized to 1. The Y-axis denotes the value of the i-th singular values. The Y-axis is also normalized for the same reason as the X-axis. The $\max(\sigma_i)$ of each CSIM is normalized to 100%. This figure



(a) Inherent low rank structure feature. (b) Training sample correlation feature.

Fig. 5. The features of CSI measurements.

demonstrates that those largest singular values always posses almost all the energy in the CSIM.

The obvious inherent low rank structures are exhibited by the conditions $\sum_{i=1}^{r}\sigma_i \approx \sum_{i=1}^{\min(n,m)}\sigma_i$ and $r << \min(n,m)$ holding. The inherent low-rank structures are the basic conditions of utilizing the CS technique [38].

B. Training Sample Correlations

Recall that when the same type of activity is repeated multiple times, there exists a strong correlation among those training samples. Based on this intrinsic phenomenon, we analyze the CSIMs in the sample dimension to reveal the sample correlations.

We measure the correlation on the training sample i at packet point j by computing the normalized difference $\Delta c(i,j)$ between adjacent packet slots in the same column:

$$\Delta c(i,j) = \frac{|c(i,j) - c(i-1,j)|}{\max(|c(I,J) - c(I-1,J)|)}, \tag{7}$$

where I varies from 1 to n, J varies from 1 to m, and $max(|c(I,J) - c(I-1,J)|)$ is the maximal difference between any two consecutive packet slots in the same column of the CSIM.

Fig. 5 (b) plots the CDF of $\Delta c(i,j)$. The X-axis denotes the difference between two consecutive packet slots $\Delta c(i,j)$. The Y-axis denotes the cumulative probability. We observe that the cumulative probability $> 75\%$ in the gait recognition, $> 72\%$ in the arm gestures, $> 65\%$ in keystroke, and $> 60\%$ in human presence activities. The value of $\Delta c(i,j)$ represents the difference degree between two consecutive packet slots. In addition, near-total ($> 95\%$) $\Delta c(i,j)$ are very small (< 0.08) in the gait recognition and arm gesture datasets. Even in the worst case scenario, the packet values between two consecutive packet slots mostly ($> 75\%$) change only slightly (< 0.35). These results demonstrate that training sample correlation exists in real activity recognition tasks. Motivated by this observation, we can adopt the training sample correlation to optimize the CS-based estimation of lost packets.

C. Compressive Sensing Based Recovery

We have validated the inherent low-rank structure of the CSIM, next we attempt to use a compressive sensing technique to recover those lost packets. Recall that our goal is to estimate $\hat{P}$. According to Eqn. (6), the SVD technique can decompose any matrix into $\sum_{i=1}^{\min(n,m)}\sigma_i u_i v_i^T$. After the inverse process, we can use the r largest singular values to create an r-rank approximation $\hat{P}$:

$$\sum_{i=1}^{\min(n,m)} \sigma_i u_i v_i^T = \hat{P}. \tag{8}$$

We know that $\hat{P}$ is the top best r-rank approximation that minimizes the error measured by the Frobenius norm. However, the optimal $\hat{P}$ cannot be derived directly in this way, because we do not know P or the appropriate rank in advance. Thus, we propose to compute $\hat{P}$ as follows:

$$\begin{aligned} \text{Objective} &: \min(rank(\hat{P})), \\ \text{Subject to} &: B \cdot \hat{P} = I. \end{aligned} \tag{9}$$

