IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 26, 2021, accepted July 7, 2021, date of publication July 12, 2021, date of current version July 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3096527

Incident Retrieval and Recognition in Video Stream Using Wi-Fi Signal

YUSHENG HAO[1,2], WEILAN WANG[3], AND QIANG LIN[2]

[1]School of Mathematics and Computer Science, Northwest Minzu University, Lanzhou 730030, China
[2]Key Laboratory of Streaming Data Computing and Application, Northwest Minzu University, Lanzhou 730124, China
[3]Key Laboratory of China's Ethnic Languages and Information Technology of Ministry of Education, Northwest Minzu University, Lanzhou 730030, China

Corresponding author: Weilan Wang (787305757@qq.com)

This work was supported in part by the Innovation Funds for Higher Education of Gansu Province under Grant 2021B-067 and Grant 2020B-069, in part by the National Natural Science Foundation of China under Grant 61562075 and Grant 61866033, in part by the Gansu Provincial First-Class Discipline Program of Northwest Minzu University under Grant 11080305, in part by the Program for Innovative Research Team of SEAC under Grant [2018] 98, and in part by the Fundamental Research Funds for the Central Universities under Grant 31920170149 and Grant 31920200067.

ABSTRACT Retrieving incidents from video stream plays an important role in many computer vision applications. However, most video surveillance system can neither recognize incidents nor support content-based retrieval before the video stream is saved into files. As an emerging type of sensing modality, Wi-Fi signal have the potential to become a signal synchronized with the video stream to perform the incidents detection and recognition. In this work, we simultaneously collect the video stream and the Wi-Fi signal in two surveillance scenarios, and develop a LSTM-based classification model that is able to recognize the incidents in surveillance scenarios. Specifically, we first deploy a video surveillance system in two scenarios to capture the video stream and the synchronized Wi-Fi signal that is very sensitive to environmental changes. Second, an incident detection method based on the entropy change of Wi-Fi signal is proposed to find out the start and end time of the incident in the CSI sequence, thus greatly reducing the computational complexity compared with shot detection in the video stream. Third, the deep network LSTM is adopted to develop an incident recognition model that would be used to classify each size-variable CSI segments into known categories corresponding to the types of the incidents. Fourth, using Wi-Fi signal to locate and recognize incidents in the video stream, we build a quick content-based video retrieval system. Last, the experimental evaluation was performed on a group of real Wi-Fi signal samples. The statistical results shows that the proposed incident detection method is feasible and effective to find out the incidents in video files with an average error of 1.5 s. And the evaluation experiment results demonstrate that the proposed multi-classification model acquires an average value of 0.972, 0.973, 0.985, 0.972 and 0.962 for recall, precision, accuracy, F-1 score and Kappa coefficient, respectively.

INDEX TERMS CSI, video surveillance, incident retrieval, time series recognition, LSTM, Wi-Fi.

I. INTRODUCTION

With technological progress and social development, video surveillance system that would be used to capture some incidents have been deployed in many applications in our daily life. However, locating and recognizing incidents from dozens to thousands of hours of video stream has always been a challenging problem and also a research hotspot in the community [1].

Strictly speaking, video retrieval problem refers to the retrieval of videos that are similar to the video clips provided by users in a large video database [2]. To complete this type of retrieval task, it is first necessary to go through shot detection [3], key frame extraction [4], feature extraction [5], index classification and other steps to construct a video feature library. Then, in the retrieval stage, the features of the video clips provided by the user are extracted and compared with the features in the database to complete the retrieval task. It should be noticed that video content recognition can't be supported by video retrieval, that is, the video clips cannot be classified into specific categories in the vedio retrieval stage.

The associate editor coordinating the review of this manuscript and approving it for publication was Khin Wee Lai.

Video content recognition which has broad range of application is a far more complicated problem compared to static image classification. No matter the previous works that use hand-crafted representations [6]–[8] or based on deep learning [9]–[11], must not only needs to overcome variations such as object diversities, scale and noise, but also has to analyze incident cues in video stream, thus leading to more computational complexity and inefficiency.

Considering about the essence of the video stream carefully, it is just a medium that truly records the infromation about the past incidents to meets human visual requirements. Due to the complex spatial-temporal characteristics of video stream, performing retrieval and recognition tasks on which is not only computationally expensive, but also lacks real-time performance. Suppose there is a signal that is synchronized with the video stream, is easy to process, and can record the spatial-temporal pattern corresponding to the incidents happened in the surveillance scenario. We can perform incident retrieval and recognition based on this signal and the video signal is just used as a storage medium to provide visual evidence, then everything will become simple and elegant. Inspired by this point of view, we set our sights on the pervasive Wi-Fi signal in our daily life.

With the rapid development of wireless communication technology, Wi-Fi signal has been gradually extended from pure communication and networking to wireless perception and localization, opening up a new research band, namely: device-free passive sensing technology [12]. Wi-Fi signal is essentially electromagnetic wave radio frequency signal, the propagation path of which will change along with the electromagnetic changes caused by human activities or some incidents in the surveillance place during its propagation from the transmitter to the receiver. Concretely, the changes will be reflected in the channel state information (CSI) sequence in wireless physical layer (PHY), by analyzing the fluctuations of which, researchers have proposed fine-grained wireless sensing solutions, such as activity recognition [13]–[17], localization [18]–[22], identification [23]–[27] and so on.

However, all current existing research efforts aims to use Wi-Fi signal as a single sensing modality to realize intelligent sensing, ignoring the fusion and integration with other sensing medium, like video stream. Different from previous work, the main goal of this work is to achieve rapid retrieval and recognition of incidents in the video stream. The main data to be processed is not the video stream itself but the Wi-Fi signal instead, thus leading to better real-time performance and lower computational complexity. Specifically, by using off-the-shelf Wi-Fi devices and calling the open-sourced firmware [28] of the wireless network card, the CSI sequence of Wi-Fi signal reflecting some incidents or human activities is synchronously collected with the video stream. For the collected CSI sequence, an algorithm based on the entropy change of the sliding window is adopted to capture the start and end time of the incidents so as to obtain the segments of CSI sequence corresponding to the incidents. Based on a dataset that we have built for specific categories of the incidents, a deep recurrent neural network (RNN) is also trained to recognize the category of each CSI segment.

To sum up, the contributions of this work are as follows.

First, we pioneered the fusion of Wi-Fi signal and video stream, and converted the traditional video retrieval and recognition problem into the analysis and processing of CSI sequence of Wi-Fi signal, thus realizing video retrieval and recognition more efficiently. The conversion of the data object from 3-D video to 2-D CSI sequence greatly reduces the computational complexity and is also beneficial to protect user privacy.

Second, we develop an algorithm based on the entropy change of the sliding window to locate the start and end time of the incidents in the CSI sequence of Wi-Fi signal, thus greatly reducing the computational complexity compared with the shot detection in video data.

Third, we construct a RNN-based multi-classification model to classify each CSI segment into specific categories corresponding to specific types of the incidents. To the best of our knowledge, this is the first work to retrieve and recognize incidents in video stream by fusing Wi-Fi signal so as to simplify the pipeline of content-based video retrieval and recognition.

Last, we construct a dataset that contains 5 644 CSI sequence samples in four categories in the dormitory hall scenario, and 4 700 samples in three categories in the garage exit scenario.

The rest of this paper is organized as follows. The dataset construction, incidents detection method and LSTM-based multi-classification model will be illustrated in detail in Part II. The experimental evaluation metric and the experimental results will be shown in Part III. A concise discussion will be presented in Part IV. And in Part V, we conclude our work and look ahead to possible future research directions.

II. MATERIALS AND METHODS

A. PRINCIPLES OF WI-FI SENSING TECHNOLOGY

Wi-Fi signal is typically the electromagnetic signal, the channel characteristics of which are determined by the transmission medium, that is, the electromagnetic environment in propagation space. It is well known that there will be a multipath effect when the Wi-Fi signal propagate from the transmitters to the receivers due to the human activities or the occurrence of incidents in the surveillance scenario. Therefore, the received signal is a combination of a series of multipath signals. Specifically, there are two types of propagation paths of Wi-Fi signal, namely line of sight (LOS) and reflection path, as shown in Fig. 1.

Assuming that the transmitted signal is a sine wave, defined as:

$$x(t) = A\cos\omega_c t \tag{1}$$

where $x(t)$ denotes the transmitted signal, A denotes the signal amplitude and t denotes the time. When the Wi-Fi signal arrives at the receiver through various paths, the received

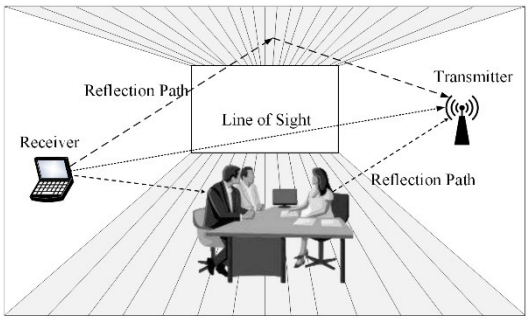

FIGURE 1. Wi-Fi signal propagation paths.

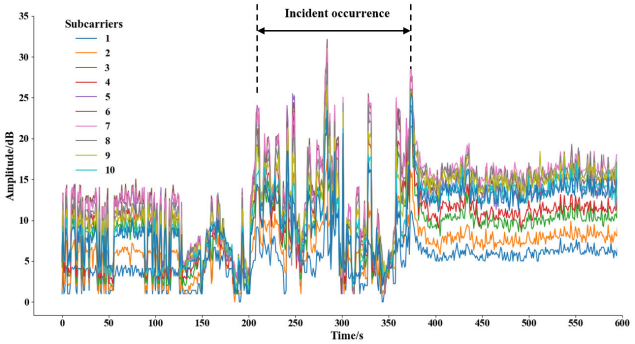

FIGURE 2. Signal pattern of the incident.

