SGRS: A Sequential Gesture Recognition System using COTS RFID

Bo Chen*, Qian Zhang*, Run Zhao†, Dong Li* and Dong Wang*
*School of Software, Shanghai Jiao Tong University, Shanghai, China
†Computer Science Department, Shanghai Jiao Tong University, Shanghai, China
Email: {chenbo.31, qwert3472, dong.l, wangdong}@sjtu.edu.cn, zhaorun@cs.sjtu.edu.cn

***Abstract*—Gesture recognition is an innovative technology which is fundamentally reshaping the way people live, entertain and work. However, most gesture recognition systems focus on the recognition of simple gestures and ignore the full potential of sequential gestures involving a series of temporally-related simple actions in order. This paper presents SGRS, a battery-free, scalable and non-specific sequential gesture recognition system based on COTS RFID. The key insight is that fine-grained phase information extracted from RF signals is capable of perceiving various gestures. In SGRS, we meticulously devise gesture recognition mechanism by incorporating the k-means based vector quantizer and string matching algorithm to enable precise and real-time sequential gesture identification. Moreover, an improved edit distance algorithm is proposed for suppressing individual diversity. We implement SGRS and comprehensively evaluate the performance by recognizing traffic command gestures of Chinese traffic police. Experimental result shows that SGRS achieves an average recognition accuracy of 96.2% with eight sequential gestures and is highly robust to both individual diversity and multipath effect.**

***Index Terms*—gesture recognition, sequential gestures, COTS RFID, edit distance**

I. Introduction

Recently, gesture recognition has drawn increasingly significant attention in Human-Computer Interaction (HCI). Compared with the traditional interaction approaches, such as keyboard and touchscreen, gesture interaction is more convenient and natural, which has fostered a broad range of emerging applications in gaming, health care, smart homes, etc.

While substantial progress has been achieved for gesture recognition, most researches focus on the recognition of simple gestures, involving a single hand action [1]–[6]. In contrast, sequential gestures, performed by more than one part of the body, are composed of a series of temporally-related simple actions in order [7]–[9]. They are more sophisticated and contain more abundant information. Therefore, sequential gestures as a way of HCI open new possibilities for enabling a variety of applications, such as motion gaming, sign language recognition, VR or AR controller, and so on. Unfortunately, only a few systems delve into this field and thereby this paper makes an exploration on the recognition of sequential gestures.

In terms of the supporting technologies, existing gesture recognition systems can be mainly classified into three categories. **Computer-vision-based systems** usually segment the video captured by camera into image sequences and employ machine learning algorithms to recognize gestures [1], [4], [7]. Although these approaches have a great advantage of high recognition accuracy, they are sensitive to Line of Sight (LOS) and light conditions extremely, and may raise serious privacy concerns. **Wearable-sensor-based systems** utilize multiple on-body motion sensors (accelerometer, gyroscope, etc.) to sense the motions of hands [5], [6], [8]. Wearable sensing methods are adept in perceiving subtle movements and thus achieve fine-gained gesture recognition. However, users are forced to wear varieties of heavy and expensive sensors, which enormously limit their freedom and reduce the scalability. Recent years have witnessed the flourishing of many innovative **RF-based systems**, which leverage the low-level signal characteristics, such as CSI, RSSI, phase and Doppler shift for localization [10], [11] and activity recognition [12]. Besides, gesture recognition systems, such as RFIPad [2], GRfid [3] and Wisee [9], have been proposed based on the primary underpinning that different gestures can result in distinguishing signal fluctuations. However, due to the signal superposition, these device-free RF-based gesture recognition systems cannot effectively differentiate signal fluctuations caused by multiple parts of the body, which handicaps the accurate identification of sequential gestures. Fortunately, by attaching tags with negligible weight on corresponding body parts, motion pattern of each part can be captured and depicted in a fine-grained manner, which facilitates the recognition of sophisticated gestures notably.

