RFlow-ID: Unobtrusive Workflow Recognition with COTS RFID

Jinshi Zhang[†], Qian Zhang[†], Dong Li[†], Run Zhao[‡] and Dong Wang[†*]

[†]School of Software, Shanghai Jiao Tong University, China
[‡]Computer Science Department, Shanghai Jiao Tong University, China
{zhangjinshi,qwert3472,dong.l,wangdong}@sjtu.edu.cn
zhaorun@cs.sjtu.edu.cn

ABSTRACT

Workflow recognition is a key technique in the field of activity recognition with benefits of monitoring the step being performed in the workflow, detecting the missing step, and providing assistance to the performer of the workflow, among others. In this paper, we present an unobtrusive workflow recognition system called RFlow-ID, which is the first device-free, battery-free and privacy-preserving workflow recognition system based on RFID technique. RFlow-ID perceives the use and movement of associated objects in the workflow using fine-grained phase information extracted from low-level RF signal, and infers the most likely sequence of workflow activities via a VQ-HMM model. We implement RFlow-ID on COTS RFID devices and evaluate it through a common biomedical experiment. The results validate the high recognition accuracy and robustness of our system.

CCS CONCEPTS

• **Human-centered computing → Ubiquitous and mobile computing systems and tools**;

KEYWORDS

workflow recognition, COTS RFID, VQ-HMM

ACM Reference format:
Jinshi Zhang[†], Qian Zhang[†], Dong Li[†], Run Zhao[‡] and Dong Wang[†]. 2017. RFlow-ID: Unobtrusive Workflow Recognition with COTS RFID. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Melbourne, VIC, Australia, November 7–10, 2017 (MobiQuitous 2017),* 10 pages.
https://doi.org/10.1145/3144457.3144463

1 INTRODUCTION

Activity recognition is one of the most hot topics in the area of ubiquitous computing. It has fostered a broad range of innovative applications in eldercare[37], healthcare[10] and smart homes[18], etc.. Basically, the goal of activity recognition is to infer the humans' behaviors from a series of observations on the humans' actions, the vital signs and the environmental conditions. For example, in an eldercare scenario, the caregivers use sensors to recognize the Activities of Daily Living (ADL) of elderly people such as walking, eating and lying down in order to provide them with the proactive assistance[31].

Although many researchers have made explorations into activity recognition, few of them focused on recognizing the complex workflow of activities. The workflow is generally defined as a temporally ordered group of procedural activities (steps) for accomplishing a task in which people and tools are involved in each step of the process[4]. Unlike ADL recognition, in which the intrinsic feature is the ability to recognize isolated actions or activities (e.g. walking, eating, lying down, etc.)[30], the workflow recognition is more complex and it needs to have context-aware capabilities since each step in the workflow is temporal dependent. For example, the workflow recognition needs to remind the novice what the next step is, and whether there are missing steps.

Existing research on automatic workflow recognition mainly relies on wearable sensors, environmental sensors and cameras. (1) Wearable-sensor-based methods utilize on-body motion sensors (accelerometer, gyroscope, etc.) to sense the movements of body parts and infer steps of the workflow[25, 35, 42, 45]. For example, authors in [35] presents a wearable system which combines Google's Glass with a wrist-worn accelerometer to capture and and recognize steps in a wet laboratory environment. Although the sensor-based methods have the ability to identify the workflow, the requirement of wearing additional devices on the body is obtrusive and inconvenient to people. (2) Environmental-sensor-based methods deploy dedicated sensors to record the environment variables such as temperature, dust and pressure and provide context-aware activity recognition[11, 17, 44]. These methods are unobtrusive since they are device-free for human body, but they require batteries to supply these sensors, which leads to significant energy consumption and deployment overhead. (3) Camera-based solutions employ cameras to record the video sequence and recognize the workflow with computer vision algorithms[19, 29, 39, 40], such as a weakly supervised workflow recognition framework which is trained from unlabeled videos of recorded experiments in a synthetic biology laboratory[19]. The camera-based solutions can provide fine-grained workflow recognition, but they may raise serious privacy concerns and not suitable for deployment in non-public areas (laboratories, workstations, etc.). Moreover, the strict requirements for light conditions and high computational cost are also the drawbacks of these solutions.

*Dong Wang is the corresponding author

MobiQuitous 2017, November 7–10, 2017, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5368-7/17/11…$15.00
https://doi.org/10.1145/3144457.3144463

In this paper, we present a device-free, battery-free and privacy-preserving solution called RFlow-ID to recognize the workflow based on the passive UHF RFID system. The RFlow-ID originally exploits Commercial Off-The-Shelf (COTS) RFID products to infer the workflow from the status of used objects. It is well-known that RFID (radio-frequency identification) is an enabling tool for automatic identification of objects, which mainly consists of a RFID reader and several RFID tags. The RFID reader uses electromagnetic fields to identify, track and supply energy to the RFID tags attached to objects (cartons, books, people, etc.). Normally, the RFID technology is used in tracking of goods and access management. However, recent research has shown that the phase information from the physical RF signal between the RFID reader and tags is a powerful object motion sensing modality for many applications such as indoor localization[41], sleep diseases monitoring[27] and gait identification[38]. Therefore, we attempt to leverage the phase information received from RFID tags affixed on the relevant objects to sense the workflow.

Based on the phase information collected from the RF signal, we apply a sliding-window based feature extraction method to process the phase data in real time, and introduce the VQ-HMM model which is widely used in speech recognition to establish the model of workflow recognition. As each step in the workflow may consist of some actions such as picking up and rotating the object, previous workflow models[35, 45] need to predefine the typical actions in the workflow by manual analysis, and then use classification algorithm such as kNN to identify these typical actions for establishing the timing model. However, the typical actions may change with another different workflow, which requires new analysis by the human, for example, the manual production line may replace the processed product every few months due to the new order, resulting in different processing actions, and these models are difficult to adapt to the change of workflow. In contrast, our method do not predefine the typical actions by human analysis, but allow the model to automatically cluster the statistical features. As a more general method, VQ-HMM leverages VQ (vector quantization) technique which is implemented through K-means algorithm, to cluster and compress feature vectors from the phase profile into a highly representative low-dimension code sequence. Then, it establishes the timing model using HMM (hidden Markov model), which can estimate the most probable hidden states (activities in the workflow) based on the observations (the code sequence). As a special form of HMM, VQ-HMM has benefits including less requirements for training data than traditional HMM model because it does not need training data for any action, and less computation overhead since the dimension of features is reduced.

We chose an application scenario of subculturing cells, a basic experiment usually done by one person in the biomedical laboratory, for testing the feasibility of our system. Subculturing, or passaging cells is frequently used in biomedical experiments, as it allows rapid culture and expansion of cell types for experimental analysis. The experiment of subculturing cells is a common procedure wherein cells from a given culture are divided into new cultures and fed with fresh media to facilitate further expansion. This procedure is a typical workflow, containing 12 sequential activities (steps) as shown in Table1. To ensure the success of subculturing cells, the entire workflow needs to be executed step by step. Recognizing each activity in the workflow has three main benefits including (1) guiding through the experiment, for example, reminding the novice what the next step is. (2) checking for errors in performing workflow activities, such as detecting if there are missing steps. (3) automatically recording time information in performing workflow activities, such as providing the duration of each step for skill assessment.