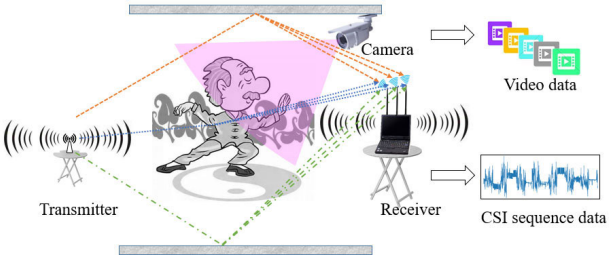

FIGURE 3. Signal collection system.

signal can be modeled as:

$$\begin{aligned} y(t) &= \sum_{i=1}^{n} A_i(t)\cos[\omega_c t + \varphi_i(t)] \\ &= \sum_{i=1}^{n} A_i(t)\cos\varphi_i(t)\cos\omega_c t - \sum_{i=1}^{n} A_i(t)\sin\varphi_i(t)\sin\omega_c t \\ &= X(t)\cos\omega_c t - Y(t)\sin\omega_c t \end{aligned} \tag{2}$$

where $y(t)$ denotes the received signal, $A_i(t)$ and $\varphi_i(t)$ respectively denotes the amplitude and phase of the signal transmitted along with the i-th path. Moreover, $X(t)$ and $Y(t)$ satisfy (3) and (4), respectively.

$$X(t) = \sum_{i=1}^{n} A_i(t)\cos\varphi_i(t) \tag{3}$$

$$Y(t) = \sum_{i=1}^{n} A_i(t)\sin\varphi_i(t) \tag{4}$$

where $X(t)$ and $Y(t)$ are mutually independent random variables. When n is large enough, both of them tend to be normally distributed. Therefore, $y(t)$, namely Rayleigh channel model, is defined as:

$$y(t) = V(t)\cos[\omega_c t + \varphi(t)] \tag{5}$$

where $V(t) = \sqrt{X^2(t) + Y^2(t)}$, denotes the envelope of the received signal and $\varphi(t) = \arctan(Y(t)/X(t))$, denotes the phase of the received signal.

When human activities or incidents happen in the surveillance scenario, both $V(t)$ and $\varphi(t)$ of the received signal will change significantly and show a specific spatial-temporal pattern corresponding to the categories of the incidents (see Fig. 2).

As a PHY layer metric of the wireless network based on Wi-Fi signal, CSI is able to reveal a group of channel measurements by sampling the timestamp, received signal strength indicator (RSSI), radio frequency chain, noise, matrix of channel frequency response and so on. Therefore, analyzing the spatial-temporal pattern of the above indicators in the CSI sequence and deriving the physical changes occurring in the surveillance scenario constitute the core of Wi-Fi sensing technology.

TABLE 1. The structured form of a CSI packet.

Field	Description
timestamp_low	Timestamp of the CSI packet
bfee_count	Statistics of the number of beams
Nrx	Number of receiving antennas
Ntx	Number of transmitting antennas
rssi_a	RSSI of antenna A
rssi_b	RSSI of antenna B
rssi_c	RSSI of antenna C
noise	Random channel noise
agc	Automatic gain control
perm	Arrangement of three receiving signals on three radio frequency chains
rate	Data transmission rate
csi	1×3×30 matrix in complex double

B. SIGNAL COLLECTION

As illustrated in Fig. 3, the CSI sequence of Wi-Fi signal is synchronously collected with the video stream. The main devices that make up the entire system are camera, wireless router and a laptop that receive Wi-Fi signal. We select a TP-LINK wireless router equipped with one single antenna as the transmitter, and use a Thinkpad laptop with a network interface card (NIC) equipped with 3 antennas as the signal receiver.

The sampling rate of the CSI packet in Wi-Fi signal is set to 200 Hz, so even the tiny channel response fluctuations in the surveillance scenario can be captured in a high precision. Massive CSI packets are stored in accessible data files (e.g.,*.dat* or *.txt* file) in a structured form as depicted in Table 1.

TABLE 2. The structure of the CSI packet.

ID	Timestamp	$CSI_{1,1}$		$CSI_{i,j}$	$CSI_{3,30}$	
		Real	Imag	...	Real	Imag
1	519750097	3	4		0	-9
2	520218097	-1	5		5	14
	520752344	-2	-3	...	14	-2
...	521220509	3	1		1	-7
n	521737636	2	2		-6	18

The channel frequency response between each pair of transmitting-receiving antennas in the form of complex, like [*Real*, *Imag*], is recorded in each CSI packet, and the value of which is recorded as a $N_t \times N_r \times 30$ complex matrix, where N_t denotes the number of transmitting antennas, N_r denotes the number of receiving antennas, and 30 represents the number of subcarriers in each radio beam according to IEEE 802.11 a/g/n standard [29]. We extract the most useful fields in each CSI packet, and organize the massive CSI packets into a time series according to the *TimeStamp* field, thus providing a CSI sequence for Wi-Fi sensing technology. As depicted in Table 2, the structure of $CSI_{i,j}$ packet with *Real* and *Imag* denoting the real and imaginary part of the channel frequency response between the i-th transmitting antenna and the j-th receiving antenna.

For an $n \times m$ multi-input and multi-output (MIMO) wireless communication system, assuming that the transmitted signal is $\mathbf{X} = [x_1, x_2, \ldots, x_n]$ and the received signal is $\mathbf{Y} = [y_1, y_2, \ldots, y_n]$, then the channel transformation from the transmitter to the receiver can be defined as:

$$\mathbf{Y} = \mathbf{H} \cdot \mathbf{X} + \mathbf{N} \tag{6}$$

where $\mathbf{N}$ is the pseudo-random noise that can be modeled by the circular symmetric complex normal distribution, that is $\mathbf{N} \sim cN(0, S)$, and $\mathbf{H}$ is the transformation matrix, defined as:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{1,1} & \mathbf{H}_{1,2} & \cdots & \mathbf{H}_{1,m} \\ \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \cdots & \mathbf{H}_{2,m} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{H}_{n,1} & \mathbf{H}_{n,2} & \cdots & \mathbf{H}_{n,m} \end{bmatrix} \tag{7}$$

where $\mathbf{H}_{i,j} (1 \le i \le n, 1 \le j \le m)$ denotes each channel transformation between the i-th transmitter and the j-th receiver.

The transformation matrix $\mathbf{H}$ contains the electromagnetic changes caused by physical environmental changes during the signal propagation from the transmitter to the receiver. Specifically, these changes containing several key indicators such as RSSI, amplitude, phase and propagation delay, reveal the signal reflection, scattering and power attenuation. The human activities or incidents in the surveillance scenario can be derived by analyzing these indicators. Mathematically, the transformation matrix $\mathbf{H}$ can be approximately estimated as:

$$\tilde{\mathbf{H}} = \frac{\mathbf{Y}}{\mathbf{X}} \tag{8}$$

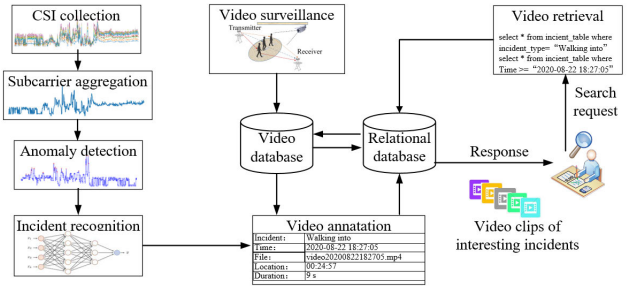

FIGURE 4. System architecture.

As a matter of fact, the CSI sequence as shown in Fig. 3 is just an estimation of matrix $\mathbf{H}$. The number of transmitting antenna N_t is set to 1, and the number of receiving antennas N_r is set to 3 in our proposed signal collection system. Therefore, at a given time t, the matrix $\mathbf{H}$ can be specifically defined as:

$$\begin{aligned} \mathbf{H}(t) &= \begin{bmatrix} \mathbf{H}_{1,1}(t) \\ \mathbf{H}_{1,2}(t) \\ \mathbf{H}_{1,3}(t) \end{bmatrix} \\ &= \begin{bmatrix} h^1_{1,1}(t) & h^2_{1,1}(t) & \cdots & h^{30}_{1,1}(t) \\ h^1_{1,2}(t) & h^2_{1,2}(t) & \cdots & h^{30}_{1,2}(t) \\ \vdots & \vdots & \vdots & \vdots \\ h^1_{1,3}(t) & h^2_{1,3}(t) & \cdots & h^{30}_{1,3}(t) \end{bmatrix} \end{aligned} \tag{9}$$

where $\mathbf{H}(t)$ denotes the CSI matrix contained in the received signal at time t, $h^k_{i,j}(t)$ $(i = 1, 1 \le j \le 3, 1 \le k \le 30)$ is a complex number, denoting the channel frequency response of the k-th subcarrier in the radio beam between the i-th transmitting antenna and the j-th receiving antenna at time t.

C. SYSTEM ARCHITECTURE

To the best of our knowledge, almost all existing video surveillance systems are deployed with camera, router, video display system and specialized network. As more and more wireless routers are deployed in video surveillance systems, making it possible to achieve video retrieval and recognition assisted by Wi-Fi signal. The whole system consists of several major modules such as video recording, CSI data collection, data preprocessing, incident detection, incident recognition, video stream annotation, and video retrieval modules (see Fig. 4).

- The video recording module is the core of the video surveillance system. It mainly collects video stream through a camera, and forms a video file in the computer disk after transmission via a specialized network.
- The collection module of CSI sequence uses the existing wireless router in the video surveillance system as the signal transmitter, and uses the computer wireless network interface card (NIC) as the signal receiver to build a Wi-Fi communication system. By calling the driver of NIC, the system collects the CSI sequence

of Wi-Fi signal and stores it in the form of accessible files to form static data. From a real-time perspective, the CSI sequence can also be dynamically processed in the memory.