Inspired by these glittering systems above, we propose SGRS, a battery-free, scalable and non-specific Sequential Gesture Recognition System with Commercial Off-The-Shelf (COTS) devices. The key insight is that the reliable and fine-grained phase information extracted from RF signals can be regarded as an indicator to perceive these sequential gestures. However, rigorous challenges still exist and the primary one is how to describe these temporally-related actions of sequential gestures effectively. To combat with this puzzle, some representative time-frequency domain features are extracted into a feature vector, which can efficiently depict each action from the perspective of motion states reflected by all tags. Besides, it is especially crucial for HCI to reduce computational complexity as well as enable real-time interaction. Hence, k-means based vector quantizer is adopted to map the high-dimensional feature vectors into a discrete subspace of lower

This research was supported by a grant from Shanghai Municipal Development & Reform Commission. Dong Wang is the corresponding author.

978-1-5386-1734-2/18/$31.00 ©2018 IEEE

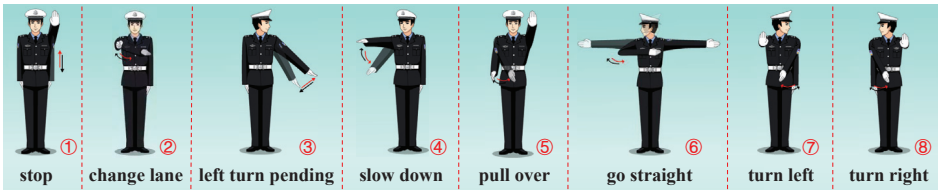


Fig. 1. The partial actions of eight traffic command gestures. Each gesture begins with attention and goes back to attention state after execution.

dimension by encoding each vector into a code, resulting in a low computational coded stream. Last but not least, how to effectively deal with individual diversity is a noticeable issue for commercial gesture recognition system. For this reason, an improved edit distance algorithm is designed to not only spot and identify gestures, but also increase intra-class similarity and reduce inter-class similarity for the sake of suppressing individual diversity. In summary, our main contributions can be concluded as follows:

(1) To the best of our knowledge, this is the first attempt to design a battery-free, scalable and non-specific sequential gesture recognition system based on COTS RFID.

(2) In order to recognize sequential gestures well and truly, we utilize the fine-grained RFID phase and propose a gesture recognition system containing four modules: Preprocessor, Feature Extractor, k-means based Vector Quantizer as well as Gesture Recognizer based on the improved edit distance.

(3) We implement SGRS and choose the traffic command gestures of Chinese traffic police as experiment subjects to evaluate the performance. Fig. 1 shows the partial actions of eight gestures and these gestures involve both simple gestures and sequential gestures, which can fully evaluate the performance of our system. The experimental result indicates that SGRS achieves an average accuracy of 96.2% and has strong robustness to individual diversity and multipath effect.

The rest of this paper is organized as follows. Section II presents the theoretical basis and overview of the system. Detailed designs of each module are described in Section III. Section IV demonstrates the implementation and evaluation of our system. Finally, we conclude our work in Section V.

II. System Model

In this section, we present the communication model of our passive Ultra High Frequency (UHF) RFID based system. Besides, we roughly introduce the overall of SGRS architecture.

A. Communication Model

A passive UHF RFID tag interacts with a reader via backscatter radio links and the relationship between backscatter signal $S(t)$ and phase $\varphi(t)$ can be expressed by:

$$S(t) = A(t)e^{-j\varphi(t)} \tag{1}$$

where $A(t)$ is the complex valued representation of attenuation. The backscatter signal $S(t)$ is the superposition of one direct path signal $S_{dir}(t)$ that travels between an antenna and a tag as well as reflected path signal $S_{mul}(t)$ including static multipath and dynamic multipath. Therefore, $S(t)$ can be also represented as follows:

$$S(t) = S_{dir}(t) + S_{mul}(t) \tag{2}$$

The phase of the direct path $\varphi_{dir}(t)$ [13] is associated with the distance $d(t)$ between the antenna and the tag as well as some hardware characteristics c [14], which can be further formulized as:

$$\varphi_{dir}(t) = \frac{4\pi}{\lambda} d(t) + c \tag{3}$$

Combined (1-3) together, the following equation can be inferred:

$$A(t)e^{-j\varphi(t)} = A_{dir}(t)e^{-j\frac{4\pi}{\lambda}d(t)+c} + S_{mul}(t) \tag{4}$$

Equation (4) indicates that the phase $\varphi(t)$ is determined by the distance $d(t)$ of the direct path and the multipath part $S_{mul}(t)$. Since the tag moves along with user's body, the phase variation is affected slightly by the multipath but dominated by the time-varying distance between the tag and the antenna, which makes SGRS robust enough to multipath effect.