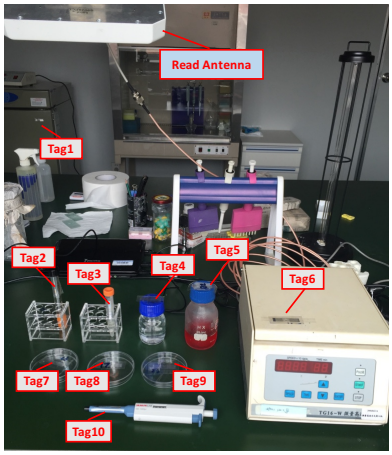


Figure 1: Application scenario

Figure 1 shows the deployment of RFlow-ID system. In most biological experiments, the researchers' activities involve interactions with experimental items or instruments. During the execution of the subculturing experiment, each activity in the workflow also involves an interaction with experimental objects such as pipette, petri dishes, and so on, which are listed in Table 1. These relevant objects interacted by the participant are affixed RFID tags, and a RFID reader is deployed on the ceiling above the experimental platform.

The main contributions of the RFlow-ID system are summarized as follows:

- To the best of our knowledge, this is the first attempt to design a device-free, battery-free and privacy-preserving workflow recognition system based on the COTS RFID. We have demonstrated that the low-level phase profiles from RF signal are rich enough for workflow recognition in a biological experiment that requires a series of refinement processes.
- We propose to use VQ-HMM algorithm to model the general workflow recognition system based on the data of RFID phase profiles with less computational overhead.
- We implement RFlow-ID and comprehensively evaluate the performance of RFlow-ID. Experimental results show that RFlow-ID can achieve 90.4% recognition accuracy in recognizing each step of the workflow.

The rest of this paper is organized as follows. In Section 2, the background of RFID technique is introduced. Section 3 describes the architecture of RFlow-ID, and presents the design of each component. The implementation and evaluation results of RFlow-ID

are presented in Section 4, and Section 5 reviews the related works from literatures. Section 6 concludes the paper.

Table 1: The workflow of subculturing cells containing 12 sequential activities

workflow of subculturing cells		associated items
A1	remove the tissue culture media	pipette (tag10), petri dish A (tag7)
A2	wash the cells with PBS	pipette (tag10), petri dish A (tag7), PBS (tag4)
A3	add trypsin to the cells	pipette (tag10), petri dish A (tag7), trypsin tube (tag2)
A4	incubate cells in CO2 culture incubator for about 1 minute	petri dish A (tag7), CO2 culture incubator (tag1)
A5	take out the petri dish from the CO2 culture incubator	petri dish A (tag7), CO2 culture incubator (tag1)
A6	stop the proteolysis by adding fresh tissue culture media	pipette (tag10), petri dish A (tag7), tissue culture media container (tag5)
A7	spray cells with pipette	pipette (tag10), petri dish A (tag7)
A8	transfer the cell suspension into a conical tube for centrifugation	pipette (tag10), petri dish A (tag7), conical tube (tag3)
A9	put the conical tube into centrifuge	conical tube (tag3), centrifuge (tag6)
A10	prepare two petri dishes with fresh tissue culture media	pipette (tag10), petri dish B (tag8) and C (tag9), tissue culture media container (tag5)
A11	take out the conical tube from the container	conical tube (tag3), centrifuge (tag6)
A12	transfer the cell suspension to the two petri dishes	pipette (tag10), conical tube, petri dish B (tag8) and C (tag9)

2 BACKGROUND

Passive UHF RFID systems work using backscatter communication. Tags harvest power from the radio signal emitted by the readers, and reply to the readers' query by modulating the backscatter signal using ON-OFF keying. Present COTS RFID readers, for example, Impinj Speedway in our implementation, can not only obtain the ID of tags, but also report the low-level backscatter signal characteristics such as RSSI, phase and Doppler shifts[13]. Prior work[47][26] has demonstrated that the detail phase information exhibits a more reliable and fine-grained indicator of multipath changes than other metrics. Thus our system exploits the phase information for workflow recognition.

The relationship between the time-varying backscatter signal $s(t)$ and its phase information $\varphi(t)$ can be expressed by

$$s(t) = a(t) \cdot e^{-j\varphi(t)} \tag{1}$$

where $a(t)$ is the complex valued representation of attenuation. Theoretically, the phase changes with the signal propagation distance $d(t)$ between the reader and the tag[46], denoted by

$$\varphi(t) = \frac{4\pi}{\lambda} d(t) + c \tag{2}$$

where c is initial phase offset which is related to the hardware characteristics. However, in most practical RFID setups, the backscatter signal between the reader and the tag is the superposition of one direct path signal $s_{dir}(t)$ and many other reflected path signal $s_{mul}(t)$ due to the multipath effect from other objects such as walls, desks and human hands, etc.. Let us consider a dynamic environment, for example, a person interacts with the object attached with the tag in the workflow activity, the number of reflected paths $N(t)$ may also change over time due to the movement of people, so the backscatter signal $s(t)$ can be expressed by the following equation:

$$\begin{aligned} s(t) &= s_{dir}(t) + s_{mul}(t) \\ &= a_{dir}(t) \cdot e^{-j\varphi_{dir}(t)} + a_{mul}(t) \cdot e^{-j\varphi_{mul}(t)} \\ &= a_{dir}(t) \cdot e^{-j\frac{4\pi}{\lambda} d_{dir}(t) + c_{dir}} + \sum_{k=1}^{N(t)} a_k(t) \cdot e^{-j\frac{4\pi}{\lambda} d_k(t) + c_k} \end{aligned} \tag{3}$$

Therefore, the received signal phase information $\varphi(t)$ can be further derived as

$$\varphi(t) = \arctan \frac{a_{dir}(t) \sin \varphi_{dir}(t) + a_{mul} \sin \varphi_{mul}(t)}{a_{dir}(t) \cos \varphi_{dir}(t) + a_{mul}(t) \cos \varphi_{mul}(t)} \tag{4}$$

Equation (3) and (4) indicate that the phase is determined by the distance $d_{dir}(t)$ of the direct path, the distance $d_k(t)$ of each reflected path, and the number of the reflected path N_t. When a person is approaching the object attached with the tag, N_t and some $d_k(t)$ will change, and when the person picks up and uses the object, $d_{dir}(t)$, N_t and some $d_k(t)$ will change, eventually leading to the phase change. In theory, if a person interacts with the tag multiple times in the same way, that is, $d_{dir}(t)$, N_t and $d_k(t)$ have the same (similar) change in each interaction, then the phase will have the same change in each interaction.

We carry out a preliminary experiment to validate our conclusion. A person is asked to perform the step A3 in Table 1 (add trypsin to the cells) 3 times. During the execution, we study the phase profiles of some associated tags (tag2 on the trypsin tube and tag10 on the pipette), and another tag (tag4 on the PBS) that is not used in A3. Figure 2 depicts the experimental scene and the phase profiles for each tag, where each tag provides the data for the 3 executions and expressed as, for example, $Tag2'$,$Tag2''$ and $Tag2'''$. We can see that the phase of each tag has an amplitude change over time in the execution, and even the tag4 (unused item) also has a large change from 12s to 15s. It is also observed that the phase profiles for each execution are highly similar, which is consistent with our theoretical speculation. This fact motivates our design of RFlow-ID, which will be introduced in the next section.