There are two motivations behind the above formulation. The first motivation is that the RCM is generated from the CCM. The second motivation is that the RCM should also have inherent low-rank structures such as the CSIM. However, it is still difficult to solve the minimization problem, which is non-convex. To overcome this challenge, we exploit the SVD-like factorization, rewritten by Eqn. (6) as follows:

$$\hat{P} = U\Gamma V^T = \Phi\Psi^T, \tag{10}$$

where $\Phi = U\Gamma^{1/2}$ and $\Psi = V\Gamma^{1/2}$. Replacing Eqn. (10) by Eqn. (9), we can solve the minimization problem based on the CS theory in [36]. Particularly, if the restricted isometry is achieved [39], minimizing the nuclear norm can be deduced as minimizing the rank of a low rank matrix. Thus, we only need to find the matrix Φ and matrix Ψ that minimize the summation of their Frobenius norms:

$$\begin{aligned} \text{Objective} &: \min(\|\Phi\|_F^2 + \left\|\Psi^T\right\|_F^2), \\ \text{Subject to} &: B \cdot (\Phi\Psi^T) = I. \end{aligned} \tag{11}$$

It is often very difficult to find Φ and Ψ that strictly satisfy Eqn. (11). This is mainly due to the approximate low rank of CSIMs, furthermore, the noise in CSI may lead to an overfitting problem when Eqn. (11) is strictly satisfied. Hereby, we solve the following equation by directly utilizing the Lagrange multiplier method and substituting Eqn. (11):

$$\min(\left\|B \cdot (\Phi\Psi^T) - I\right\|_F^2 + \lambda(\|\Phi\|_F^2 + \left\|\Psi^T\right\|_F^2)), \tag{12}$$

where the Lagrange multiplier λ can be tuned to obtain a tradeoff between rank minimization and accuracy fitness. A low-rank approximation with not strict satisfaction is provided in the above solution.

In Eqn. (12), 1) B and I are known, 2) any $\|\cdot\|_F^2$ is non-negative, and 3) minimizing all non-negative parts obtains the optimal values that approximate 0. Accordingly, by tuning λ, Φ and Ψ can be estimated under this optimization.

D. Improved Compressive Sensing

In this section, we present an improved CS-based lost packet recovery approach, called IMOSAIC. To improve the performance of lost packet reconstruction, we first transform the CSIM into a new matrix to remove a portion of noise caused by measurements. Then, we utilize the correlations among training samples to facilitate the objective of the CS formula and to reconstruct the lost packets more accurately.

Denoising the CSIM: We perform a special linear transformation on the CSIM to remove the measurement noise. To illustrate it formally, we define the CSIM as P with size $n \times m$. The transformation follows the following steps:

Step 1: Choose k rows randomly from P to generate k vectors $\{D_{(1)}, D_{(2)}, \cdots, D_{(k)}\}$.

Step 2: Record the missing entries of every vector in $\{D_{(1)}, D_{(2)}, \cdots, D_{(k)}\}$. For instance, if the first and third entry of $D_{(1)}$ are missing, we record $S_{(1)} = \{1, 3\}$; if the second, forth, and fifth entry of $D_{(2)}$ are missing, we record $S_{(2)} = \{2, 4, 5\}$. Then, we perform the same operations on the remaining $k-2$ vectors to generate $\{S_{(3)}, \cdots, S_{(k)}\}$. Next, we obtain the missing positions set $S = S_{(1)} \cup S_{(2)} \cup \cdots \cup S_{(k)}$. Suppose that $|S| = q$.

Step 3: We generate a new length-n vector G . The elements $\{G_{S_1}, G_{S_2}, \cdots, G_{S_q}\}$ in G are marked as missing entries. For each element G_i in G, if G_i is not marked as a missing entry, then $G_i = \frac{\sum_{j=1}^{k} D_{(j)_i}}{k}$.

Step 4: Delete one row in P randomly, and add G as a row into P. Then, we skip to Step 1 to apply the above operations on P repeatedly for t times.

This special transformation can remove some measurement noise by averaging the k vectors. In addition, it does not increase the rank of the CSIM by transforming linearly and deleting random rows.