B. System Overview

Fig. 2 demonstrates the overview framework of SGRS, which is composed of four modules: Preprocessor, Feature Extractor, Vector Quantizer as well as Gesture Recognizer.

In **Preprocessor**, a series of preprocessing measures are performed. The raw phase profiles are unwrapped to eliminate phase periodicity and then passed through the Hampel identifier as well as the weighted moving average filter in order to remove the outliers and smooth the signals, respectively. At last, linear interpolation is adopted to obtain evenly spaced phase samples. The **Feature Extractor** then divides the phase streams into a set of sliding windows and extracts some significant features of the time-frequency domain into a feature vector within each window. Besides, at the training stage, a gesture extractor is employed to segment the profiles corresponding to the gestures before feature extraction. The **Vector Quantizer** consists of two parts: a codebook builder and a quantizing encoder. At the off-line training stage, a codebook builder leverages k-means algorithm to cluster the feature vectors into k clusters and build a codebook. Each cluster represents a kind of action belonging to the sequential gestures. During the process of on-line recognition, a quantizing encoder is utilized to map each feature vector into a code (cluster index) based on its Euclidean distance from the cluster centroid, thus forming a coded stream. The **Gesture**

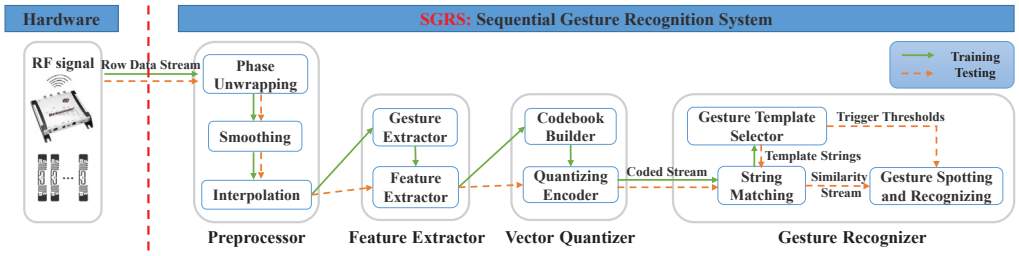


Fig. 2. System architecture of Sequential Gesture Recognition System.

Recognizer uses a gesture template selector to choose a template for each gesture and collect some statistics (e.g. trigger thresholds) by an improved edit distance algorithm. During the on-line identification process, the gesture recognizer calculates the similarities between the coded stream and each template, which generates a stream of similarity for each gesture. When the similarity exceeds the corresponding trigger threshold, a spotted occurrence of the particular gesture is reported.

III. System Methodology

In this section, we show a detailed demonstration of the methodology of SGRS and exhaustively analyze the core techniques, based on the aforementioned four basic functional modules shown in Fig. 2.

A. Preprocessor

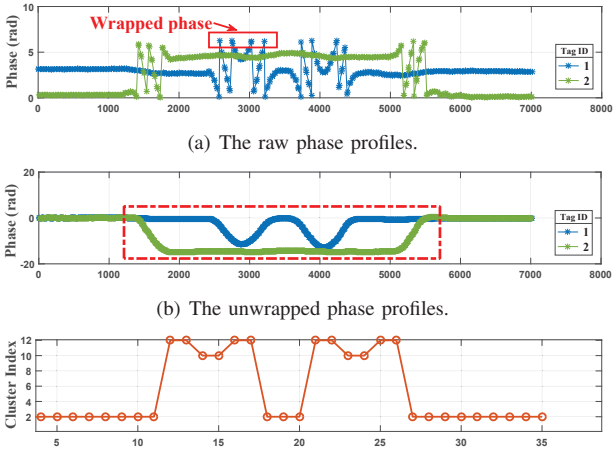
Phase Unwrapping The first step to process the raw phase profiles is unwrapping because the phase reported by the RFID reader is a periodic function with period 2π radians [14] (Fig. 3(a)). Hence, the approach in [15] is adopted to unwrap the phase readings, which assumes that the absolute difference of two adjacent phase values should be smaller than π and the unwrapped result is shown in Fig. 3(b).

