Optimal time window for temporal segmentation of sensor streams in multi-activity recognition

Jonathan Liono
School of Science
Computer Science and
Information Technology
RMIT University
Melbourne, Australia
jonathan.liono@rmit.edu.au

A. K. Qin
School of Science
Computer Science and
Information Technology
RMIT University
Melbourne, Australia
kai.qin@rmit.edu.au

Flora D. Salim
School of Science
Computer Science and
Information Technology
RMIT University
Melbourne, Australia
flora.salim@rmit.edu.au

ABSTRACT

Multi-activity recognition in the urban environment is a challenging task. This is largely attributed to the influence of urban dynamics, the variety of the label sets, and the heterogeneous nature of sensor data that arrive irregularly and at different rates. One of the first tasks in multi-activity recognition is temporal segmentation. A common temporal segmentation method is the sliding window approach with a fixed window size, which is widely used for single activity recognition. In order to recognise multiple activities from heterogeneous sensor streams, we propose a new time windowing technique that can optimally extract segments with multiple activity labels. The mixture of activity labels causes the impurity in the corresponding temporal segment. Hence, larger window size imposes higher impurity in temporal segments while increasing class separability. In addition, the combination of labels from multiple activity label sets (i.e. number of unique multi-activity) may decrease as impurity increases. Naturally, these factors will affect the performance of classification task. In our proposed technique, the optimal window size is found by gaining the balance between minimising impurity and maximising class separability in temporal segments. As a result, it accelerates the learning process for recognising multiple activities (such as higher level and atomic human activities under different environment contexts) in comparison to laborious tasks of sensitivity analysis. The evaluation was validated by experiments on a real-world dataset for recognising multiple human activities in a smart environment.

CCS Concepts

•Human-centered computing → Ubiquitous and mobile computing;

Keywords

temporal segmentation; multivariate sensor streams; multi-activity recognition; data stream processing; multi-objective function; optimal window size

MOBIQUITOUS '16, November 28-December 01, 2016, Hiroshima, Japan

ISBN 978-1-4503-4750-1/16/11...$15.00

DOI: http://dx.doi.org/10.1145/2994374.2994388

1. INTRODUCTION

The advances in ubiquitous computing in recent years have driven the research to automatically observe, monitor and recognise contextual information for various benefits, which are enabled by direct applications of machine learning algorithms. As an example, resident monitoring for smart home applications can facilitate tracking and recognition of daily activities for Alzheimer's patients and elderly care. Several initiatives such as [19, 8, 18, 22, 7] have brought various researchers to achieve significant contributions for activity recognition in smart environments.

Human annotated data may not be aligned consistently in terms of the boundary between activities. In other words, the impurity of the associated activity labels within a temporal segment may affect the performance quality of classification tasks for activity recognition. It should be noted that throughout the course of this paper, labels and annotations are used interchangeably in the context of activity recognition corresponding to class labels.

On the other hand, it is crucial to consider a more dynamic and scalable use case for activity recognition. For example, multiple activities that can be observed in real-world situation from streaming of ubiquitous sensor data. However, this is not well explored in the current literature. Existing works mainly focus on single activity recognition from one label set. In common practice, the techniques associated to temporal segmentation involve optimised processes for one-dimensional space of an activity label set. It is a standard mechanism before recognition process (i.e. classification of activities). However, these techniques may need to be adjusted for multi-activity recognition scenario.

For many years, major works have been performed in order to understand the human behavior and mobility patterns in various interdisciplinary studies. Previous work by González et al. [11] had shown the individual mobility patterns from trajectories of 100,000 mobile phone users in order to signify that each individual is characterized by time-independent travel distance and the probability of return to frequently visited locations. As the results, they could obtain the likelihood of a mobile phone user in any location. Given the opportunities by ubiquitous computing capabilities, the application scenarios of activity recognition in unconstrained environment could be limitless.

To illustrate the importance such recognition in a smart environment, possible scenarios that can be applied in the medical sector include automatic recognition of user (staff) activities in smart hospital [32] and emotion recognition for measuring mental fitness [34, 28]. Therefore, the inferences of multi-activity recognition from ubiquitous sensors can be leveraged to facilitate human mobility and contextual modeling in these dynamic smart environments.