3 SYSTEM DESIGN

In this section, we present the design of RFlow-ID. We start by an overview of RFlow-ID architecture. Then, we lay out detailed description on each module in subsequent subsections.

3.1 System Overview

Figure 3 shows the overview of our system. The RFlow-ID architecture consists of five main modules: the phase collector, the feature extractor, the vector quantization (VQ) module, the HMM module and the sequence analysis (SA) module. The RFlow-ID first monitors and abstracts the raw phase profile of each RFID tag from the RF signal using the phase collector. A series of preprocessing processes are performed in the phase collector, for example, 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 to remove the outliers induced by burst noises and smooth the values in the phase collector. The feature extractor then divides the streaming phase profiles into a set of sliding

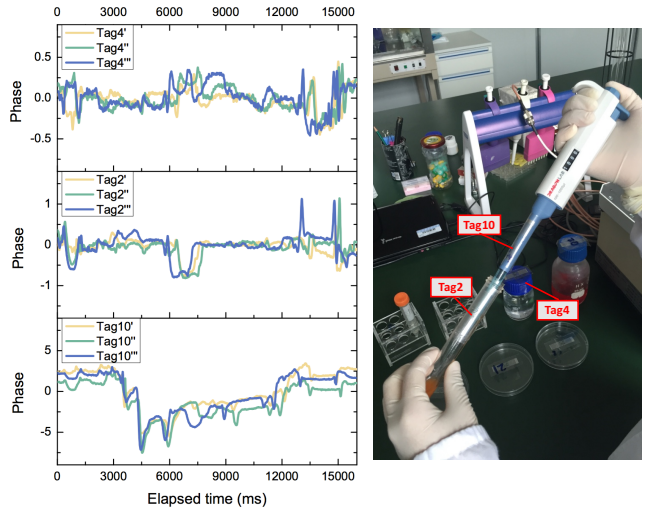


Figure 2: The phase profiles of tag4, tag2 and tag10 are collected when the participant performs step A3 in the workflow

windows and extracts the novel features such as standard deviation and signal energy into a feature vector from the phase profiles within each sliding window. And then, the VQ module leverages a codebook builder to cluster the feature vectors of all windows and uses the category number and its centroid point as an element in the codebook, with respect of the training set. And a quantizing encoder is used to encode a feature vector into the category number based on its Euclidean distance from the centroid in the codebook. After that, the feature vectors of a series of windows are encoded into a coding sequence, provided to the HMM module, which uses Baum-Welch Algorithm to train the parameters of the model and the Viterbi algorithm to estimate the most probable hidden states (activities of the workflow) with the coding sequence. At last, the SA module analyzes the duration of each activity and gives the final speculative information, for example, it is able to remind what the next step is, and whether there are missing steps.

3.2 Phase Collector

The phase collector is responsible for monitoring and abstracting the raw phase profile of each RFID tags from the RF signal. A series of preprocessing processes are performed in the phase collector as follows.

Phase Unwrapping The first step to process the raw phase measurements is phase unwrapping[47]. The reason is that the signal phase reported by the reader is a periodic function ranging from 0 to 2π, termed as wrapped phase[47]. When the phase changes, it may increase gradually to 2π, then jump to 0, or may decrease to 0, then continue to decrease from 2π, leading to inconvenience for further analysis. We adopt the method in [23] to unwrap the phase values, which assumes the absolute difference of two adjacent reading phase value is smaller than π.

Smoothing After phase unwrapping, RFlow-ID uses the Hampel identifier to eliminate the outliers induced by some burst noises. It declares the phase samples out of the interval $[\mu - \gamma \times \delta, \mu + \gamma \times \delta]$ as outliers, which means replacing any samples in the sliding window that is more than or less than γ times median absolute deviation (MAD) from the median μ of the window with this median. The sliding window is composed of the current sample and its 2γ surrounding samples, and the most widely used value of γ is 3. Then we employ a weighted moving average filter to further reduce the high-frequency noise and smooth the data. At last, the system removes the Direct Current(DC) component of phase stream by subtracting the constant offset which can be calculated via a long-term averaging over the stream.

Interpolation In RFID communications, the tags reply unevenly spaced in time domain due to tags collision, packet loss and other delays. Even though there exists only one tag in the interrogation area, the phase sequences are still uniformly sampled. This makes it difficult for further process, like feature extraction. To obtain evenly spaced phase samples, we adopt linear interpolation with 5ms apart between consecutive values to process the phase stream.

3.3 Feature Extractor

The feature extractor is responsible for dividing the streaming phase profiles into a set of sliding windows, and extracting the novel features into a feature vector from the phase profiles within each sliding window. Assuming that the workflow involves r RFID tags, the collected phase profiles for all tags can be denoted by $S = \{\varphi_1, \varphi_2, \cdots, \varphi_r\}$. For each phase profile, we divide the entire sampling sequence into a set of sliding windows, with window size w, that means a window contains w phase sampling points (In our implementation, the window size is 100 and the overlap is 0.5). The set of phase profiles in the j_{th} window is therefore

$$S_j = \left\{\left\langle \varphi_{1,j}, \varphi_{1,j+1}, \cdots, \varphi_{1,j+w}\right\rangle, \cdots, \left\langle \varphi_{r,j}, \varphi_{r,j+1}, \cdots, \varphi_{r,j+w}\right\rangle\right\} \tag{5}$$

Feature Selection Now we should select features from the sampling sequence over the sliding window. In one sliding window, we let $x_n \in \left\langle \varphi_{i,j}, \varphi_{i,j+1}, \cdots, \varphi_{i,j+w}\right\rangle$ refers to one particular phase value with respect to tag i in the window segment. We calculate the statistical features in the time-frequency domain and adopt the MRMR method[32] to select three features that yield the best performance. The first feature is standard deviation, which reflects the degree of data discretization. We observe that the standard deviation of the phase profile is almost zero when the object affixed tag is still, and the standard deviation is much greater than zero when the object is picked up or moved. Thus, standard deviation is an important feature of distinguishing object motion or still. The standard deviation is defined as

$$Std = \sqrt{\frac{1}{w}\sum_{n=1}^{w}(x_n - \bar{x})^2} \tag{6}$$

where $\bar{x}$ is the mean of the phase sampling sequence in the window. The second feature is the peak-to-peak amplitude, which represents the change between the peak (highest value) and trough (lowest value) of the signal. The peak-to-peak amplitude can help distinguish between activities that are characterized by different ranges of movements, and it is defined as

$$P2PA = max\,(x) - min\,(x) \tag{7}$$

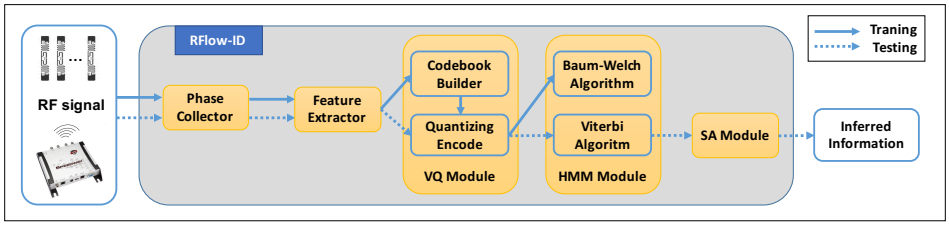


Figure 3: Architecture of RFlow-ID

where $max\,(x)$ refers to the maximum in the window segment while $min\,(x)$ refers to the minimum value. The third feature is the signal energy which refers to the area between the signal curve and the time axis. The signal energy reflects the overall signal strength which is expressed as

$$E = \sum_{n=1}^{w} x_n^2 \tag{8}$$

For the phase profiles of r tags, each phase file can be extracted with three features, which means that the set of phase profiles S_j is extracted as $F_j = \langle \alpha_1, \alpha_2, \cdots, \alpha_m \rangle$, where α refers to the feature and the number of the features is $m = 3 \times r$.