Smoothing After phase unwrapping, SGRS uses a Hampel identifier [16] to eliminate the outliers induced by some burst noises. Then a weighted moving average filter is employed to further reduce the high-frequency noise. Furthermore, the DC (Direct Current) component is removed by subtracting the constant offset which can be calculated via a long-term averaging over the stream.

Interpolation In RFID communications, tags reply unevenly spaced in time domain due to tags collision, packet loss and other delays, which makes it difficult for subsequent processing, like feature extraction. Therefore, we adopt linear interpolation with 5ms apart between consecutive values to obtain evenly spaced phase stream.

B. Feature Extractor

Feature Extraction After preprocessing, the phase profiles are segmented into a set of sliding windows whose length is set to 50 with 50% overlap. A feature extractor is applied to calculate the statistical features of the time-frequency domain and select four features (showed in Table. I) that yield the best performance by adopting the mRMR (minimal-redundancy-maximal-relevance) method [17]. For the phase profiles of r tags, the phase profile of each tag within each window can be extracted with above four features, which means that the feature vector for the j-th window can be denoted as $F_j = \langle \alpha_1, \alpha_2, \ldots, \alpha_m \rangle$, where α refers to the feature and the number of features m satisfies $m = 4 \times r$. In this manner, a sequential gesture involving a series of simple actions can be defined as an ordered sequence of feature vectors.



(a) The raw phase profiles.

(b) The unwrapped phase profiles.

(c) The coded stream after vector quantizer.

Fig. 3. The phase profiles and coded stream of gesture 7 (namely, turn left).

TABLE I
TIME-FREQUENCY DOMAIN FEATURES

Feature	Description
Mean	Average phase
Standard deviation	Phase fluctuation
Peak-to-peak amplitude	Magnitude of phase discretization
Standard deviation of PSD*	Energy strength fluctuation

*PSD (power spectral density function) can be obtained by FFT.

Gesture Extraction At the training stage, a gesture extractor is deployed to extract the signal fragment corresponding to the gestures before feature extraction, such as the fragment in the red frame of Fig. 3(b). Inspired by the Foreground detection method widely used in image recognition field, we design a gesture detection method to segment the off-line phase profiles and extract gesture precisely. When the body is in the quiescent state, the phase values are close to zero owing to the preprocessing operation of removing DC component and

the phase distribution is regarded as background. Those that do not fit the background are considered as foreground and extracted. Hence, the phase fragments unrelated to the expected gestures can avoid being processed by feature extractor, free from taking a heavy toll on the follow-up analysis.

C. Vector Quantizer

After extracting the features into high-dimensional feature vectors, a vector quantizer is adopted to map the feature vectors into a discrete subspace of lower dimension by encoding each one into a finite set of codes. The coded stream requires less storage space and can be calculated quickly, which is suitable for real-time HCI applications. This module consists of two parts: an off-line codebook builder and an on-line quantizing encoder.

Codebook Builder At the off-line training stage, the codebook builder is designed to build a codebook. Given a set of feature vectors $\{F_1, F_2, ..., F_n\}$ as training data, where each one is a m-dimensional vector of a sliding window, the codebook builder leverages k-means algorithm [18] to partition the n feature vectors into k ($k \leq n$) clusters based on the Euclidean distance. The elbow method mentioned in [19] is adopted to select the suitable value of k by taking a measure of appropriateness for clusters. In our experiment, the radius of cluster is chosen as measurement criteria and the value of k is set to 12. For a particular cluster i whose centroid vector is represented as c_i, the average distance μ_i from centroid vector and standard deviation σ_i are calculated. Consequently, each cluster i is represented by a set of parameters $\langle i, c_i, \mu_i, \sigma_i \rangle$, which is considered as an item in the codebook.