- The data preprocessing module mainly performs carrier aggregation and other preprocessing operations on the CSI sequence to provide data support for incident detection and recognition.
- The video stream annotation module consists of incident detection and incident recognition submodules. The main purpose of the incident detection module is to locate the start time and end time of the incident in the video stream. An anomaly detection algorithm is implemented based on the entropy change of the sliding window on the CSI sequence. It finally find out the start time of the incident in the video stream and tracks its duration. Simultaneously, the incident recognition module classifies the CSI segments output by the incident detection module into specific categories of the incidents via the trained deep learning classification model.
- The content-based video retrieval module is completed by inserting the results of the incident detection and recognition into the relational database. Specifically, for each incident in the CSI sequence, its start time, duration, category and corresponding file names of video data and Wi-Fi signal data are all organized as a fixed record and inserted it into a table of the relational database, thus providing the possibility to achieve content-based video retrieval via SQL statements. In the retrieval stage, the user can send a query request by constructing a SQL statement like '' SELECT $*$ FROM incident_table WHERE incident_type = 'walking into' AND start_time $\geq$ '2020-07-28 09:18:00' '', where 'incident_table' is the table name of the database, and 'incident_type' and 'start_time' are two fields in the table. The database engine will cut out specific video clips in the corresponding video file according to the query result of the relational database and present it to the user.

D. INCIDENT DETECTION

As depicted in Fig. 2, when no incident occurs in the surveillance scenario, the electromagnetic environment of the Wi-Fi signal remains stable and the channel state is not affected by drastic changes in the physical environment, so the CSI sequence shows a relatively ''stable'' state. The occurrence of human activities or incidents will cause drastic fluctuations in the CSI sequence, which is regarded as an abnormality of the Wi-Fi signal. When the incident go past, the CSI sequence returns to be ''stable'' again. Intuitively, the abnormality in CSI sequence denotes the human activities or the occurrence of incidents in the surveillance scenario.

In our proposed system, there are three pairs of transmitting-receiving antennas, each pair has 30 subcarriers it its radio beam, so there are a total of 90 subcarriers. In order to facilitate the anomaly detection in the CSI sequence, we first aggregate 90 subcarriers into one signal according to equation (10).

$$\mathbf{H}(t) = \frac{1}{90}\sum_{k=1}^{90}\frac{f_k}{f_{i,j}} \times \left|h_{i,j}^{k}(t)\right| \tag{10}$$

where $\mathbf{H}(t)$ is the aggregated signal, $h_{i,j}^{k}(t)$ is the k-th ($1\leq\ k\ \leq 30$) subcarrier between the i-th ($i = 1$) transmitting antenna and the j-th ($1 \leq j \leq 3$) receiving antenna, f_k is the frequency of the k-th subcarrier, and $f_{i,j}$ is the center frequency of the radio beam.

Due to the high synchronization of CSI sequence and video stream in time, we developed an anomaly detection algorithm based on Isolation Forest [30] to detect outliers in the CSI sequence to locate the start time of the incident. Isolation Forest is an unsupervised anomaly detection algorithm, and the principle of which is that the anomalous points are those points that are sparsely distributed and far away from a high-density group. In the feature space, the probability of an incident occurring in a sparse area is much lower, thus the anomalous points can be segmented in fewer calculation steps during segmentation. The anomaly detection algorithm is based on two definitions, namely *isolated tree* and *path length*, which are defined as follows.

Isolated Tree: Let T be a node of the isolated tree. It is either a lead node or an internal node with two child nodes (T_l, T_r). In each step of segmentation, for the feature q and the threshold p, the points that satisfies $q < p$ are segmented to the left child node T_l, and the points that satisfies $q \geq p$ are segmented to the right child node T_r.

For a given set $X = \{x_1, x_2, \ldots, x_3\}$, assume that the feature dimension is d. When constructing an isolated tree, you must select feature q and the segmentation threshold p, then recursively segment X until any of the following conditions are met.

- The height of the tree reaches the limit value.
- There is only one sample on the node.
- All features of the sample on the node are the same.

Path Length h(x): The number of edges traversed from the root node of the isolated tree to the leaf nodes.

The shorter the path length, the higher the probability that the sample point is an abnormal point. We define an anomaly score index to evaluate the probability of whether a sample point is an anomaly one. Given a data set X, which contains n sample points, the average path length of the tree is defined as:

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{11}$$

where $c(n)$ is the average path length of the isolated tree when the size of the sample set is n, which is used to standardize the path length $h(x_i)$ of each sample x_i. And $H(i)$ denotes the harmonic number that is estimated using $ln(i)$+0.5772156649. Then the anomaly score of the sample x_i is defined as:

$$s(x_i, n) = 2^{-\frac{E(h(x_i))}{c(n)}} \tag{12}$$

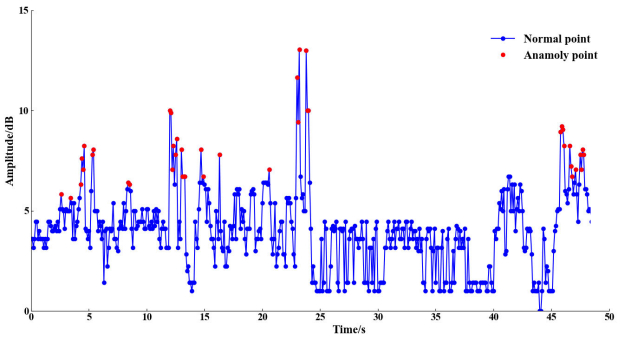

FIGURE 5. Anomaly detection in CSI sequence.

where $E(h(x_i))$ is the expectation of the path length of the sample x_i in a batch of isolated trees.

When $E(h(x_i)) \to c(n)$, $s(x_i, n) \to 0.5$, that is, the average length of the sample x_i is close to the average length of the tree, and it is difficult to distinguish whether x_i is an abnormal point.

When $E(h(x_i)) \to 0$, $s(x_i, n) \to 1$, that is, the average length of the sample x_i is close to 1, and x_i is judged to be an abnormal point.

When $E(h(x_i)) \to n-1$, $s(x_i, n) \to 0$, x_i is judged as a normal point.

Fig. 5 shows the anomaly detection results (red points) of the algorithm on an instance of CSI sequence.

The first red point in Fig. 6 denotes the beginning of the incident. To extract a specific segment of the incident saved in the video file, in addition to record the start time of the incident, it is also necessary to track the duration of it to ensure the time integrity.

E. CALCULATE THE DURATION OF THE INCIDENT

We measure the fluctuations of the CSI sequence by tracking the approximate entropy [31] of a sliding window, thus calculating out the duration of the fluctuation caused by the incident. The rationale behind this is that the occurrence of the incident makes the CSI sequence continue to fluctuate until the incident go past. During the duration of the incident, the approximate entropy of the CSI sequence is much greater than that of the usual time when no incident occurs.

The approximate entropy is a non-negative number that can measure the complexity of a time series. The more complex the time series, the greater the approximate entropy. In other words, the more obvious the fluctuation of the CSI sequence, the greater the approximate entropy of it. For a given time series $\{u(i)\}$, its approximate entropy can be calculated according to the Algorithm 1.

Usually, m is set to 2, and r is in the interval $[0.1 \times std, 25 \times std]$, where std is the standard deviation of $\{u(i)\}$.

Furthermore, the impact of the occurrence of the incident on the CSI sequence is local rather than global. Therefore, we develop an approximate entropy tracking method based on sliding window, which can judge the local fluctuations of CSI sequence by calculating and comparing the local approximate entropy. From the beginning of the incident, the approximate entropy of a sliding window with fixed size is tracked, when it is less than a preset threshold ε, the incident is considered to be over and the corresponding time of the sliding window is regarded as the end time of the incident. The specific process is illustrated in Algorithm 2.

Algorithm 1 ApEn($\{u(i)\}, m, r$)

Input:

$\{u(i)\}$ — Time series.

m — Length of sub-sequence.

r — Preset threshold.

Output:

$ApEn(\{u(i)\}, m, r)$ — Approximate entropy of time series $\{u(i)\}$.

Process:

(1) Reconstruct the vector $X(i)$ of length m.

$$X(i) = [u(i), u(i+1), \ldots, u(i+m-1)],$$
$$i = 1 \sim N - m + 1$$

(2) Calculate the distance between the vector $X(i)$ and the rest of the vector $X(j)$.

for each $X(i)$

$$d[X(i), X(j)] = \max_{k=0\sim m-1} |u(i+k) - u(j+k)|$$

end for

(3) Calculate the ratio of the number of vectors with $d[X(i), X(j)] < r$ to the total number of vectors $N-m+1$.

for each $X(i)$

$$C_i^m(r) = \frac{size(d[X(i),X(j)]<r)}{N-m+1}$$

end for

(4) Take the logarithm of $C_i^m(r)$ and find the average value.

$$\Phi^m(r) = \frac{\sum_{i=1}^{N-m+1} \ln C_i^m(r)}{N-m+1}$$

(5) Add 1 to the value of m and repeat the process of (1) $\sim$ (4) to get $\Phi^{m+1}(r)$.