Feature Scaling If a feature has a larger range of values than the other features, then the further analysis result will be dominated by this feature. Feature scaling technique is used to scale the feature values and normalize them between 0 and 1. For any feature α_i in F_j, a Min-Max scaling approach is implemented by this formula:

$$\alpha_i{}' = \frac{\alpha_i - min\,(\alpha)}{max\,(\alpha) - min\,(\alpha)} \tag{9}$$

This method scales the feature values a fixed range 0 to 1, where the α_i' is the normalized feature values, and $max\,(\alpha)$ and $min\,(\alpha)$ are the maximum and minimum values of the original feature values, respectively. Consequently, the output of feature extractor for the j_{th} sliding window is $F_j = \langle \alpha_1{}', \alpha_2{}', \cdots, \alpha_m{}' \rangle$.

3.4 VQ Module

Vector quantization (VQ) is an efficient technique that was originally used for data compression and has been successfully applied in various recognition applications such as speech recognition [5] and pattern recognition[2]. The essence of VQ is a lossy compression method, which encodes values from a multidimensional vector space into a finite set of values from a discrete subspace of lower dimension. With VQ module, we do not predefine the typical actions by manual analysis since it automatically analyzes and summarizes the multidimensional feature vector. In addition, VQ is suitable for real-time applications such as fast planar-oriented ripple search method[6] and real time video-based event detection[21] since the lower-dimension vector requires less storage space and achieves faster processing speed

The basic idea of VQ module is to map (or compress) the m-dimensional vector, that is the output of the feature extractor $F_j = \langle \alpha_1{}', \alpha_2{}', \cdots, \alpha_m{}' \rangle$ in Section 3.3, to a finite set $cb = \{c_1, c_2, \cdots, c_k\}$. The set cb is called codebook which can be generated in spatial domain by clustering algorithms such as the classical Linde-Buzo-Gray (LBG) algorithm[34] and the K-means based optimization

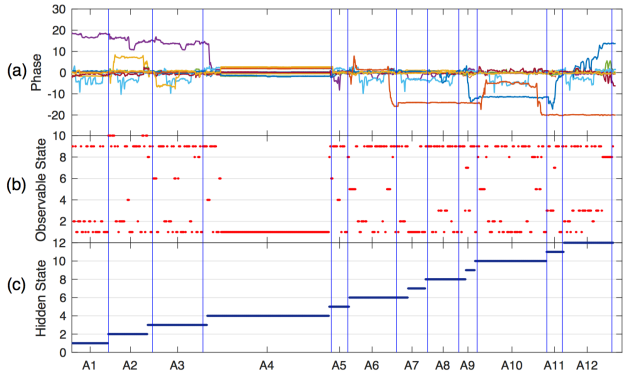


Figure 4: The collected phase profiles from tag1 to tag10, and the output of VQ module and HMM module

method[16]. The K-means algorithm[24] is an unsupervised clustering algorithm that can partition a series of feature vectors into k clusters in which each feature vector belongs to the cluster with the nearest mean. In this paper, we choose the K-means based optimization method to generate the codebook.

The VQ module consists of two parts: a codebook builder and a quantizing encoder. Given a set of feature vectors $\{F_1, F_2, \cdots, F_n\}$ as training data, where each vector is a m-dimensional feature vector of one sliding window, the codebook builder leverage the K-means algorithm to partition the n feature vectors into k ($k \leq n$) clusters. Each cluster is represented by its cluster index and centroid vector. So the indexes of each cluster are used as elements in the codebook, that means $cb = \{1, 2, \cdots, k\}$. After that, the quantization encoder can map a feature vector to a cluster index in the codebook $F_j \rightarrow O_j \in cb$, which is closest to this feature vector, according to the Euclidean distance between the feature vector and all the centroid vectors. Thus, the O_j is the output of VQ module which can be provided to HMM module as observable state. Figure 4(b) illustrates the output of VQ module when we encode the feature vectors of the phase profile in Figure 4(a) into 10 clusters (observable states) for each sliding window.

3.5 HMM Module

A hidden Markov model (HMM) is a popular tool for modeling time series data that is widely used in the field of speech recognition[1, 3, 12]. HMM belongs to generative probabilistic model that consists of an observable state and a hidden state at each sliding time window. In this paper, the observable state O_j in the j_{th} time window is

obtained from the VQ module and the hidden state is the activity (step) labels in the workflow as shown in Table 1, denoted by $H_j \in \{A1, A2, \cdots, A12\}$. Given a sequence of observable states, we cannot directly infer the activity sequence that produced these observations, since multiple activities may generate similar observed states, for example, the step1 (remove the tissue culture media) and the step7 (spray cells with pipette) involve the same items (pipette and petri dish A), so a similar phase profile may be generated and eventually the same observed state may be obtained. Thus the activity cannot be recognized by noticing the observable states in isolation. HMM can take into account the observable states in the sequence and provide a context-aware activity recognition method.

The goal of the HMM module is to infer most probable activity series (hidden state sequence of HMM) from the observation series, which is a labeling problem of single-layer HMM[33]. The hidden Markov model relies on the assumption that the hidden state (step) at time t depends only on the previous hidden state at time $t-1$, and the current observable state depend only on the current hidden state. With this assumption, we can specify an HMM using a triplet consisting of $\lambda = (\pi, T, E)$, where π is the vector of initial state probabilities, representing the probability with which a step occurs at the beginning of the hidden state sequence, T is the matrix of the state transition probability between two steps, and E is the matrix of emission/observation probability, representing the probability of a particular step generating a specific associated observable state. Given the observable state sequence O and the hidden state sequence H, the hidden Markov model is in fact a probability model with the hidden variable denoted by

$$P(O|\lambda) = \sum_{H} P(O|H, \lambda)P(H|\lambda) \tag{10}$$

This triplet of HMM parameters λ can be trained by Baum-Welch algorithm[33] when a subject performs the activity series of the complete workflow. Given a set of observable state sequences, the Baum-Welch algorithm leverages the classical EM algorithm[7] to find the maximum likelihood estimate of the parameters $\lambda^* = argmax_\theta P(O|\lambda)$. After training the model parameters, if the VQ module produces the observation sequence $\langle O_1, O_2, \cdots O_T \rangle$, we can utilize Viterbi algorithm[33] to generate the most likely sequence of workflow activities $\langle H_1, H_2, \cdots H_T \rangle$ in real time based on the estimated triplet of HMM parameters λ^*. Figure 4(c) depicts the estimated hidden states using Viterbi algorithm based on the sequence of observable states in Figure 4(b), and the horizontal axis represents the actual period of each step, separated by the blue vertical lines.