Quantizing Encoder At the on-line identification stage, the quantizing encoder is utilized to encode each feature vector into a code (cluster index). The Euclidean distance d_i between the target feature vector and each centroid vector c_i is calculated and those that satisfy the constraint condition of $d_i \leq \mu_i + 3\sigma_i$ make up a set S. When S is a nonempty set, the feature vector can be encoded into the cluster index i corresponding to the minimum value in the set S, whereas the feature vector is encoded into the default cluster index 0. Thus, vector quantizer converts the stream of feature vector into the coded stream which is shown in Fig. 3(c).

D. Gesture Recognizer

Improved Edit Distance Edit distance algorithm [20] calculates the distance between two strings in terms of the minimum number of edit operations needed for transforming one to another. There are three edit operations: insertion, deletion and substitution. Suppose that $D(i, j)$ represents the minimum edit operation cost needed to transform the first i symbols of string S_1 into the first j symbols of string S_2, namely $S_1(1...i) \rightarrow S_2(1...j)$. $D(i, j)$ can be computed by dynamic programming principle, which is formulized as (5).

$$D(i,j) = min[D(i-1,j)+p, D(i,j-1)+q, D(i-1,j-1)+r] \tag{5}$$

where p, q, r represent the cost of insertion, deletion and substitution respectively. p and q are set to 1. r is set to 0, when $S_1(i) = S_2(j)$, while set to 1, when $S_1(i) \neq S_2(j)$. Therefore, the edit distance between S_1 and S_2 can be obtained by $D(m, n)$, where m, n denote the length of S_1 and S_2.

We observe that different gestures have distinct coded strings. However, owing to the individual diversity, the same gesture has highly similar but not identical coded strings, which can be visually described by the following instance.

$G_{11} = \{1, 1, 1, 1, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 1, 1\}$
$G_{12} = \{1, 1, 1, 3, 3, 3, 3, 4, 4, 4, 3, 3, 3, 1, 1, 1\}$
$G_{21} = \{1, 1, 1, 1, 5, 5, 5, 5, 4, 4, 4, 4, 1, 1, 1, 1\}$

where G_{11} and G_{12} are two strings belonging to gesture 1 while G_{21} is the string of gesture 2.

With the purpose of overcoming the problem mentioned above, we elaborately design an improved edit distance algorithm to further increase intra-class similarity and reduce inter-class similarity. Firstly, the cost of edit operations is reduced to $e(e < 1)$ when the current symbol is the same as the previous one, which can narrow the gap incurred by the variety of action duration and suppress individual diversity effectively. And then, the similarity between S_1 and S_2 is defined as:

$$T(S_1, S_2) = 1 - \frac{D(m,n) + C(m,n)}{max(m,n)} \tag{6}$$

where $C(m, n)$ stands for the additional cost of the unexpected symbols and is represented as:

$$C(m,n) = m - \sum_{i=1}^{m} \mathbf{1}_{S_2}(S_1(i)) + n - \sum_{j=1}^{n} \mathbf{1}_{S_1}(S_2(j)) \tag{7}$$

where $\mathbf{1}_A(x)$ is an indicator function, which is presented by:

$$\mathbf{1}_A(x) \coloneqq \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases} \tag{8}$$

In this way, the similarity between G_{11} and G_{12} is increased from 0.85 to 0.91, while the similarity between G_{11} and G_{21} is reduced from 0.62 to 0.12, which suppresses individual diversity effectively.