(6) Calculate the approximate entropy of the time series $\{u(i)\}$.

$$ApEn(u(i), m, r) = \Phi^m(r) - \Phi^{m+1}(r)$$

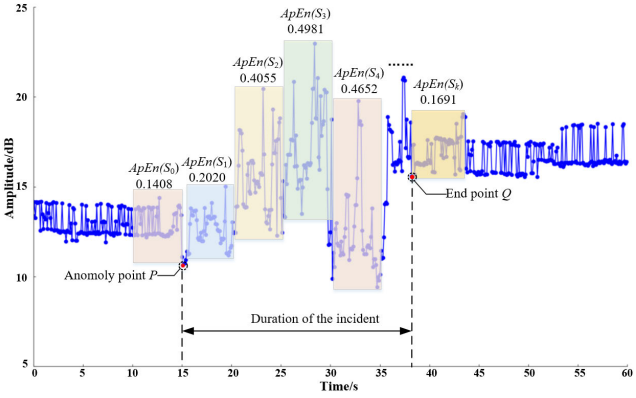

FIGURE 6. The process of tracking approximate entropy via sliding windows.

Algorithm 2 FindTail({u(i)}, pos, s, ε)

Input:

$\{u(i)\}$ — Time series.

pos — Beginning time of incident.

$size$ — Size of the sliding window.

ε — Preset threshold of approximate entropy.

Output:

FindTail($\{u(i)\}$, *pos, size*, ε) — The end time of the incident.

Process:

(1) Construct a CSI subsequence with size of s before the abnormal point $u(pos)$ to form the first sliding window S_0.

$$S_0 = [u(pos - m - 1), \ldots\ldots u(pos - 2), u(pos - 1), u(pos)]$$

(2) Calculate the approximate entropy of S_0 according to Algorithm 1.

$$E_0 = ApEn(S_0, m, r)$$

(3) Starting from the abnormal point $u(pos)$, a sliding window of size s is constructed, and the sliding step is set to 1. For each sliding window, calculate its approximate entropy E_k according to Algorithm 1. Then calculate the absolute error between E_k and E_0, when $|E_0\text{-}E_k| < \varepsilon$, the window stops sliding.

$\boldsymbol{k = 1}$

do{

$S_k = [u(pos + k), u(pos + k + 1), \ldots, u(pos + k + m - 1)]$

$E_k = ApEn(S_k)$

$k = k + 1$

}while($|E_0\text{-}E_k| < \boldsymbol{\varepsilon}$)

(4) Return $pos + k$ as the end time of the incident.

Fig. 6 shows the process of tracking the approximate entropy of sliding windows via Algorithm 1 and Algorithm 2. After detecting the anomaly point P, first use the CSI sequence before the point P to construct the sliding window S_0 of size 5 seconds and calculate the approximate entropy of S_0 as $E_0 = 0.1408$. After that, the sliding window begin to slide, thus producing a series of sliding windows $S_1, S_2, \ldots, S_k$ with the approximate entropy of $E_1 = 0.2020$, $E_2 = 0.4055, \ldots, E_k = 0.1691$, respectively. During the whole sliding process, the absolute error between E_0 and E_k ($k \geq 1$) is always tracked. When it is less than the preset threshold $\varepsilon = 0.05$, the window stops sliding. The starting point Q of the last window S_k is considered to be the end point of the incident and the distance between P and Q is the duration of the incident.

So far, we have used Wi-Fi signal as the key signal to synchronize with the video stream, and obtained the start time and duration of the incident on the time axis. We store these key information in the designed relational database table (see Table 3), which lays the foundation for content-based video retrieval via SQL statements.

TABLE 3. Registration form of the incidents.

ID	FileName	Pos	Len	Type
1	031511.avi	00:26:35	7	single passing
2	031512.avi	00:39:17	6	many passing
3	031513.avi	00:47:24	25	tarrying
4	031608.avi	00:12:17	9	car passing
...	.....	.....	...	.....
n	031609.avi	00:19:13	35	jam

Although Table 3 has a quite simple structure, the value of which is quite important for building a content-based video retrieval system. Each record in the table corresponds to an incident. The fields of *ID, FileName, Pos, Len* and *Type* of the table respectively denotes the index, the video file name, the start time, duration and category of the incident. It is worth noting that the value of *Type* field relies on the multi-classification model based on deep learning, and the specific implementation of the model will be presented in Section F, Part II.

F. INCIDENT RECOGNITION

All supervised deep learning methods heavily rely on the dataset, and the network models can learn a large number of parameters from the massive labeled samples. In this section, we first construct a dataset of CSI segments corresponding to the specific categories of the incidents. Then, using the constructed dataset, a multi-classification model based on long-short term memory (LSTM) network is trained.

1) WI-FI SIGNAL DATASET

In a video surveillance system, there is no doubt that the incidents that have occurred are stored in files in the form of video clips, which can directly provide human visual evidence. However, it is also accompanied by some troubles, such as complex calculations, poor real-time performance, and disclosure of user privacy. In our system, the occurrence of the incident is recorded as a CSI segment, in which the signal spatial-temporal pattern when the incident occurs is stored. This novel way of saving incident history with CSI sequence of Wi-Fi signal reduces the computational complexity of information retrieval and protects user privacy when necessary.

For each CSI segment corresponding to a specific category of the incidents, 10 hand-crafted features (see Table 4) are extracted from which to form the training samples that would be fed into the multi-classification model.

Specifically, by manually labelling each CSI segment of Wi-Fi signal corresponding to a specific incident in the video stream, we construct a dataset containing 10 344 CSI segment samples in two surveillance scenarios of dormitory hall and garage exit. More specifically, there are four types of incidents in the dormitory hall scenario, including 1 000 samples of "no one" type, 1 024 samples of "single passing" type, 1 060 samples of "many passing" type and 960 samples of "tarrying" type. Similarly, there are three types of

TABLE 4. Features of the CSI segment.

Symbols	Definitions	Feature description
MV	$\frac{1}{n}\sum_{i=1}^{n}u_i$	Mean
RANGE	$u_{\max}-u_{\min}$	Range
MAE	$\frac{\sum_{i=1}^{n}n\lvert u_i-\bar{u}\rvert}{n}$	Mean Absolute Error
VAR	$\frac{1}{n}\sum_{i=1}^{n}(u_i-\bar{u})^2$	Variance
TOCD	$\frac{1}{n}\sum_{i=1}^{n}(u_i-\bar{u})^3$	Third order center distance
LEPT	$\frac{\frac{1}{n}\sum_{i=1}^{n}(u_i-\bar{u})^4}{(\frac{1}{n}(u_i-\bar{u})^2)^2}-3$	Leptokurtosis
IR	Q_3-Q_1	Interquartile range
SUM	$\sum_{i=1}^{n}u_i$	Sum
RMS	$\sqrt{\frac{\sum_{i=1}^{n}u_i^2}{n}}$	Root Mean Square
SKE	$\frac{\frac{1}{n}\sum_{i=1}^{n}(u_i-\bar{u})^3}{(\frac{1}{n}(u_i-\bar{u})^2)^{\frac{3}{2}}}$	Skewness

incidents in the garage exit scenario, including 1 200 samples of ''no car'' type, 1 100 samples of ''car passing'' type, and 1 050 samples of ''jam'' type.

2) MULTI-CLASSIFICATION MODEL

The problem of incident recognition is essentially a problem of pattern classification. Based on the establishment of the dataset, we use a data-driven supervised deep learning model to solve the problem. Specifically, we convert the video recognition problem into the classification of CSI segments, thus greatly reducing the computational complexity and improving the real-time performance of the system. Due to the advantages of recurrent neural network (RNN) [32] in processing time series, we adopt long short-term memory (LSTM) [33] network based on RNN to classify CSI segments.

The LSTM network is a special recurrent neural network that is capable of learning long-term dependencies. The basic unit of LSTM is the memory cell that is able to remove and add information to the cell state. The structure of the memory cell is depicted in Fig. 7.

For a LSTM network, each layer of which is composed of memory cells that has the ability to maintain, remove or add information to the cell state through three types of gates. The gates are a way to optionally let information through. They are composed of a sigmoid or tanh neural net layer and pointwise operation.

- *Forget gate:* The first step in the cell is to decide what information we are going to remove from the cell state. A sigmoid layer namely ''forget gate'' is used to make the decision. For two inputs x_t and h_{t-1}, the sigmoid function with the expression $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$ output a number between 0 and 1 for each number in the cell state C_{t-1}. A 1 means ''keep this information'' and a 0 represents ''throw it away''.
- *Input gate:* The second step is to decide what information we are going to add in the cell state. First, a sigmoid layer with the expression $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$ is utilized to decide which information will be updated. Second, a tanh layer with the expression $\hat{C}_t = \sigma(W_c \cdot [h_{t-1}, x_t] + b_c)$ is used to create a new vector of candidate values that could be added to the cell state. Third, the old cell state, C_{t-1}, is going to be updated with the expression $C_t = f_t * C_{t-1} + i_t * \hat{C}_t$, in which the things we are going to forget is expressed by $f_t * C_{t-1}$ and the information to be added is expressed by $i_t * \hat{C}_t$.
- *Output gate:* The final step is to decide the output of current cell. First, a sigmoid layer, expressed by $O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$, is run to decide which part of the cell state we are going to output. Then, the cell state C_t is feed into the tanh layer (to put the value to be between -1 and 1) and multiply it by the output of the sigmoid layer, so the output of output gate is expressed by $h_t = O_t * \tanh(C_t)$.

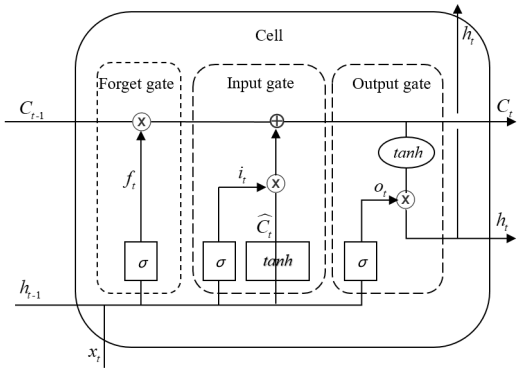

FIGURE 7. The structure of the memory cell in LSTM. h_{t-1} is the output of the previous cell, C_{t-1} is the cell state at time t-1, h_t is the output of current cell, C_t is the cell state at time t, and the symbols $\oplus$ and $\otimes$ denotes the pointwise addition and pointwise multiplication.