After denoising the CSIM, IMOSAIC leverages the training sample correlations to improve the estimation accuracy of lost packets. In particular, we extend Eqn. (12) by considering the correlations among training samples:

$$\min(\left\|B \cdot (\Phi\Psi^T) - I\right\|_F^2 + \lambda(\|\Phi\|_F^2 + \left\|\Psi^T\right\|_F^2) + \left\|\Phi\Psi^T\mathbb{C}^T\right\|_F^2), \tag{13}$$

where $\mathbb{C}$ is the sample correlation matrix. We set $\left\|\Phi\Psi^T\mathbb{C}^T\right\|_F^2$ and $\left\|B \cdot (\Phi\Psi^T) - I\right\|_F^2$ equally to a similar order of magnitude or else they could overshadow the other when solving Eqn. (13).

Training sample correlation improvement: The sample correlation constraint matrix $\mathbb{C}$ denotes the training sample correlation feature, which reveals that the change in packet values between the same positions in adjacent training samples is small. Hereby, we set $\mathbb{C} = Toeplitz(0, 1, -1)_{m \times m}$. The Toeplitz matrix is defined with a central diagonal given by 1; the first lower diagonal is all -1s, and the remaining terms are by 0, i.e.,

$$\mathbb{C} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \ddots & \vdots \\ 0 & -1 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix}_{m \times m}. \tag{14}$$

This Toeplitz matrix adds the sample correlation constraint to estimation. Introducing $\left\|\Phi\Psi^T\mathbb{C}\right\|_F^2$ into Eqn. (13) is equivalent to including an extra constraint into the initial optimization problem. Because the sample correlation constraint is a natural feature of the CSI training data, this extra constraint will remove more noise and errors measured in the estimation of $\left\|\Phi\Psi^T\right\|$.

Algorithm 1: IMOSAIC

Input: $I_{n\times m}$ (CSI matrix) $B_{n\times m}$ (binary index matrix) r (rank bound) λ (tradeoff coefficient) $NumIter$ (iteration times)
Output: $\hat{P}_{n\times m}$: estimated CSI matrix

```
Φ ← random_matrix(n, m);
for 1 to NumIter do
    Ψ ← CSinverse(I, B, r, λ, Φ);
    Φ ← CSinverse(I^T, B^T, r, λ, Φ^T);
    v ← ||B · (ΦΨ^T) − I||_F^2 + λ(||Φ||_F^2 + ||Ψ^T||_F^2) + ||ΦΨ^T C^T||_F^2;
    if v < v̂ then
        Φ̂ ← Φ; Ψ̂ ← Ψ; v̂ ← v;
P̂ ← Φ̂Ψ̂^T;
return P̂;
```

Algorithm 2: CSinverse(I, B, r, λ, Φ)

Input: $I_{n\times m}$ (CSI matrix) $B_{n\times m}$ (binary index matrix) r (rank bound) λ (tradeoff coefficient) Φ (SVD factor)
Output: Y

```
foreach i=1 to t do
    S_i ← [Diag(B_(i)) * Φ; √λ I_r];
    W_i ← [I_(i); 0_r];
    Y(:, i) = (S_i^T S_i) \ S_i^T W_i;
return Y;
```

Algorithm 1 shows the details of IMOSAIC. First, we scale the $\mathbb{C}$, as all $\|\cdot\|_F^2$ in Eqn. (13) have a similar order of magnitude. Therefore, they will not affect each other. The scaling technique is similar to [40]. Hence, IMOSAIC operates in an iterative manner. Φ is initialized randomly, and thus, Ψ can be computed by solving the following equation:

$$\begin{bmatrix} B \cdot (\Phi\Psi^T) \\ \sqrt{\lambda}\Psi^T \end{bmatrix} = \begin{bmatrix} I \\ 0 \end{bmatrix}. \tag{15}$$

This equation can be rewritten as:

$$\begin{bmatrix} Diag(B_{(i)}) * \Phi\Psi_{(i)}^T \\ \sqrt{\lambda}\Psi_{(i)}^T \end{bmatrix} = \begin{bmatrix} I_{(i)} \\ 0 \end{bmatrix}. \tag{16}$$

where i ranges from 1 to m. This is a combination of multiple standard linear least squares problems. We then have $R_{(i)}^T = (S_i^T S_i) \setminus S_i^T W_i$, where $S_i = [Diag(B_{(i)}) * \Phi; \sqrt{\lambda} I_r]$ and $W_i = [I_{(i)}; \mathbf{0}_r]$. This procedure is reflected by the subfunction *CSinverse* in Algorithm 2. On the other hand, Φ can also be computed by fixing Ψ. This process repeats until the optimal solution is achieved.