Gesture Template Selector Gesture template selector is designed to choose a template for each sequential gesture and generate the corresponding trigger threshold. The similarity $T(S_p, S_q)$ between any two strings S_p and S_q is calculated by the improved edit distance. For a specific gesture g, the string that maximizes intra-class similarity (the sum of similarities to all other candidates of the same gesture g) and minimizes inter-class similarity (the sum of similarities to all strings of other gestures) is selected as the template. Besides, the matching similarities between the selected template and all candidates of the gesture g are calculated. The mean μ_g and standard deviation σ_g of these matching similarities are computed and a gesture-related trigger threshold k_g is derived by following formula:

$$k_g = \mu_g - 3\sigma_g \tag{9}$$

Gesture Spotting and Recognition After receiving a new code from vector quantizer, the similarity between each template S_i and the coded stream is computed according to the improved edit distance, giving rise to a stream of matching

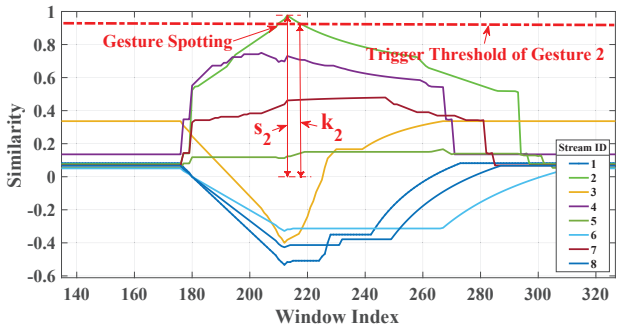


Fig. 4. Gesture spotting and recognition in the similarity stream.

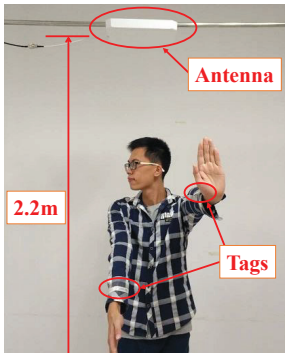


Fig. 5. Experimental implementation.

similarity for each gesture. Each similarity value is decided by the highest similarity between S_i and the substring of coded stream whose length ranges from $l_i/2$ to $2l_i$, where l_i is the length of S_i. When the similarity exceeds the corresponding trigger threshold k_i, a spotted occurrence of the particular gesture i is reported. Fig. 4 visually shows the process of spotting gesture 2 (namely, change lane).

However, since the spotting process is carried out in parallel for all gestures, temporal collisions between spotted gestures occur occasionally. It is worth noting that using absolute index, like similarity s_i, as judge standard for resolving conflict may lead to misdiagnosis. That is because the lengths of templates differ a lot due to variant complexity of sequential gestures. As a consequence, we choose the confidence level cl_i of colliding spotted gestures as judge standard, which is defined as follows.

$$cl_i = \frac{s_i}{k_i} \tag{10}$$

The gesture with the highest confidence level is treated as the identification result. Compared with the judge standard of similarity s_i, the precision rate of conflict resolution raises from 88.2% to 95.5%, which confirms its effectiveness.

The time complexity of the gesture recognition process is $O(km^2n)$ which seems high, but the lengths of both templates and coded streams are quite short thanks to the well-designed vector quantizer (e.g. the longest template only contains about 40 symbols). We evaluate the run time on a laptop equipped with an Intel Core i5 CPU running at 2.8GHz and the result shows that the average time required to process 125ms of data is about 8ms, which means that the laptop can easily handle real-time gesture recognition.

IV. Performance Evaluation

In this section, we first describe the implementation of SGRS and then thoroughly evaluate the performance of SGRS from the perspective of recognition accuracy, the effect of suppressing individual diversity, resisting multipath as well as diverse positions.

A. Implementation

We implement SGRS using COTS RFID devices without any hardware modification. As shown in Fig. 5, a directional antenna (Laird S9028PCR with $9dBi$ gain) is fixed at the ceiling of a room (2.2m high), connected with an ImpinJ R420 reader operating at a fixed frequency of 920.675MHz. The type of RFID tags is AZ-9640 with a size of $98mm \times 12mm$.