In this work, we develop a LSTM-based multi-classification model consisting of one input layer, five hidden LSTM layers, a fully connection layer, a softmax layer and a classification output layer (see Fig. 8).

The details of the multi-classification model based on LSTM are described as follows.

- *Input Layer:* A $10 \times t$ feature matrix extracted from each CSI segment will be fed into the input layer, where t denotes the duration of the CSI segment corresponding to the incident.
- *LSTM Hidden Layers:* Three are five LSTM hidden layers in our multi-classification model and the number of memory cells within each hidden layer is 150, 125, 100, 75 and 50, respectively.

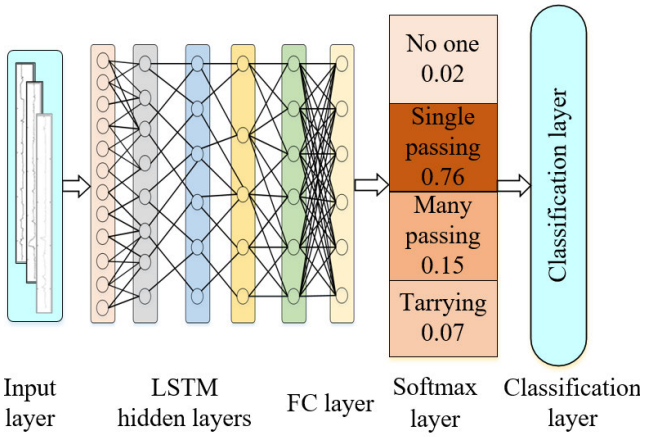

FIGURE 8. The structure of the multi-classification model based on LSTM.

- *Fully Connected Layer:* The fully connected layer plays the role of mapping the learned "distributed feature representation" to the labelled space of the samples, that is, it can convert the 2-D features of the LSTM hidden layers to 1-D output corresponding to the specific category of the CSI segments.
- *Softmax Layer:* The Softmax function, also known as the normalized exponential function. It is the promotion of sigmoid in multi-classification model, the purpose is to show the results of multi-classification in the form of probability.
- *Classification Layer:* This layer is to transform the output probability of the softmax to the categories of the CSI segments, i.e., no one, single passing, many passing or tarrying.

III. EXPERIMENTAL EVALUATION

In this part, the developed LSTM-based multi-classification model will be evaluated by using a group of labelled samples of CSI segments in two different surveillance scenarios.

A. EXPERIMENTAL SCENARIOS

In order to verify that CSI sequence of Wi-Fi signal have different responses to the incidents of different granularities and can record their spatial-temporal patterns accurately, we deploy our experiments in two completely different surveillance scenarios, each of which have different action granularities.

As depicted in Fig. 9(a), the signal collection system was deployed to simultaneously collect the video stream and the CSI sequence of Wi-Fi signal for capturing real behaviors of human in dormitory hall scenario. A TP-LINK wireless router equipped with one single antenna is used as the transmitter, and a Thinkpad laptop with the network interface card (NIC) equipped with 3 antennas as the receiver to collect CSI sequence of Wi-Fi signal. The height of the transmitter and receiver was set to 1.5 m, and the distance between them is set to 3.5 m. Meanwhile, an off-the-shelf camera was used to record the video information in the scenario. The camera

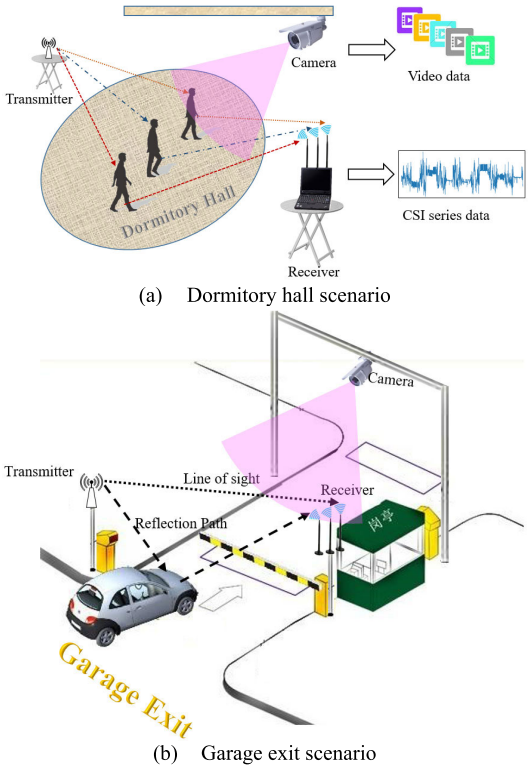

FIGURE 9. Two experimental scenarios.

is fixed on the ceiling of the hall to ensure that the whole scenario can be covered by its field of vision. In this scenario, we mainly focus on four types of incidents composed of human actions, namely, no one in the hall, a single person passing, multiple people passing at the same time and someone tarrying at the hall (tarrying).

Compared with the fine-grained human actions in the dormitory hall scenario, we also set our sights on three types of larger-grained incidents in the garage exit scenario, namely, no vehicles passing, vehicles passing and traffic jams. The deployment of the experiment is shown in Fig. 9(b).

B. EVALUATION METRIC

Succinctly speaking, this work is to use Wi-Fi signal as the medium to solve the problem of detecting and recognizing incidents in video surveillance scenarios. It is necessary for us to measure the time accuracy of the incident detection algorithm and the classification performance of the multi-classification model.

For the incident detection algorithm in the video stream, the performance of which is mainly reflected in the time. An algorithm that can accurately locate the start and end time of an incident is considered to be good. Therefore, we first propose a comprehensive error index that measures the time accuracy of the incident detection algorithm, namely *CE*, defined as:

$$CE = \frac{\sum_{i=1}^{N} \alpha \left| T_{pb\_i} - T_{gb\_i} \right| + \sum_{i=1}^{N} \beta \left| T_{pe\_i} - T_{ge\_i} \right|}{N} \quad (13)$$

where N is number of samples in the test set, T_{pb_i}, T_{pe_i} denotes the predicted start time and the predicted end time of the i-th incident in the test set, and T_{gb_i}, T_{ge_i} denotes the ground truth start time and the ground truth end time of the i-th incident, respectively. α and β are the scale factors, which are set to $\alpha = 0.8$, $\beta = 0.2$ in this work. This is because we believe that the start time of the incident in the video stream has a greater weight, and the greater the error in its estimation, the lower the performance of the incident detection algorithm.

Another metric we made is for the classification performance of the multi-classification model based on LSTM. Each CSI segment fed into the classification model will falls into one of the four categories.

- *True Positive (TP):* A positive sample is correctly recognized as a positive sample.
- *False Positive (FP):* A negative sample is incorrectly recognized as a positive sample.
- *False Negative (FN):* A positive sample is incorrectly recognized as a negative sample.
- *True Negative (TN):* A negative sample is correctly recognized as a negative sample.

Based on the above indicators, we define recall (*Rec*), precision (*Prec*), accuracy (*Acc*) and F-1 score as:

$$Rec = \frac{TP}{TP + FN} \tag{14}$$

$$Prec = \frac{TP}{TP + FP} \tag{15}$$

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \tag{16}$$

$$F - 1 = \frac{2 \times Prec \times Rec}{Prec + Rec} \tag{17}$$

It is necessary to point out that the above indicators are only suitable for evaluating two-classification model, but not suitable for evaluating multi-classification model. In this work, we make some modifications to the definitions of the above indicators to enable them to evaluate multi-classification model.

As depicted in Fig. 10, we give the definition of *TP, TN, FP* and *FN* on the confusion matrix for the two-classification, three-classification and four-classification situation. The confusion matrix is used to demonstrate the classification result by the model, each row of which represents the total number of samples of a specific category, and each column of which represents the number of samples classified into each category by the model.

We also evaluate the effectiveness of the multi-classification model by the Kappa coefficient (K), defined as:

$$K = \frac{Acc - Pe}{1 - Pe} \tag{18}$$

where the probability P_e is defined as:

$$Pe = \frac{\sum_{i=1}^{C} C_i \times M_i}{N^2} \tag{19}$$

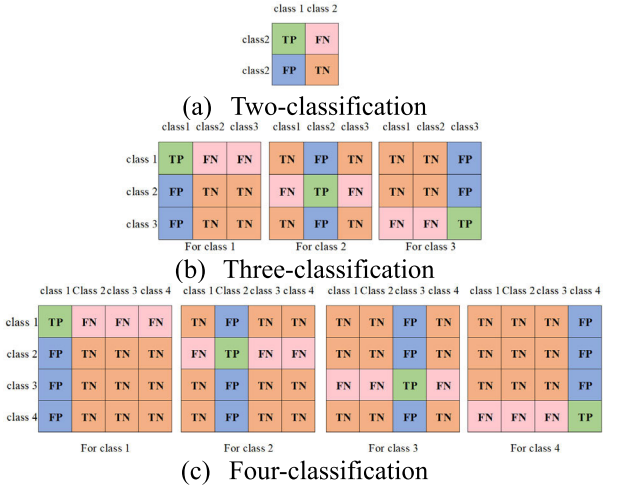

FIGURE 10. Definitions of evaluation metric for multi-classification model. For a sample in a certain category, if the model classifies it into that category, it will be regarded as a positive sample; for samples that are not classified into this category, it will be regarded as a negative sample. Therefore, the definitions of the indicators *TP, TN, FP* and *FN* vary with different types of the samples in multi-classification model.

where C is the number of categories of the dataset, C_i is the number of samples of the i-th category, M_i is the number of correctly classified samples of the i-th category and N is the total number of samples in the dataset.