We analyze the complexity of the IMOSAIC algorithm. The crucial step is to compute the inverse matrix, providing the best approximate solution to the contradictory equation. This

step is completed by a matrix multiplication. Hence, the time complexity of the algorithm is $O(rnm)$. Because IMOSAIC repeats this step g times, the total complexity is $O(grnm)$.

The rank bound r and tradeoff coefficient λ are two parameters that influence the accuracy of $\hat{P}$ estimation. We adopt a genetic algorithm to explore appropriate values for those two parameters. Estimation errors serve as the fitness, which is stalled after several generations to derive the optimal parameters.

E. Analysis for Reconstructing Lost Packets in the Testing Set

We know that packet loss exists in both the training set and the testing set. The packets in the training set are collected in advance, and we can use improved CS to recover these lost packets off-line. However, for the real-time activity recognition, we need to process these lost packets in the testing set on-line. From the very start, we consider interpolation for these lost packets in the testing set. Given that we do not know which lost packets correspond to which type of activity, we cannot exploit the CS technique to recover them. Other interpolation methods, such as KNN, DT, and MSSA, do not consider abrupt changes, which exist in CSI waves pervasively. These interpolations introduce more errors and noises, degrading the performance of recognition systems. Therefore, we decide not to recover these lost packets in the testing set. To improve the performance of IMOSAIC, we will study the appropriate interpolation method for testing sets in the future work.

VI. Performance Evaluation

In this section, we evaluate the performance of IMOSAIC.

A. Experiment Methodology

For the ICMP packet collection, we employ a TP-LINK TL-WR842N wireless router with 2 antennas as the transmitter operating in IEEE 802.11n AP mode and a Lenovo X200 laptop with an Intel Link 5300 WiFi NIC with 3 antennas as the receiver. Next, we collect the ICMP packets to reproduce activity recognition systems, including human presence detection, various gait recognition tasks, arm gesture recognition, and keystroke recognition. The experimental configurations and parameter settings that we adopt refer to [6]–[9].

The evaluation requires complete CSIMs P to compute the correct ratio. The CSI training datasets are collected from keystroke recognition, arm gestures recognition, gait recognition, and human presence detection. In the phase of denoising the CSIM, we empirically set k and t as 5 and 7, respectively.

To verify the effectiveness of IMOSAIC, we choose four types of classical interpolation methods for comparison: K-Nearest Neighbor (KNN) [24], with K set to $\sum_{i=1}^{n} H_{(i)}/n$; Multi-channel Singular Spectrum (MSSA) [26], with parameter M set to 32; DT [25]; and CS [36]. The simulation steps include the following: 1) Generate the BIM B based on four missing packet patterns. 2) Compute the CCM I according to Eqn. (2). 3) Take the CCMs as input and generate the RCMs to test all interpolation methods. 4) The interpolation quality metric CR is computed for all methods and training datasets for the performance evaluation.

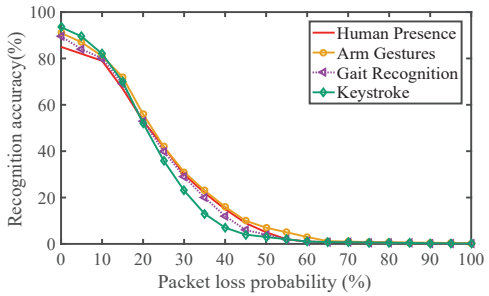


Fig. 6. Recognition accuracy of different activities with different packet loss probabilities.