B. Recognition Accuracy Analysis

In order to explore the overall performance of SGRS for sequential gestures, 5 volunteers are invited to perform each of 8 traffic command gestures shown in Fig. 1 for 100 times. And then we perform 10-fold cross-validation on the phase profile dataset and obtain an average confusion matrix shown in Fig. 6. The average recognition accuracy is 96.2%, which indicates that SGRS not only can identify more sophisticated sequential gestures, but also achieves competitive accuracy compared with Wisee [9] (94%) and GRfid [3] (96.5%). Due to the packet loss incurred by large-scale action, the recognition precision of gesture 6 (namely, go straight) decreases to 89.2%, which can be improved by using higher-performance RFID antenna and tag. According to the confusion matrix, the average precision, recall and F1-scores are 0.98, 0.96 and 0.97 respectively, which illustrates its superior performance.

C. Effect of Suppressing Individual Diversity

In order to explore the robustness of SGRS against individual diversity, 15 volunteers are asked to perform each gesture for 50 times. These volunteers are 9 male and 6 female university students with the age ranging from 21 to 28. We collect the phase profiles from one of those volunteers as the training set to build the codebook and select templates, and then perform two sets of experiments. The per-participant validation uses the phase profiles of the same volunteer as the test set while the cross-participant validation uses the phase profiles of other volunteers. Fig. 7 depicts the performance impact of individual diversity and the error bar specifies the maximum, minimum and average accuracy under different volunteers. We can conclude that the cross-participant validation can still reach a high average accuracy of 94.6%, only a slight drop compared to the per-participant validation (96.8%), which verifies the effectiveness of suppressing individual diversity.

D. Effect of Resisting Multipath

One of the main drawbacks of RF-based systems, such as GRfid [3] and Wisee [9], is susceptible to multipath effect,

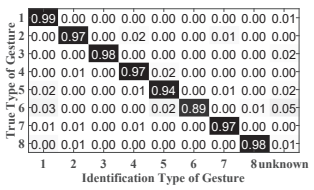


Fig. 6. Confusion matrix of gesture recognition accuracy.

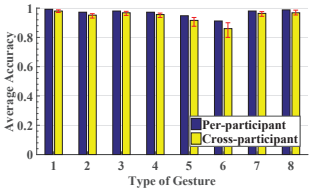


Fig. 7. Effect of suppressing individual diversity.

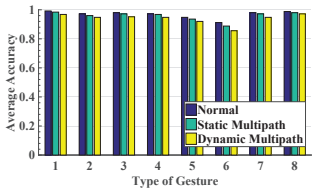


Fig. 8. Effect of resisting multipath.

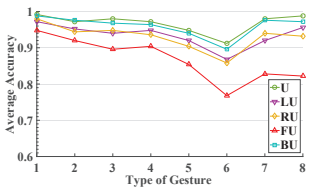


Fig. 9. Effect of diverse positions.

which may easily incur instability in practice. By contrast, SGRS is insensitive to ever-changing surroundings, which has been briefly touched upon in section II. In order to visualize the capability of resisting multipath, we perform two kinds of experiments and collect each gesture for 50 times. In the static multipath scenario, a volunteer is performing gestures and a metallic board is laid aside. In the dynamic multipath scenario, two to five volunteers are asked to walk around when a volunteer is performing gestures. Experimental results shown in Fig. 8 illustrate that the average accuracies in static multipath (95.8%) and dynamic multipath scenarios (93.5%) do not have obvious degradation, which demonstrates the stability and feasibility in daily life.

E. Effect of Diverse Positions

In order to evaluate the effect of diverse positions, we take data collected right under the antenna (recorded as U) as training set and data of other positions for testing. We collect each gesture for 50 times in four different positions, namely, left of U (LU), right of U (RU), front of U (FU) and back of U (BU) whose distances from U are about 20cm to 50cm. Fig. 9 shows the average accuracy of recognizing gestures over different positions. Compared with the baseline U, SGRS achieves analogous performance in position LU, RU and BU, especially the position BU. However, due to the disturbance of the human body, the average accuracy in position FU degrades to 86.8%. It is noteworthy that we can expand the scope of identification by using multifarious improvement measures, such as enhancing transmitting power, deploying multi-antenna and so on.