Kappa coefficient can evaluate the effectiveness of the model in classification. When $0 \leq K \leq 0.2$, the model effectiveness is "extremely low"; When $0.2 < K \leq 0.4$, the model effectiveness is "normal"; When $0.4 < K \leq 0.6$, the model effectiveness is "medium"; When $0.6 < K \leq 0.8$, the model effectiveness is "high"; When $0.8 < K \leq 1.0$, the model effectiveness is "extremely high".

C. EXPERIMENTAL RESULTS

As mentioned in Section F, Part II, we construct a dataset of CSI segments in two completely different surveillance scenarios. The timestamp and specific category of the samples are labelled by manually efforts thus providing the ground truth to evaluate the incident detection algorithm and the multi-classification model.

1) INCIDENT DETECTION RESULTS

We first measure the time accuracy of the incident detection algorithm using equation (13) defined in section B, Part III. For each incident in the dataset, the absolute error between the predicted start time in CSI sequence and the ground truth start time manually labelled in the video stream is calculated, and the absolute error between the predicted end time in CSI sequence and the ground truth end time labelled in the video stream is also be calculated. The statistical results demonstrate that the *CE* of the incident detection algorithm on the entire dataset is 1.5 s. Table 5 shows 20 incident detection records randomly extracted from the dataset.

TABLE 5. Incident detection records.

i	T_{gb}	T_{pb}	$error1$	T_{ge}	T_{pe}	$error2$
1	21:07:49	21:07:48	1	21:08:07	21:08:09	2
2	10:02:44	10:02:42	2	10:02:49	10:02:48	1
3	15:04:53	15:04:51	2	15:05:02	15:05:04	2
4	19:19:16	19:19:16	0	19:19:34	19:19:33	1
5	14:32:08	14:32:07	1	14:32:16	14:32:18	2
6	11:00:42	11:00:44	2	11:00:49	11:00:47	2
7	11:43:49	11:43:51	2	11:44:02	11:44:05	3
8	21:45:10	21:45:09	1	21:45:27	21:45:32	5
9	10:35:08	10:35:10	2	10:35:15	10:35:11	4
10	11:38:49	11:38:47	2	11:39:05	11:39:03	2
11	18:19:11	18:19:11	0	18:19:18	18:19:18	0
12	11:09:59	11:09:59	0	11:10:24	11:10:24	0
13	14:28:24	14:28:26	2	14:28:34	14:28:33	1
14	13:05:57	13:05:55	2	13:06:01	13:05:56	5
15	14:58:04	14:58:02	2	14:58:24	14:58:24	0
16	16:40:42	16:40:43	1	16:40:54	16:40:54	0
17	20:38:02	20:38:01	1	20:38:10	20:38:14	4
18	12:38:24	12:38:22	2	12:38:31	12:38:32	1
19	16:38:13	16:38:15	2	16:38:23	16:38:28	5
20	15:35:08	15:35:08	0	15:35:27	15:35:29	2

TABLE 6. Distribution of samples in the dataset.

Scenario	Sample classes	Train set	Test set
Dormitory hall	No one	1 000	400
	Single passing	1 024	400
	Many passing	1 060	400
	Tarrying	960	400
	Subtotal	**4 044**	**1 600**
Garage exit	No car	1 200	450
	Car passing	1 100	450
	Jam	1 050	450
	Subtotal	**3 350**	**1 350**

TABLE 7. Main parameters setting of the model.

Parameters	Value
Gradient decay factor	0.900 0
Squared gradient decay factor	0.999 0
Epsilon	0.000 000 01
Initial learning rate	0.001
L2 regulation	0.000 1
Gradient threshold method	L2 normalization
Gradient threshold	1
Maximum epochs	80
Minimum batch size	13
Shuffle	once
Execution environment	single GPU

2) INCIDENT RECOGNITION RESULTS

In the experiments of this work, we utilize 70% of the dataset for training the model, and the rest 30% for testing. Table 6 shows the specific distribution of the samples in two different scenarios.

As we can see in Table 6, there are seven types of incidents in two scenarios. However, in order to verify that the CSI sequence has different responses to the incidents with different granularities, we input all samples of the train set into one network for training. Table 7 shows the main parameters setting of the model in this work.

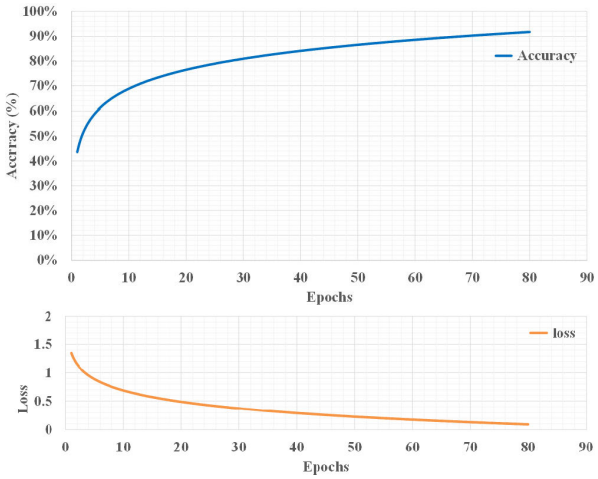

FIGURE 11. Training process of the multi-classification model based on LSTM.

TABLE 8. Evaluation metrics for dormitory hall scenario.

Incident	*Rec*	*Prec*	*Acc*	*F-1*	*K*
No one	0.998	1.000	0.999	0.999	
Single passing	0.958	0.903	0.964	0.927	—
Many passing	0.900	0.955	0.964	0.927	
Tarrying	1.000	1.000	1.000	1.000	
Average	**0.964**	**0.965**	**0.982**	**0.964**	**0.953**

TABLE 9. Evaluation metrics for garage exit scenario.

Incident	*Rec*	*Prec*	*Acc*	*F-1*	*K*
No car	0.973	1.000	0.991	0.987	
Car passing	0.967	0.973	0.980	0.970	—
Jam	1.000	0.968	0.990	0.984	
Average	**0.980**	**0.980**	**0.987**	**0.980**	**0.971**

According to the network model shown in Fig. 8 and the main parameters settings in Table 7, a LSTM-based multi-classification model is trained and tested on a workstation, the basic configuration of which are as follows.

- CPU: Intel CORE^{TM} i7 8700
- Memory size: 16 GB
- GPU: NVIDIA GTX 1080 Ti
- Platform: Windows 10 + Matlab R2019a

As shown in Fig. 11, as the number of training iterations continues to increase, the accuracy of the multi-classification model is increasing, and the loss of the network is decreasing. It can be seen that the entire training process is basically stable, and there is no obvious fluctuation in the accuracy curve and the loss curve.

Finally, the LSTM-based multi-classification model obtains a mini-batch accuracy of 92% after 80 epochs of training for a total of 45 440 iterations, and the loss is less than 0.18.

Table 8 and Table 9 show the quantitative experimental results of the indicators defined in Section B, Part III in the two surveillance scenarios.

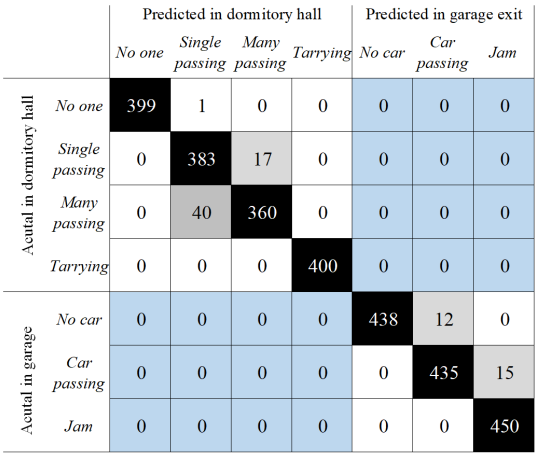

FIGURE 12. The confusion matrix obtained by the LSTM-based multi-classification model on the test set.

Analyzing the experimental results, we can at least make the following well-reasoned inferences.

- The LSTM-based multi-classification model can classify the CSI segments into different categories with a very high recall and precision, thus obtaining a high accuracy in two surveillance scenarios.
- As listed in Table 5, although the number of samples of the garage exit scenario is smaller than that of the dormitory hall scenario, the trained model is more accurate in classifying the samples in garage exit scenario. This shows that the LSTM-based multi-classification model has better classification performance for larger-grained incidents.
- The Kappa coefficient (K) of the model is extremely high, in other words, the model has a very good effectiveness in classifying the samples of the dataset.

As depicted in Fig. 12, by giving the confusion matrix, we further examine that the CSI sequence have different responses to the different-grained incidents.

We can see in the above confusion matrix, the classification performance of ''Tarrying'' class is the best in the dormitory scenario, and there is no confusion with other classes. Among the 400 samples of class ''Many passing'', 360 are correctly classified, but 40 are incorrectly classified into class ''Single passing''. Besides, out of 400 samples of class ''Single passing'', 383 are correctly classified and 17 are incorrectly classified into class ''Many passing''. Furthermore, out of 400 sample of class ''No one'', only 1 is mistakenly classified into class ''Single passing''.

Using the same method to examine the model's classification performance of samples in garage exit scenario. It is obvious that most of the samples of each class can be correctly classified, and only a few samples are confused during classification.

It is worth noting that in the same scenario, the model confuses samples of different categories with a very low probability. However, the samples between dormitory hall scenario and garage exit scenario are not confused (all cells with the light blue background have the value of 0 in Fig. 12), which shows that CSI sequence have different responses to the different-grained incidents. This might be due to the fact that the incidents in the two types of scenarios have different granularities and the CSI sequence can delicately capture the incidents with different granularities and retain the important features for distinguishing them.

TABLE 10. Comparison results of algorithm performance.