Three series of experiments are evaluated. The first experiment evaluates the performance of four types of methods against general random packet loss probabilities. The second experiment compares the Accuracy Ratio (AR) between the reconstructed CSI packet dataset and the original CSI packets without interference to recognize human activities. The last experiment measures the performance on different loss packet patterns.

B. Accuracy of Activity Recognition with Lost Packets

We study the accuracy of activity recognitions with lost packets, including human presence detection, gait recognition, arm gesture recognition, and keystroke recognition. As shown in Fig. 6, the X-axis denotes the packet loss probability, and the Y-axis denotes the recognition accuracy. In Fig. 6, the recognition accuracy of every activity is high when the packet loss probability equals zero. We achieve this by turning off all the other wireless routers to ensure minimal interference. Then, we can see that the recognition accuracy drops dramatically with increasing packet loss probability. Taking the keystroke as an example, the recognition accuracy is lower than 50% when the packet loss probability is 25%. Therefore, we need to reconstruct the lost packets to have a robust performance on the recognition accuracy of different types of activities.

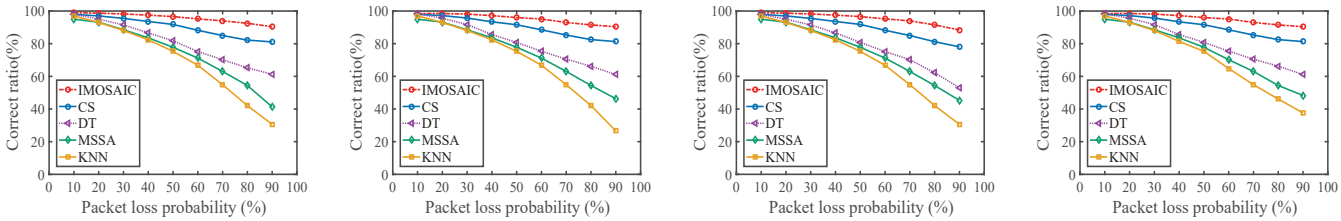
C. Correct Ratio Comparison

In the Correct Ratio (CR) comparison, we test the CR of different methods. The packet loss rate ranges from 10% to 90%. Fig. 7 shows the CR comparison results. The X-axis denotes the packet loss probability, and the Y-axis presents the value of CR. Generally, CR decreases with loss rate.

In CSI training dataset of keystroke recognition, IMOSAIC shows the best performance. Even when 90% packets are lost, IMOSAIC still can reconstruct the entire dataset with CR$\geq$90%, while the CR of CS is approximately 82%, that of DT is approximately 62%, and that of MSSA and KNN is less than 40%. In the CSI training datasets for arm gesture recognition, similarly, IMOSAIC also achieves the best performance among these interpolation methods. Due to the same nature of the CSI training samples in different activity recognitions, the performances of these five interpolation methods on different CSI datasets are very similar as indicated in Fig. 7.

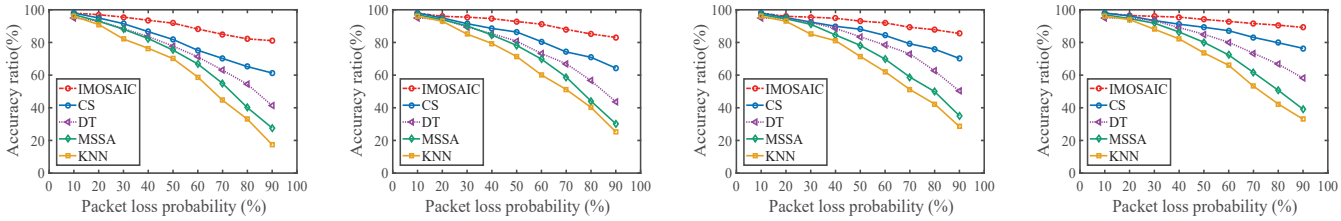
D. Accuracy Ratio Comparison

In the Accuracy Ratio (AR) comparison, we test the AR of different methods. The AR is a metric for measuring the effectiveness of interpolation methods on real CSI training data