3.6 SA Module

The sequence analysis (SA) module is responsible for counting the duration of each activity of workflow from the hidden state sequence $\langle H_1, H_2, \cdots H_T \rangle$. The duration of one activity A_i can be calculated by multiplying the number of consecutive occurrences of A_i in a hidden state sequence by the duration of a sliding window. The SA Module can be designed for different applications, for instance, to detect whether there are any key steps to be missed, or to detect which step is being performed now and to assess the proficiency of the beginner. In addition, the SA module can be used to reduce the recognition error of the previous module, by detecting and removing outliers in the hidden state sequence. In statistics, an outlier is the individual value that is significantly deviated from other values, which can be detect by Grubbs' test[9]. For example, if the activity's average execution time is one minute, but the sequence shows that an activity lasts only one second, it is most likely that there was an recognition error in the previous modules, which could be avoided by the SA module.

4 IMPLEMENTATION & EVALUATION

In this section, the implementation of RFlow-ID is described and the evaluation results are presented. The recognition accuracy of workflow activity will be verified by experiments. Then, we evaluate some applications of RFlow-ID workflow recognition. At last, we discuss the factors affecting the recognition performance of RFlow-ID.

4.1 Implementation

Hardware We implement RFlow-ID using off-the-shelf commercial RFID devices without any hardware modification. We employ an Impinj Speedway R420 reader equipped with a directional antenna (Laird S9028PCR with 9 dBi gain). The reader is compatible with EPC Gen2 standard, and operates at a fixed working frequency of 920.675MHz. To ensure a high sampling rate of phase data, the reader mode is set to "Max Throughput", which supports the highest data rate, and the search mode is set to "dual target"[14]. The size of the reader antenna is $259mm \times 259mm \times 33.5mm$. The type of the tags is ALN-9710 with a size of $94.8mm \times 8.1mm$.

Software We adopt the Low Level Reader Protocol (LLRP) to support the communication between the reader and the tags. The Impinj Reader provides the API for RF phase reporting via an extension of the EPC LLRP. We implement the software using C# for top-level module connection and phase collector module, and Matlab for other modules like feature extractor, VQ module, etc..

4.2 Recognition Accuracy of Workflow Activity

The first experiment for evaluating the performance of RFlow-ID is to ask a participant to perform a complete workflow step by step from A1 to A12 for 20 times. The RFlow-ID acquires the phase data from the phase collector during each execution of workflow, and provides the inferred sequence of activity labels by the HMM module. For evaluating the inferred sequence, the ground truth is an actual sequence of activity labels gathered by manually recording the duration of each activity, and annotating each time window of the collected phase profile with actual activity label.

We perform 10-fold cross validation on the participant's phase profile dataset and generate an average confusion matrix. Figure 5(a) shows the confusion matrix. Each row of the matrix represents the actual activity label and the number of corresponding data points which are normalized, while each column of the matrix represents the inferred activity label by the RFlow-ID and the normalized number of corresponding data points. In contrast, Figure 5(b) shows the confusion matrix of accelerometer based method. A wrist 3D acceleration sensing unit is used to capture motion data from the participant's dominant hand with a sampling rate of 50Hz. The data of the X, Y and Z triaxial accelerometers collected during the execution of the workflow are also processed by the feature

extractor, the VQ module and the HMM module, and the sequence of activity labels can also be inferred.

Recognition accuracy is the most common metric for evaluating recognition performance which is calculated as the ratio of correctly-classified data points to total data points, this means the accuracy can be calculated from the confusion matrix by summing the values along the diagonal and dividing by the sum of all of the values in the matrix. Referring to Figure 5(a) and 5(b), the recognition accuracy of RFlow-ID is around 90.4%, while the recognition accuracy of accelerometer based method is around 75.8%. Furthermore, precision, recall and F1-scores can be calculated based on the confusion matrix of each method, which are listed in Table 2. It is visible that the RFlow-ID method yields better recognition performance due to higher precision, recall and F1-scores than accelerometer based method.

Table 2: Precision/Recall/F1-scores of RFlow-ID method and accelerometer based method

	RFlow-ID method			Accelerometer based method		
	Precision	Recall	F1-score	Precision	Recall	F1-score
A1	1.00	0.97	0.98	0.97	0.91	0.94
A2	0.96	0.90	0.93	0.75	0.65	0.70
A3	0.86	0.93	0.89	0.65	0.88	0.75
A4	0.97	0.90	0.94	1.00	0.83	0.91
A5	0.92	0.94	0.93	0.90	0.77	0.83
A6	0.92	0.97	0.94	0.61	0.94	0.74
A7	0.86	0.93	0.89	0.91	0.34	0.50
A8	0.75	0.88	0.81	0.69	0.81	0.74
A9	0.98	0.67	0.80	0.77	0.86	0.81
A10	0.85	0.99	0.92	0.76	0.85	0.80
A11	0.96	0.85	0.90	0.93	0.31	0.46
A12	0.89	0.93	0.91	0.64	0.96	0.77
Mean	0.91	0.90	0.90	0.80	0.76	0.75

4.3 Applications of Workflow Recognition

In this experiment, we explore the effectiveness of RFlow-ID in specific applications of workflow recognition. Consider the case where a novice has just begun to learn to do an experiment which contains a lot of steps and operations, a crucial learning assistant means is to remind beginners what is the next step. The prerequisite for providing such a reminder is that the assistant system needs to know which step the beginner is doing now. In addition, the common mistake when a novice learns an operational process, is missing an experiment step. Therefore, we provide the following two applications as an example to verify RFlow-ID system's capabilities.

Monitoring the step being performed In order to recognize what the current step is in real time, we select a participant to perform the complete workflow for 20 times as a training set, and then we provide 12 different test sets, which are performed by the participant from A1 to A2, from A1 to A3 until execution from A1 to A12, where each test set contains 10 repetitive execution data. Figure 6 shows the results of test set identification using the confusion matrix. The vertical axis represents the step in which the participant actually performs, and the horizontal axis represents the estimated step using RFlow-ID. The identification accuracy can be calculated by summing the values along the diagonal and dividing by the sum of all of the values in the matrix, which is around 94.2%.