Methods	Dim.	Param. No.	Complexity	Memory/MB
Gao et al. [34]	3-D	$>3\times10^3$	$O(H{\cdot}W)$	> 13
Zhang et al. [35]	3-D	$>5\times10^3$	$O(H{\cdot}W)$	> 25
Fei et al. [36]	3-D	$>6.5\times10^4$	$O(\sum_{i=1}^{8} H_i \bullet W_i)$	>15
Khare et al. [37]	3-D	$>6\times10^2$	$O(H{\cdot}W)$	> 2.5
Jamil et al. [38]	3-D	$>1.5\times10^5$	$O(H{\cdot}W)$	> 70
Wei et al. [39]	3-D	$>3\times10^4$	$O(H{\cdot}W \log H{\cdot}W)$	> 10
Wang et al. [40]	3-D	$>5\times10^3$	$O(H{\cdot}W)$	> 5
3-D ConvNet [41]	3-D	$>2.3\times10^7$	$O(\sum_{i=1}^{N} H_i \bullet W_i)$	> 100
Two-Stream. [42]	3-D	$>2\times10^7$	$O(\sum_{i=1}^{N} H_i \bullet W_i)$	> 93
CNN-LSTM [43]	3-D	$>3.5\times10^7$	$O(\sum_{i=1}^{N} H_i \bullet W_i)$	> 105
Our method	**2-D**	**10**	**$O(200{\cdot}N)$**	**< 20**

3) PERFORMANCE ANALYSIS

The incident recognition completely relies on the way of the information representation. In the traditional incident representation methods, the information carrier is undoubtedly the video signal, and the representation method is either based on hand-crafted features or based on machine-learning feature map. The biggest difference from the previous works is that we use the CSI sequence of the Wi-Fi signal to represent the incident in this work. This novel way of presentation makes incident presentation more efficient. In this section, we compare the performance differences between our proposed method and previous typical methods, and safely draw the conclusion that the method based on CSI sequence representation of incidents can obtain higher performance.

Shot detection is an important step for video incident detection in the video-represented methods. The typical methods are absolute inter-frame difference [34], color histogram [35], frame pixel difference [36], frame correlation coefficient [37], compressed domain difference [38], edge tracking [39], motion vector [40] and some deep learning methods, such as 3-D ConvNet [41], Two-Stream CNN [42] and CNN-LSTM [43]. We compared the above method with our method in terms of the dimension of raw data, the number of parameters representing the incident, the time complexity of the feature extraction algorithm, and the memory required by the algorithm. The detailed experimental results are shown in Table 10.

As shown in Table 10, each frame of the video is an image, plus the time dimension, the video sequence is actually 3-D sequence data. Assume that the width and height of each video frame is H and W. In practical applications, the video resolution $H*W$ is often much greater than 320∗480. We conservatively assume that the video resolution is 320∗480 when estimating the performance of the algorithm. We also assume that the duration of an incident is t, the number of key frames representing the incident is N, and the number of features extracted per frame is M.

Analyzing the data in Table 10, it is not difficult to find that the video-represented method needs to process more complex 3-D video data when detecting and recognizing incidents. The amount of model parameters is huge (see ''Param. No.'' column in Table 10), the time complexity is high, and the memory required to process a single incident is large. In sharp contrast, the CSI-represented method relies on the ''record'' of environmental changes in the CSI sequence of the Wi-Fi signal when completing incident detecting and recognizing. According to the IEEE 802.11 standard, the radio wave between a pair of antennas contains 30 subcarriers. When the number of transmitting antennas is 1, the number of receiving antennas is 3, and the data sampling rate is 200 Hz, the size of one frame of the CSI data is only 720 B, which is much smaller than one frame of the video.

From the perspective of feature extraction, whether they are based on hand-crafted features or convolutional feature map, video-represented methods have high computational cost to locate the region of interest (ROI) in video frames. Besides, comparing to the CSI-represented methods, the time complexity of video-represented methods are much higher in the feature extraction process due to the variable background, lighting changes in video frames. Specifically, the time complexity of the video-represented methods are often positively correlated with the video resolution (see ''Complexity'' column in Table 10), and the situation even worsens as the number of convolutional neural network layers increases.

After representing incident by CSI sequence of Wi-Fi signal, the situation has greatly changed. In the subcarrier space of Wi-Fi signal, the impact of incidents on the signal is recorded in detail in the CSI sequence. Since CSI sequence is essentially 2-D signal, compared with 3-D video data, the computational cost of its feature extraction process is relatively low, the incident can be characterized by fewer features, and the required memory is also very large less (see ''Memory'' column in Table 10). In general, the algorithm performance is obviously better than the video-represented methods.

IV. DISCUSSION

Different from previous traditional works in which the video stream was the main object data to be processed, we took another way to go. By using the Wi-Fi signal as an auxiliary signal synchronized with the video stream and realizing incident detection and recognition in the surveillance scenario, a content-based video retrieval and recognition solution was provide.

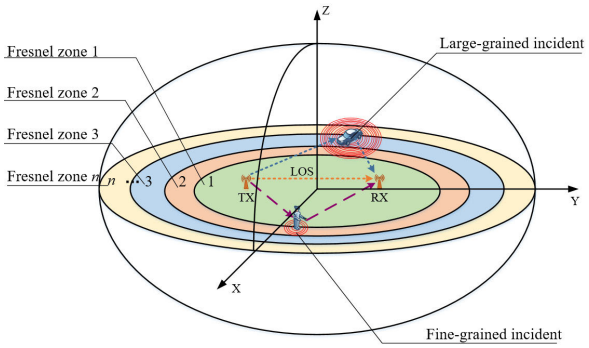

FIGURE 13. Different-grained incidents in fresnel zones of Wi-Fi signal.

From the experimental results, we can see that 1) the proposed incident detection algorithm is effective to locate the start time and track the duration of the incident in CSI sequence; 2) the proposed LSTM-based multi-classification model is able to classify the CSI segments into known categories with a high accuracy. The approach of using the CSI sequence of Wi-Fi signal as the processing object greatly reduces the computational complexity, and in some scenarios, it can prevent the video signal from leaking user privacy. Of course, if necessary, we can easily implement video retrieval based on SQL statements according to the database table obtained by the results of CSI incident detection and recognition. Furthermore, in some occasions where real-time performance is required, the proposed system can perform incident detection and recognition on real-time Wi-Fi signal, thereby overcoming the drawbacks of traditional video surveillance systems, that is, the system needs to wait until the video files are generated before data processing and lacks real-time performance.

Notably, we verified that the CSI sequence have different responses to the different-grained incidents. Jumping out of inherent pattern-based thinking mode, we try to make explanation from a model-based way. To the best of our knowledge, the Fresnel zone model [44] can meticulously describe the relationship between the motion position, granularity of the sensing target and the induced CSI power amplitude variations caused by the motion of the target. As depicted in Fig. 13, different subcarriers of Wi-Fi signal will form a series of concentric ellipsoids, namely Fresnel zones. When the sensing target moves from one Fresnel zone to another, the Wi-Fi signal will be positively enhanced or reversely weakened. Furthermore, the greater the granularity of the incident, the more Fresnel areas will be affected, and the more complex the spatial-temporal pattern of the CSI segment. Conversely, the number of Fresnel areas affected by fine-grained incident is smaller, and the signal spatial-temporal pattern is relatively simpler. This basically explains why the CSI sequence has different classification responses to the different-grained incidents.

However, things are far from over. As a novel sensing modality, the CSI sequence of Wi-Fi signal contains detailed information that can reflect electromagnetic changes in the physical environment. We have not yet developed and utilized the phase or other indicators of the signal, and the related research, such as the issues of position dependence, blind zone of sensing, and transfer learning in incident recognition, needs to be further in-depth.

V. CONCLUSION

Focusing on the retrieval and recognition of the incidents in the video stream, we have provided a novel solution based on the ubiquitous Wi-Fi signal in this work. First, an incident detection algorithm was developed to locate the start and end time of the incident in the CSI sequence of Wi-Fi signal that is is highly synchronous with the video stream. Second, a LSTM-based deep learning multi-classification network was trained on a dataset and used to classify the CSI segments into specific categories of the incidents, thus achieving the incident recognition. Third, we verified that the CSI sequence have different classification responses to different-grained incidents in the surveillance scenarios. Last, the experimental evaluation have been performed on the test set in two different scenarios. The experimental results have demonstrated that our proposed incident detection algorithm is effective to capture the incidents in video stream with an average error of 1.5 s, and that the developed LSTM-based multi-classification model is feasible and effective to recognize the CSI segments with an average value of 0.972, 0.973, 0.985, 0.972 and 0.962 for recall, precision, accuracy, F-1 score and Kappa coefficient, respectively.

Looking forward to the future, we make a plan to extend our work in the following directions.

- First, further improve the Wi-Fi-based incident sensing system and deploy it in security scenarios that have certain requirements for real-time and privacy protection.
- Second, we plan to make use of the phase and other indicators in CSI sequence to dig out the patterns of which corresponding to the physical changes.
- Third, we plan to extend Wi-Fi-based sensing technology to more fine-grained scenarios, such as respiration monitoring, gesture recognition, indoor localization, etc.
- Last, we intend to transplant the model to embedded systems such as Raspberry Pi and try to provide a possibility that the LSTM-based Wi-Fi signal detection and recognition system can play an active role in various IoT scenarios.

REFERENCES

[1] L. Rossetto, R. Gasser, J. Lokoč, W. Bailer, K. Schoeffmann, B. Muenzer, T. Soucek, P. A. Nguyen, P. Bolettieri, A. Leibetseder, and S. Vrochidis, "Interactive video retrieval in the age of deep learning—Detailed evaluation of VBS 2019," *IEEE Trans. Multimedia*, vol. 23, pp. 243–256, Mar. 2021, doi: 10.1109/TMM.2020.2980944.