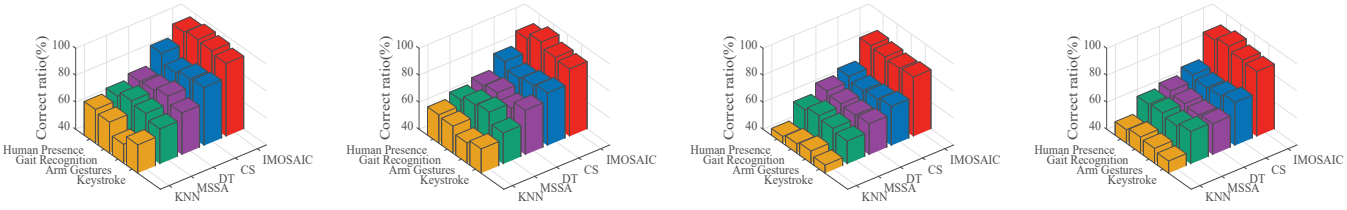
(a) Keystroke recognition (b) Arm gesture recognition (c) Gait recognition (d) Human presence detection

Fig. 7. Correct ratio of different interpolation methods for activity recognitions.



(a) Keystroke recognition (b) Arm gesture recognition (c) Gait recognition (d) Human presence detection

Fig. 8. Accuracy ratio of different interpolation methods for activity recognitions.



(a) Random loss (b) Frequent loss in row (c) Successive loss in row (d) Combinational loss

Fig. 9. Correct ratio of different interpolation methods in different packet loss patterns.

of different activity recognitions. It is defined as a ratio of the accuracy of the activity recognition using reconstructed CSI data and that using the raw data with different packet loss rates.

Fig. 8 shows the AR comparison results. The X-axis also denotes the packet loss probability, and the Y-axis presents the value of AR. Generally, AR decreases with increasing packet loss rate.

In the training dataset for keystroke recognition, as shown in Fig. 8 (a), IMOSAIC also outperforms other interpolation methods. When 90% of the CSI data are lost, IMOSAIC can still achieve an AR of at least 80%. The AR of CS is approximately 72%, and the AR of DT, MSSA, and KNN are less than 40%. We can see that vertically, in general, the AR is significantly lower than the corresponding CR. This is because a small error in CSI data is amplified in the process of activity recognition. Therefore, a lower CR indicates an even lower AR.

Horizontally, as shown in Fig. 8, the AR increases gradually in all recognitions, including keystroke recognition, arm gestures recognition, gait recognition and human presence detection. This is because the CSI data granularity required in different applications are different. In keystroke recognition, the amplitudes of fingers and arms are very small, and the changes in CSI data values are also very small. Therefore, it requires a higher data reconstruction quality. In contrast, the amplitudes of arms are larger in arm gesture recognition, and thus, the requirement could be lower.

E. Packet Loss Pattern Comparison

In the analysis of the lost packet locations, we find four typical patterns. The first is the lost packet are randomly distributed; the second is the lost packet are distributed frequently in rows; the third is the lost packet are distributed successively in rows; the last is the combination of above three patterns. We also evaluate the performance of IMOSAIC on the four different packet loss patterns.

Fig. 9 (a) shows the CR of different interpolation methods on random loss data. In data with random loss patterns, the data points are randomly selected, and 50% of packets are lost. We can see that the CR of IMOSAIC is greater than 95% on average, which is significantly higher than that of other interpolation methods. Through the comparison between CS and IMOSAIC, we can see that the training sample correlation optimization has a crucial contribution to the better performance of recovering the missing data points.

Fig. 9 (b) shows the CR of different interpolation methods on data with the pattern of frequent losses in row. For this kind of data, the rows are randomly selected, and 50% of all data are missing. Similar to the results in Fig. 9 (a), the CR

of IMOSAIC is also significantly higher than that of other interpolation methods.