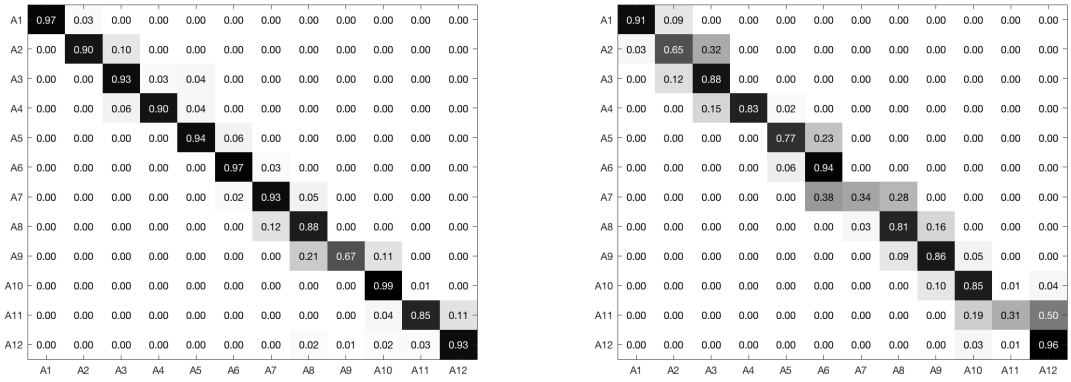
Detecting the missing step We use the same training set data as above, that is from 20 executions of complete workflow performed by one participant. In order to detect the missing step, the test set data is acquired from a participant performing the workflow from A1 to A12 but missing one of the steps. Thus, there are 12 different test sets including missing A1, missing A2, and so on. Each test set contains 10 repetitive execution data. Figure 7 illustrates the identification accuracy of detecting the missing step. We can see that the identification accuracy of missing A9 is the lowest. This is consistent with the result of recognizing the complete workflow in Figure 5(a), that is, the feature extracted in A9 will have a certain probability of being identified as A8 or A10. Nonetheless, the identification accuracy of missing A9 is still 70%, and the average identification accuracy is 83.3% for all test sets.

4.4 Factors affecting Recognition Performance

In this section, we investigate the factors affecting recognition performance. We mainly focus on three factors: cross-participant, training set size, and group size. The cross-participant describes a situation which is often found in practice, that is, after we have trained the system model, a person who is not in the training set is used as a test sample, so the recognition performance may be affected due to the individual diversity. The training set size refers to the number of samples used to train the model, and the group size refers to the number of participants used in the training model. We apply cross-participant validation for evaluating the impact of cross-participant, which is achieved with a Leave-One-Participant-Out validation. For instance, we consider the following experimental scenario: there are 5 participants, we collect the data from one of those participants to perform the complete workflow 10 times as a test set, and the data of the other 4 participants to perform the complete workflow 20 times in total as a training set, where each participant provides 5 training samples, so this cross-participant validation is executed with the training set size of 20 and the group size of 4.

Cross-participant Figure 8 depicts the performance impact of cross-participant validation when the training set size is 20 and the group size is 1, where the per-participant means that the data for the training set and the test set are from the same participant, so the group size is 0 and the training set size is also 20. The results show the average F1-score for each step of per-participant validation is 0.90, while the average F1-score for each step of per-participant validation is 0.74. On average, F1-Score of the cross-participant is 18% worse than per-participant, which is most likely caused by that different participants performing the workflow in a slightly different fashion.

Training set size Figure 9 indicates the impact of training set size on the recognition performance. The training set size increases from 10 to 50 for both cross-participant validation with group size of 1 and per-participant validation. We can find that increasing the training set size does not improve the recognition performance of per-participant validation, but cause a slight decline of around 3%. For cross-participant validation, the most improved F1-score is around 10% from training set size of 10 to 20, but there is no significant change after the training set size increased to 20. Thus,



(a) Confusion matrix of RFlow-ID

(b) Confusion matrix of accelerometer based method

Figure 5: Confusion Matrices for analyzing the Recognition Performance

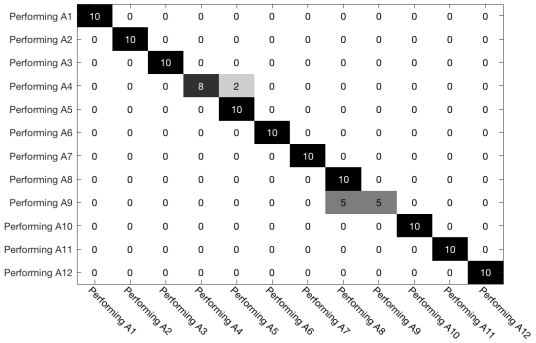


Figure 6: The confusion matrix of monitoring the step being performed

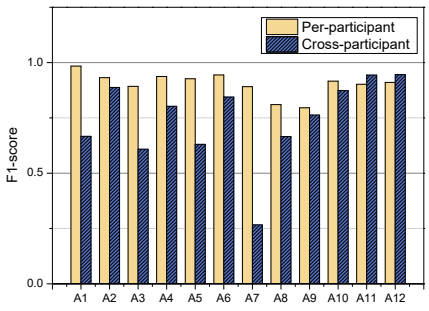


Figure 8: The impact of cross-participant

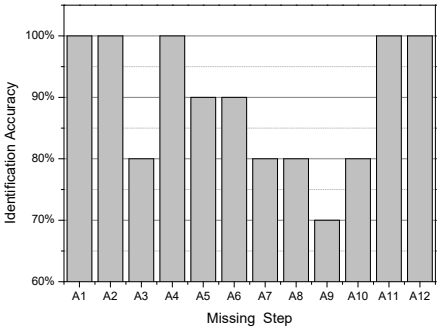


Figure 7: The identification accuracy of detecting the missing step

the impact of the training set size on the recognition performance is relatively small.

Group size Figure 10 shows the impact of group size on the cross-participant validation when the training set size is 20. It shows that the performance improved by about 5% from group size of 1 to 2, and increased by about 15% from group size of 2 to 3, but decreased by about 5% from group size of 3 to 4. During the analysis, it has been concluded that the accuracy increases with the group size (from 1 to 3) in the beginning since a larger group contains more individual diversity. But then the system performance degrades when the group size exceeds 3, this is because the training set size in the experiment do not increase with the increase in group size, which leads to the phenomenon of under-fitting. Therefore, increasing the group size while also increasing the training set size is conducive to improving the robustness of the system.

5 RELATED WORK

In this section, we review the related literature in RFID-based sensing and workflow recognition model.

5.1 RFID-based sensing

RFID is normally considered as the enabling tool for automatic identification of objects. However, recent research has shown that the physical RF signal between RFID readers and tags can be a powerful sensing modality for many applications and much attention has

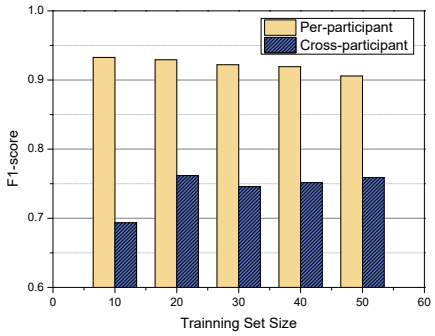


Figure 9: The impact of training set size

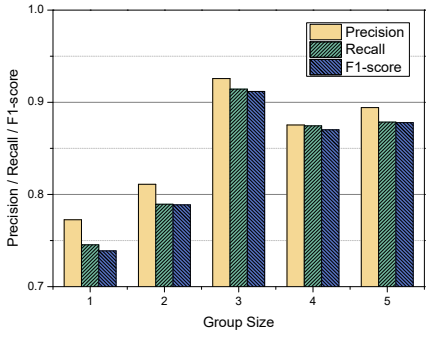


Figure 10: The impact of group size

been paid on activity recognition, like fall detection[43], shopping behavior mining[36] and exercise monitoring[8]. Authors in[15] enable real-time fluid intake monitoring by a well-designed tag attached smart cup, which can be used to ensure adequate hydration in older people. Researchers further explore the feasibility of RFID-based human-computer interaction. Authors in[22]present a battery-free device called Tagball for 3D human-computer interaction by tracing the motion states of a group of tags. PaperID[20] uses sensing and signal processing techniques to turn RFID tags into simple paper input devices and further creates a wide variety of interaction sensing types. Most of the state-of-the-art RFID researches aim to recognize isolated actions or activities (e.g. walking, eating, lying down, etc.). RFlow-ID differs from the researches in that we focus on workflow recognition which has a temporal relationship between activities, and we leverage the operation-induced low-level RF signal phase information to recognize the workflow without wearing RFID tags or readers.