V. Conclusion

In this paper, we propose SGRS, a sequential gesture recognition system based on COTS RFID. The distinctive phase fluctuations related to specific gesture patterns are regarded as a particular indicator to recognize sequential gestures. SGRS consists of four functional components, including Preprocessor, Feature Extractor, k-means based Vector Quantizer as well as Gesture Recognizer based on the improved edit distance. We choose eight traffic command gestures as experiment subjects to evaluate the performance comprehensively and the experimental results demonstrate that SGRS can achieve an average recognition accuracy of 96.2%. Meanwhile, SGRS shows strong robustness to individual diversity and multipath effect, which validates its robustness and feasibility.

References

[1] B.-Y. Chen, C.-J. Chiu, and K.-T. Feng, "Particle-based window rotation and scaling scheme for real-time hand recognition and tracking," in *Proceedings of IEEE WCNC*, 2017.

[2] H. Ding, C. Qian, J. Han, G. Wang, W. Xi, K. Zhao, and J. Zhao, "Rfipad: Enabling cost-efficient and device-free in-air handwriting using passive tags," in *Proceedings of IEEE ICDCS*, 2017.

[3] Y. Zou, J. Xiao, J. Han, K. Wu, Y. Li, and L. M. Ni, "Grfid: A device-free rfid-based gesture recognition system," *IEEE Transactions on Mobile Computing*, 2017.

[4] H.-K. Lee and J.-H. Kim, "An hmm-based threshold model approach for gesture recognition," *IEEE Transactions on pattern analysis and machine intelligence*, 1999.

[5] S. Shen, H. Wang, and R. Roy Choudhury, "I am a smartwatch and i can track my user's arm," in *Proceedings of ACM MobiSys*, 2016.

[6] R. I. Ramos-Garcia, E. R. Muth, J. N. Gowdy, and A. W. Hoover, "Improving the recognition of eating gestures using intergesture sequential dependencies," *IEEE journal of biomedical and health informatics*, 2015.

[7] A.-Y. Park, S.-W. Lee *et al.*, "Gesture spotting in continuous whole body action sequences using discrete hidden markov models," in *Gesture Workshop*. Springer, 2005, pp. 100–111.

[8] X. Yang, X. Chen, X. Cao, S. Wei, and X. Zhang, "Chinese sign language recognition based on an optimized tree-structure framework," *IEEE journal of biomedical and health informatics*, 2017.

[9] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proceedings of ACM MobiCom*, 2013.

[10] R. de Amorim Silva and P. A. d. S. Gonçalves, "Enhancing the efficiency of active rfid-based indoor location systems," in *Proceedings of IEEE WCNC*, 2009.

[11] L. Shangguan, Z. Zhou, and K. Jamieson, "Enabling gesture-based interactions with objects," in *Proceedings of ACM MobiSys*, 2017.

[12] T.-H. Chiang, Y.-T. Chuang, C.-L. Ke, L.-J. Chen, and Y.-C. Tseng, "Calorie map: An activity intensity monitoring system based on wireless signals," in *Proceedings of IEEE WCNC*, 2017.

[13] P. V. Nikitin, R. Martinez, S. Ramamurthy, H. Leland, G. Spiess, and K. Rao, "Phase based spatial identification of uhf rfid tags," in *Proceedings of IEEE RFID*, 2010.

[14] Impinj, *Speedway revolution reader application note - low level user data support*, 2013, https://support.impinj.com/hc/en-us/articles/202755318-Application-Note-Low-Level-User-Data-Support.

[15] T. Liu, L. Yang, X.-Y. Li, H. Huang, and Y. Liu, "Tagbooth: Deep shopping data acquisition powered by rfid tags," in *Proceedings of IEEE INFOCOM*, 2015.

[16] L. Davies and U. Gather, "The identification of multiple outliers," *Journal of the American Statistical Association*, 1993.

[17] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, 2005.

[18] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 1967.

[19] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, 1999.

[20] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Tröster, "Wearable activity tracking in car manufacturing," *IEEE Pervasive Computing*, 2008.