Fig. 9 (c) shows the CR of different interpolation methods on data with the pattern of successive losses in row. The packet loss rate is again set as 50%. Their CRs are lower than the frequent loss in row pattern averagely. There are fewer constraints in the CSI training samples. The packets are lost successively in row; sometimes, all the elements in a row are lost.

Fig. 9 (d) shows the CR of different interpolation methods on data with a combinational loss pattern. We set the ratio of the above three loss patterns as 50%, 25%, and 25%. We can see the CR of IMOSAIC is still higher than 85% in the combinational loss packet pattern.

In summary, our IMOSAIC outperforms MSSA, DT, KNN, and CS on all packet loss patterns.

VII. Conclusion

In this paper, we have studied probe packet loss and reconstruction in CSI-based activity recognition systems. We first have found the locations of these lost packets and derived the CSI Matrix from the CSI measurements, and then we have dug up the training sample correlations and inherent low rank structures of CSI measurements. Motivated by these two observations, we have proposed the IMOSAIC algorithm to estimate these lost packets. The experiments demonstrate that the IMOSAIC algorithm outperforms other interpolation algorithms. Finally, IMOSAIC can achieve an accuracy ratio of 80.21% even when 90% of the packets are lost, thereby improving the robustness of activity recognitions.

References

[1] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: device-free location-oriented activity identification using fine-grained wifi signatures," in *Proceedings of ACM MobiCom*, 2014.

[2] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of wifi signal based human activity recognition," in *Proceedings of ACM MobiCom*, 2015.

[3] J. Wang, H. Jiang, J. Xiong, K. Jamieson, X. Chen, D. Fang, and B. Xie, "Lifs: low human-effort, device-free localization with fine-grained subcarrier information," in *Proceedings of ACM MobiCom*, 2016.

[4] X. Zheng, J. Wang, L. Shangguan, Z. Zhou, and Y. Liu, "Smokey: Ubiquitous smoking detection with commercial wifi infrastructures," in *Proceedings of IEEE INFOCOM*, 2016.

[5] W. Xi, C. Qian, J. Han, K. Zhao, S. Zhong, X.-Y. Li, and J. Zhao, "Instant and robust authentication and key agreement among mobile devices," in *Proceedings of ACM CCS*, 2016.

[6] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using wifi signals," in *Proceedings of ACM MobiCom*, 2015.

[7] R. Nandakumar, B. Kellogg, and S. Gollakota, "Wi-fi gesture recognition on existing devices," *arXiv preprint arXiv:1411.5394*, 2014.

[8] W. Wang, A. X. Liu, and M. Shahzad, "Gait recognition using wifi signals," in *Proceedings of ACM UbiComp*, 2016.

[9] Z. Zhou, Z. Yang, C. Wu, and L. Shangguan, "Towards omnidirectional passive human detection," in *Proceedings of IEEE INFOCOM*, 2013.

[10] J. Hua, H. Sun, Z. Shen, Z. Qian, and S. Zhong, "Accurate and efficient wireless device fingerprinting using channel state information," in *Proceedings of IEEE INFOCOM*, 2018.

[11] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "Zigfi: Harnessing channel state information for cross-technology communication," in *Proceedings of IEEE INFOCOM*, 2018.

[12] C. Du, X. Yuan, W. Lou, and Y. T. Hou, "Context-free fine-grained motion sensing using wifi," in *Proceedings of IEEE SECON*, 2018.

[13] M. Kotaru, P. Zhang, and S. Katti, "Localizing low-power backscatter tags using commodity wifi," in *Proceedings of ACM CoNEXT*, 2017.

[14] A. Khamis, C. T. Chou, B. Kusy, and W. Hu, "Cardiofi: Enabling heart rate monitoring on unmodified cots wifi devices," in *Proceedings of ACM MobiQuitous*, 2018.

[15] C. Han, Q. Tan, L. Sun, H. Zhu, and J. Guo, "Csi frequency domain fingerprint-based passive indoor human detection," *Information*, vol. 9, no. 4, p. 95, 2018.