5.2 Workflow recognition model

In order to establish a workflow recognition model, the most prevalent methods are designed based on HMM. Oliver et al. [28] present layered-HMMs model for event recognition in meetings based on the data of multi-sensor fusion. The layered-HMMs model is used to diagnose states of a user's activity based on real-time streams of video, audio, and computer interactions. The layered-HMMs model contains different levels of data granularity, where each layer is connected to the next layer via its inferential results, and the event classification using multi-modal information is applied to this model. However, with the increase in the number of layers, the computational overhead will be greatly increased, so this method is not suitable for real-time monitoring.

Philipp et al. [35] recognize the workflow using kNN-HMM model based on the data of 3D accelerometer. This model consists of two parts: action detection and protocol step detection. The action detection first leverages kNN to recognize the basic actions such as pestling and pipetting, then the protocol step detection uses HMM to recognize the steps of workflow, where hidden states refer to steps in the experiment protocol, and observable states map to the actions involved in that steps. However, in this model, the recognition performance of the protocol step detection part is greatly influenced by the action detection part. The results shows that the mean F1-score of the action detection part is around 70%, and the protocol step detection part is only 56%.

Compared to existing modeling approaches, RFlow-ID applies VQ-HMM for modeling the workflow recognition system based on the data of RFID phase profiles. We use VQ method combined with single-layer HMM as much as possible to reduce the computational overhead of the model, while the system can achieve high recognition performance.

6 CONCLUSION

This paper proposes a workflow recognition system based on COTS RFID, called RFlow-ID. In order to recognize each step in a workflow, the RFlow-ID leverages the phase information from low-level RF signal to perceive the use and movement of associated objects in the workflow, and uses VQ-HMM model to infer the most likely sequence of workflow activities. The system is implemented on COTS RFID devices, and extensive experimental evaluation under various conditions validates the high recognition accuracy. Experimental results show that RFlow-ID can achieve 90.4% recognition accuracy in recognizing each activity of the workflow. Besides, RFlow-ID has around 94.2% identification accuracy for monitoring the step being performed, and around 83.3% for detecting the missing step. We envision that our system can be an enabling tool for many personalized services such as learning assistant and skills assessment.

ACKNOWLEDGMENT

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

REFERENCES

[1] Ossama Abdel-Hamid and Hui Jiang. 2013. Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. 7942–7946.

[2] AHMED A. ABDELWAHAB and NORA S. MUHARRAM. 2011. A FAST CODEBOOK DESIGN ALGORITHM BASED ON A FUZZY CLUSTERING METHODOLOGY. *International Journal of Image & Graphics* 7, 02 (2011), –.

[3] Alex Acero, Li Deng, Trausti Kristjansson, and Jerry Zhang. 2000. HMM ADAPTATION USING VECTOR TAYLOR SERIES FOR NOISY SPEECH RECOGNITION. In *The Proceedings of the*. 869–872.

[4] Ardhendu Behera, Anthony G. Cohn, and David C. Hogg. 2012. *Workflow Activity Monitoring Using Dynamics of Pair-Wise Qualitative Spatial Relations.* Springer Berlin Heidelberg. 196–209 pages.

[5] D Burton, J Shore, and J Buck. 1983. A generalization of isolated word recognition using vector quantization. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP*. 1021–1024.

[6] C. C. Chang and W. C. Wu. 2007. Fast planar-oriented ripple search algorithm for hyperspace VQ codebook. *IEEE Trans Image Process* 16, 6 (2007), 1538–1547.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society* 39, 1 (1977), 1–38.

[8] Han Ding, Longfei Shangguan, Zheng Yang, Jinsong Han, Zimu Zhou, Panlong Yang, Wei Xi, and Jizhong Zhao. 2015. Femo: A platform for free-weight exercise monitoring with rfids. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 141–154. https://doi.org/10.1145/2809695.2809708

[9] Frank E. Grubbs. 1950. Sample Criteria for Testing Outlying Observations. *Annals of Mathematical Statistics* 21, 1 (1950), 27–58.

[10] Tian Hao, Guoliang Xing, and Gang Zhou. 2013. iSleep:unobtrusive sleep quality monitoring using smartphones. In *ACM Conference on Embedded Networked Sensor Systems*. 1–14.

[11] Peter Hevesi, Sebastian Wille, Gerald Pirkl, Norbert Wehn, and Paul Lukowicz. 2014. Monitoring household activities and user location with a cheap, unobtrusive thermal sensor array. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 141–145.

[12] X. D Huang, Y Ariki, and Mervyn A Jack. 1989. Hidden Markov Models for Speech Recognition. 33, 3 (1989), 251–272.

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

[14] Impinj. 2017. Optimizing Tag Throughput Using ReaderMode. https://support.impinj.com/hc/en-us/articles/202756368-Optimizing-Tag-Throughput-Using-ReaderMode. (2017).

[15] Asangi Jayatilaka and Damith C Ranasinghe. 2016. Real-time fluid intake gesture recognition based on batteryless UHF RFID technology. *Pervasive and Mobile Computing* (2016).

[16] H. B. Kekre and Ms. Tanuja K. Sarode. 2009. Vector Quantized Codebook Optimization using K-Means. *International Journal on Computer Science & Engineering* 1, 3 (2009), 283–290.

[17] Aftab Khan, Sebastian Mellor, Eugen Berlin, Robin Thompson, Roisin Mcnaney, Patrick Olivier, and Thomas Tz. 2015. Beyond activity recognition: skill assessment from accelerometer data. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 1155–1166.

[18] Orr Robert Abowd Gregory D Atkeson Christopher G Essa Irfan A Macintyre Blair Mynatt Elizabeth D Starner Thad Kidd, Cory D and Wendy Newstetter. 2002. The Aware Home: A Living Laboratory for Ubiquitous Computing Research. In *International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*. 191–198.

[19] Chandrashekhar Lavania, Sunil Thulasidasan, Anthony Lamarca, Jeffrey Scofield, and Jeff Bilmes. 2016. A weakly supervised activity recognition framework for real-time synthetic biology laboratory assistance. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 37–48.