[2] G. Pal, D. Rudrapaul, and S. Acharjee, "Video shot boundary detection: A review," in *Emerging ICT for Bridging the Future—Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2*. 2015, pp. 119–127.

[3] H. Chang and M. Zhang, "An algorithm of video shot boundary detection based on SVM," *Graph. Image*, vol. 20, no. 7, pp. 73–77, 2016.

[4] Y. Li, S. Lee, C.-H. Yeh, and C.-C. J. Kuo, "Techniques for movie content analysis and skimming: Tutorial and overview on video abstraction techniques," *IEEE Signal Process. Mag.*, vol. 23, no. 2, pp. 79–89, Mar. 2006.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[6] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, nos. 2–3, pp. 107–123, Sep. 2005.

[7] G. Willems, T. Tuytelaars, and L. Van Gool, "An efficient dense and scale-invariant spatio-temporal interest point detector," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 650–663.

[8] V. Kantorov and I. Laptev, "Efficient feature extraction, encoding, and classification for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2593–2600.

[9] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.

[10] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4694–4702.

[11] T. Han, W. Xie, and A. Zisserman, "Video representation learning by dense predictive coding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop*, Oct. 2019, pp. 1–10, doi: 10.1109/ICCVW.2019.00186.

[12] Z. Yang, Z. M. Zhou, and Y. H. Liu, "From RSSI to CSI: Indoor localization via channel response," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 1–33, 2014.

[13] L. Zhang, C. Wang, M. Ma, and D. Zhang, "WiDIGR: Direction-independent gait recognition system using commercial Wi-Fi devices," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1178–1191, Feb. 2020, doi: 10.1109/JIOT.2019.2953488.

[14] X. Wu, Z. Chu, P. Yang, C. Xiang, X. Zheng, and W. Huang, "TW-See: Human activity recognition through the wall with commodity Wi-Fi devices," *IEEE Trans. Veh. Tech.*, vol. 68, no. 1, pp. 306–319, Jan. 2019.

[15] Y. Lu, F. Wu, S. Tang, L. Kong, and G. Chen, "Pushing the limit of CSI-based activity recognition: An enhanced approach via packet reconstruction," in *Proc. IEEE Int. Conf. Sens., Commun., Netw.*, Jun. 2019, pp. 1–9, doi: 10.1109/SAHCN.2019.8824896.

[16] Z. Chen, L. Zhang, C. Jiang, Z. Cao, and W. Cui, "WiFi CSI based passive human activity recognition using attention based BLSTM," *IEEE Tans. Mobile Comput.*, vol. 18, no. 11, pp. 2714–2724, Nov. 2019.

[17] Z. Tang, Q. Liu, M. Wu, W. Chen, and J. Huang, "WiFi CSI gesture recognition based on parallel LSTM-FCN deep space-time neural network," *China Commun.*, vol. 18, no. 3, pp. 205–215, Mar. 2021.

[18] Z. Tian, Z. Li, M. Zhou, Y. Jin, and Z. Wu, "PILA: Sub-meter localization using CSI from commodity Wi-Fi devices," *Sensors*, vol. 16, no. 10, pp. 1–20, 2016, doi: 10.3390/s16101664.

[19] X. Wang, L. Gao, and S. Mao, "BiLoc: Bi-modal deep learning for indoor localization with commodity 5 GHz WiFi," *IEEE Access*, vol. 5, no. 99, pp. 4209–4220, 2017, doi: 10.1109/ACCESS.2017.2688362.

[20] H. Li, X. Zeng, Y. Li, S. Zhou, and J. Wang, "Convolutional neural networks based indoor Wi-Fi localization with a novel kind of CSI images," *China Commun.*, vol. 16, no. 9, pp. 250–260, Sep. 2019, doi: 10.23919/JCC.2019.09.019.

[21] D. Liu, Z. Liu, and Z. Song, "LDA-based CSI amplitude fingerprinting for device-free localization," in *Proc. Chin. Control Decis. Conf.*, Aug. 2020, pp. 2020–2023, doi: 10.1109/CCDC49329.2020.9164348.

[22] W. Xun, L. Sun, C. Han, Z. Lin, and J. Guo, "Depthwise separable convolution based passive indoor localization using CSI fingerprint," in *Proc. IEEE Wireless Commun. Netw. Conf.*, May 2020, pp. 1–6, doi: 10.1109/WCNC45663.2020.9120638.

[23] Q. Li, H. Fan, W. Sun, J. Li, L. Chen, and Z. Liu, "Fingerprints in the air: Unique identification of wireless devices using RF RSS fingerprints," *IEEE Sensors J.*, vol. 17, no. 11, pp. 3568–3579, Jun. 2017.

[24] Q. Xu, Y. Chen, B. B. Wang, and K. J. R. Liu, "Radio Biometrics: Human recognition through a wall," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 5, pp. 1141–1155, May 2017.

[25] J. Ding, Y. Wang, and X. Fu, "Wihi: WiFi based human identity identification using deep learning," *IEEE Access*, vol. 8, pp. 129246–129262, 2020, doi: 10.1109/ACCESS.2020.3009123.

[26] X. Tong, Y. Wan, Q. Li, X. Tian, and X. Wang, "CSI fingerprinting localization with low human efforts," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 372–385, Feb. 2021, doi: 10.1109/TNET.2020.3035210.

[27] J. Jung, H.-C. Moon, J. Kim, D. Kim, and K.-A. Toh, "Wi-Fi based user identification using in-air handwritten signature," *IEEE Access*, vol. 9, pp. 53548–53565, vol. 2021, doi: 10.1109/ACCESS.2021.3071228.
[28] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, p. 53, 2011.
[29] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements*, IEEE Standard 802.11nTM, 2009.
[30] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discovery Data*, vol. 6, no. 1, pp. 1–39, 2012.
[31] B. Hong, Q. Y. Tang, and F. S. Yang, "The properties of approximate entropy, mutual approximate entropy, fast algorithm and its preliminary application in EEG and cognitive research," *Signal Process.*, vol. 15, no. 2, pp. 100–107, 1999.
[32] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[33] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[34] F. Gao and Y. Lu, "Moving target detection using inter-frame difference methods combined with texture features and lab color space," in *Proc. Int. Conf. Artif. Intell. Adv. Manuf.*, Oct. 2019, pp. 76–81, doi: 10.1109/AIAM48774.2019.00022.
[35] Y. Zhang, "Object tracking based on similar background and color histogram in HSV color space," *Electron. Opt. Control*, vol. 26, no. 4, pp. 100–105, 2019.
[36] X. Liu, K.-A. Toh, and J. P. Allebach, "Pedestrian detection using pixel difference matrix projection," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1441–1454, Apr. 2020.
[37] B. R. Mounika, O. Prakash, and A. Khare, "Fusion of zero-normalized pixel correlation coefficient and higher-order color moments for keyframe extraction," in *Recent Trends in Communication, Computing, and Electronics*. 2018, pp. 357–364.
[38] A. Jamil, M. Majid, and S. M. Anwar, "An optimal codebook for content-based image retrieval in JPEG compressed domain," *Arabian J. Sci. Eng.*, vol. 44, no. 11, pp. 9755–9767, Nov. 2019.
[39] W. J. Heng and K. N. Ngan, "An object-based shot boundary detection using edge tracing and tracking," *J. Vis. Commun. Image Represent.*, vol. 12, no. 3, pp. 217–239, Sep. 2001.
[40] Z. Wang and Y. Zhu, "Video key frame monitoring algorithm and virtual reality display based on motion vector," *IEEE Access*, vol. 8, pp. 159027–159038, 2020, doi: 10.1109/ACCESS.2020.3019503.
[41] W. Li, N. Xu, and G. Liu, "Segments-based 3D ConvNet for action recognition," in *Proc. Int. Conf. Comput. Sci. Commu. Tech. (ICCSCT)*, 2020, pp. 353–359, doi: 10.1088/1742-6596/1621/1/012042.
[42] C. Liu, J. Ying, H. Yang, X. Hu, and J. Liu, "Improved human action recognition approach based on two-stream convolutional neural network model," *Vis. Comput.*, vol. 37, no. 5, pp. 1327–1341, 2021.
[43] Y. Guan, W. Hu, and X. Hu, "Abnormal behavior recognition using 3D-CNN combined with LSTM," *Multimedia Tools Appl.*, vol. 80, no. 19, pp. 18787–18801, 2021.
[44] H. Wang, D. Zhang, J. Ma, Y. Wang, Y. Wang, D. Wu, T. Gu, and B. Xie, "Human respiration detection with commodity WiFi devices: Do user location and body orientation matter?" in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, 2016, pp. 25–36.

YUSHENG HAO received the M.S. degree in computer science and technology from Lanzhou Jiaotong University, Lanzhou, in 2014. He is currently pursuing the Ph.D. degree with the Key Laboratory of China's Ethnic Languages and Information Technology of Ministry of Education, Northwest Minzu University. He is currently a Lecturer with the School of Mathematics and Computer Science, Northwest Minzu University. His research interests include signal processing, pervasive computing, and the IoT.

WEILAN WANG received the B.S. degree in mathematics from Northwest Normal University, Lanzhou, China, in 1983. From 2006 to 2007, she was a Visiting Scholar with Indiana University at Bloomington, Bloomington, IN, USA. She is currently a Professor with the School of Mathematics and Computer Science, Northwest Minzu University, Lanzhou. Her research interests include image processing, pattern recognition, and Tibetan information processing.

QIANG LIN received the Ph.D. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, in 2014. He is currently an Associate Professor with the School of Mathematics and Computer Science, Northwest Minzu University. His research interests include medical image computing, pervasive computing, intelligent information processing, and human behavior sensing.