[16] Q. Tan, C. Han, L. Sun, J. Guo, and H. Zhu, "A csi frequency domain fingerprint-based method for passive indoor human detection," in *Proceedings of IEEE TrustCom/BigDataSE*, 2018.

[17] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas *et al.*, "Towards environment independent device free human activity recognition," in *Proceedings of ACM MobiCom*, 2018.

[18] J.-Y. Chang, K.-Y. Lee, K. C.-J. Lin, and W. Hsu, "Wifi action recognition via vision-based methods," in *Proceedings of ICASSP*, 2016.

[19] A. Pokkunuru, K. Jakkala, A. Bhuyan, P. Wang, and Z. Sun, "Neuralwave: Gait-based user identification through commodity wifi and deep learning," in *Proceedings of ACM IECON*, 2018.

[20] X. Tian, S. Zhu, S. Xiong, B. Jiang, Y. Yang, and X. Wang, "Performance analysis of wi-fi indoor localization with channel state information," *IEEE Transactions on Mobile Computing*, 2018.

[21] H. Li, X. Chen, H. Du, X. He, J. Qian, P.-J. Wan, and P. Yang, "Wi-motion: A robust human activity recognition using wifi signals," *arXiv preprint arXiv:1810.11705*, 2018.

[22] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *Acm Sigcomm Computer Communication Review*, vol. 41, no. 1, p. 53, Jan. 2011.

[23] Z. Yang, Z. Zhou, and Y. Liu, "From rssi to csi: Indoor localization via channel response," *ACM Computing Surveys (CSUR)*, vol. 46, no. 2, p. 25, 2013.

[24] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[25] L. Kong, D. Jiang, and M. Y. Wu, "Optimizing the spatio-temporal distribution of cyber-physical systems for environment abstraction," in *Proceedings of IEEE ICDCS*, 2010.

[26] H. Zhu, Y. Zhu, M. Li, and L. M. Ni, "Seer: Metropolitan-scale traffic perception based on lossy sensory data," in *Proceedings of IEEE INFOCOM*, 2009.

[27] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 449–462, 2010.

[28] J. K. Tan, "An adaptive orthogonal frequency division multiplexing baseband modem for wideband wireless channels," Ph.D. dissertation, Master's thesis, MIT, 2006.

[29] L. He, L. Fu, L. Zheng, Y. Gu, P. Cheng, J. Chen, and J. Pan, "Esync: An energy synchronized charging protocol for rechargeable wireless sensor networks," in *Proceedings of ACM MobiHoc*, 2014.

[30] C. Wu, Z. Yang, Z. Zhou, K. Qian, Y. Liu, and M. Liu, "Phaseu: Real-time los identification with wifi," in *Proceedings of IEEE INFOCOM*, 2015.

[31] Y. Zhuo, H. Zhu, and H. Xue, "Identifying a new non-linear csi phase measurement error with commodity wifi devices," in *Proceedings of IEEE ICPADS*, 2017.

[32] P.-H. Kuo, H. Kung, and P.-A. Ting, "Compressive sensing based channel feedback protocols for spatially-correlated massive antenna arrays," in *Proceedings of IEEE WCNC*, 2012.

[33] Z. Gao, L. Dai, and Z. Wang, "Structured compressive sensing based superimposed pilot design in downlink large-scale mimo systems," *Electronics Letters*, vol. 50, no. 12, pp. 896–898, 2014.

[34] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[35] Z. Yu, S. Hoyos, and B. M. Sadler, "Mixed-signal parallel compressed sensing and reception for cognitive radio," in *Proceedings of IEEE ICASSP*, 2008.

[36] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[37] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," in *Proceedings of ACM SIGMETRICS*, 2004.

[38] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, "Data loss and reconstruction in sensor networks," in *Proceedings of IEEE INFOCOM*, 2013.

[39] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *Society for Industrial and Applied Mathematics Review*, vol. 52, no. 3, pp. 471–501, 2010.

[40] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," in *Proceedings of ACM SIGCOMM*, 2009.