[20] Hanchuan Li, Eric Brockmeyer, Elizabeth J Carter, Josh Fromm, Scott E Hudson, Shwetak N Patel, and Alanson Sample. 2016. PaperID: A Technique for Drawing Functional Battery-Free Wireless Interfaces on Paper. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 5885–5896. https://doi.org/10.1145/2858036.2858249

[21] H. Y. Liao, Duan Yu Chen, Chih Wen Sua, and Hsiao Rang Tyan. 2006. Real-time event detection and its application to surveillance systems. In *IEEE International Symposium on Circuits and Systems*. 4 pp.–512.

[22] Qiongzheng Lin, Lei Yang, Yuxin Sun, Tianci Liu, Xiang-Yang Li, and Yunhao Liu. 2015. Beyond one-dollar mouse: A battery-free device for 3d human-computer interaction via rfid tags. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1661–1669. https://doi.org/10.1109/INFOCOM.2015.7218546

[23] Tianci Liu, Lei Yang, Xiang-Yang Li, Huaiyi Huang, and Yunhao Liu. 2015. TagBooth: Deep shopping data acquisition powered by RFID tags. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1670–1678. https://doi.org/10.1109/INFOCOM.2015.7218547

[24] J Macqueen. 1967. Some Methods for Classification and Analysis of MultiVariate Observations. In *Proc. of Berkeley Symposium on Mathematical Statistics and Probability*. 281–297.

[25] Futoshi Naya, Ohmura Ren, Fusako Takayanagi, Haruo Noma, and Kiyoshi Kogure. 2006. Workers' Routine Activity Recognition using Body Movements and Location Information. In *IEEE International Symposium on Wearable Computers*. 105–108.

[26] Pavel V Nikitin, Rene Martinez, Shashi Ramamurthy, Hunter Leland, Gary Spiess, and KVS Rao. 2010. Phase based spatial identification of UHF RFID tags. In *RFID, 2010 IEEE International Conference on*. IEEE, 102–109. https://doi.org/10.1109/RFID.2010.5467253

[27] C Occhiuzzi and G Marrocco. 2010. The RFID technology for neurosciences: feasibility of limbs' monitoring in sleep diseases. *IEEE Transactions on Information Technology in Biomedicine A Publication of the IEEE Engineering in Medicine and Biology Society* 14, 1 (2010), 37–43.

[28] Nuria Oliver, Ashutosh Garg, and Eric Horvitz. 2004. Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision & Image Understanding* 96, 2 (2004), 163–180.

[29] Nicolas Padoy, Tobias Blum, Seyed Ahmad Ahmadi, Hubertus Feussner, Marie Odile Berger, and Nassir Navab. 2012. Statistical modeling and recognition of surgical workflow. *Medical Image Analysis* 16, 3 (2012), 632.

[30] N. Padoy, D. Mateus, D. Weinland, and M O. Berger. 2009. Workflow monitoring based on 3D motion features. In *IEEE International Conference on Computer Vision Workshops*. 585–592.

[31] Stoyanov Danail Mcilwraith Douglas Lo Benny P. L. Pansiot, Julien and G. Z. Yang. 2007. Ambient and Wearable Sensor Fusion for Activity Recognition in Healthcare Monitoring Systems. In *International Workshop on Wearable and Implantable Body Sensor Networks, Bsn 2007, March 26-28, 2007, Rwth Aachen University, Germany*. 208–212.

[32] Hanchuan Peng, Fuhui Long, and Chris Ding. 2005. *Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy*. IEEE Computer Society. 1226 pages.

[33] L. R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* 77, 2 (Feb 1989), 257–286. https://doi.org/10.1109/5.18626

[34] K Sayood, J. D Gibson, and Martin C Rost. 1984. An algorithm for uniform vector quantizer design. *IEEE Transactions on Information Theory* 30, 6 (1984), 805–814.

[35] Philipp M Scholl, Matthias Wille, and Kristof Van Laerhoven. 2015. Wearables in the wet lab: a laboratory system for capturing and guiding experiments. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 589–599.

[36] Longfei Shangguan, Zimu Zhou, Xiaolong Zheng, Lei Yang, Yunhao Liu, and Jinsong Han. 2015. Shopminer: Mining customer shopping behavior in physical clothing stores with cots rfid devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 113–125. https://doi.org/10.1145/2809695.2809710

[37] Yan Tang, Shuangquan Wang, Yiqiang Chen, and Zhenyu Chen. 2012. PPCare: A Personal and Pervasive Health Care System for the Elderly. In *International Conference on Ubiquitous Intelligence, Computing and nternational Conference on Autonomic Trusted Computing*. 935–939.

[38] Yoshihiro Uemura, Jianlong Zhou, Yusuke Kajiwara, Fang Chen, and Hiromitsu Shimakawa. 2015. Estimating Human Physical States from Chronological Gait Features Acquired with RFID Technology. (2015).

[39] Athanasios Voulodimos, Dimitrios Kosmopoulos, Georgios Vasileiou, Emmanuel Sardis, Vasileios Anagnostopoulos, Constantinos Lalos, Anastasios Doulamis, and Theodora Varvarigou. 2012. A Threefold Dataset for Activity and Workflow Recognition in Complex Industrial Environments. *IEEE Multimedia* 19, 3 (2012), 42–52.

[40] Athanasios S. Voulodimos, Dimitrios I. Kosmopoulos, Nikolaos D. Doulamis, and Theodora A. Varvarigou. 2014. A top-down event-driven approach for concurrent activity recognition. *Multimedia Tools and Applications* 69, 2 (2014), 293–311.

[41] Jue Wang and Dina Katabi. 2013. Dude, where's my card?:RFID positioning that works with multipath and non-line of sight. In *Acm Sigcomm Conference on Sigcomm*. 51–62.

[42] Jamie A. Ward, Paul Lukowicz, Gerhard Troster, and Thad E. Starner. 2006. Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 10 (2006), 1553.

[43] Asanga Wickramasinghe, Roberto Luis Shinmoto Torres, and Damith C Ranasinghe. 2016. Recognition of falls using dense sensing in an ambient assisted living environment. *Pervasive and mobile computing* (2016). https://doi.org/10.1016/j.pmcj.2016.06.004

[44] A Wood, J Stankovic, G Virone, L Selavo, Zhimin He, Qiuhua Cao, Thao Doan, Yafeng Wu, Lei Fang, and R Stoleru. 2008. Context-aware wireless sensor networks for assisted living and residential monitoring. *IEEE Network the Magazine of Global Internetworking* 22, 4 (2008), 26–33.

[45] Jianxin Wu, Adebola Osuntogun, Tanzeem Choudhury, Matthai Philipose, and James M. Rehg. 2007. A Scalable Approach to Activity Recognition based on Object Use. In *IEEE International Conference on Computer Vision*. 1–8.

[46] Lei Yang, Yekui Chen, Xiang-Yang Li, Chaowei Xiao, Mo Li, and Yunhao Liu. 2014. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 237–248. https://doi.org/10.1145/2639108.2639111

[47] Yongpan Zou, Jiang Xiao, Jinsong Han, Kaishun Wu, Yun Li, and Lionel M Ni. 2016. GRfid: A Device-free Gesture Recognition System Using COTS RFID Device. 16, 2 (2016), 381–393. https://doi.org/10.1109/TMC.2016.2549518