These directions suggest the tremendous needs for innovations and efficient methods to handle streaming of ubiquitous, yet irregular multivariate sensor data. Hence, it is a significant challenge to improve the performance of multi-activity recognition. This is due to the fact that human activity in a smart environment is bound to be dynamic. In addition, dynamic changing of activities in data streams will incur a mixture of these activity labels during temporal segmentation process. Therefore, methodologies in minimising the purity in temporal segments are needed without sacrificing much on the performances of activity recognition. In this paper, a window refers to a temporal segment in sliding window model. Moreover, scalability of these methodologies is required to fit into multi-activity recognition scenario.

Hence, the contribution of this paper focuses on the preprocessing of ubiquitous sensor streams for activity recognition tasks. In particular, our proposed technique provides the following contributions:

1. The multi-objective technique of finding optimal window size for time-interval based temporal segmentation in streaming fashion. It is derived based on gaining the balance between minimising label impurity in a segment and maximising factor for class separability (divergence) of ubiquitous sensor features towards class labels.

2. Robust recommendation of optimal time-interval window sizes for temporal segmentation in multi-activity recognition.

2. RELATED WORKS

The study of human activity recognition is a well-known area in many research communities, including computer vision [37, 2]. As the proliferation of mobile and ubiquitous devices become prominent, activity recognition from streaming sensor data becomes an emerging research area. It is difficult to be neglected due to its apparent realisation in near future by enabling Internet of Things (IoT) technology. The inherent challenges of dealing with data streaming from these ubiquitous sensor devices (such as wearables, mobile devices and on-body sensors) are related to noisy sensor reading due to hardware limitation and environmental influences. Su et al [33] recently presented a general overview of techniques and challenges in performing activity recognition from the smartphone sensors. In the study of activity recognition, it is often treated as a classification problem since there are annotations associated with certain states (e.g. human locomotion activities) for training and testing phases. However, it is common practice to use one second window size for temporal segmentation in many experiments [27, 41, 38, 12]. Torres et al. [35] in their recent study verified that fixed time window (FTW) achieved high performance of real-time activity recognition. In this paper, we refer their FTW as time-interval based segmentation. They also have performed comprehensive performance evaluation of other segmentation techniques such as activity windowing, dynamic windowing and mutual information windowing.

In addition, Guo et al. [14] have proposed an adaptive approach for online segmentation through PCA feature selection and model selection. In this approach, the window size can expand with incorporating the next frame according to feature selection and model selection criteria. However, the mixture of labels during temporal segmentation is not included as a critical aspect of their approach. It is important to note that human activities are dynamic and subject to continuous change in data streams. Therefore, we consider that maintaining high label purity in a temporal segment is crucial for activity recognition application. However, it is insufficient to simply consider label purity alone. Another objective that can be considered in maintaining label purity of temporal segments is related to the importance of features for activity labels. In many image classification experiments such as [13], class separability is crucial to select important features in order to improve and accelerate classification tasks.

Therefore, the focus study of this paper is aimed towards finding the optimal time window size to mitigate from simple selection of window size from common heuristics (e.g. one second windowing). In the past literature, dynamic programming [3] and $k-$segmentation [17] can be used to perform time series segmentation for the purpose of context recognition. However, the drawback of these techniques is that it requires offline data processing for such context recognition. In real world application, streaming of sensor data can be irregular, especially when a sensor device is unavailable under certain circumstances. For example, GPS sensor of smartphone is unreliable when the user is inside a building, or in an underground tunnel. It is also not suitable for processing temporal data in streaming fashion, that can include event based sensor data. Krishnan and Cook [21] recently proposed their sliding window approach for performing activity recognition in a streaming fashion, which can be adapted for event based sensor streaming. Their method incorporates time decay and mutual information based weighting of sensor events within a window. This is due to the fact to their assumption that different activities can be characterised with different sensor lengths of sensor events. Particularly, their work is validated with real-world smart home dataset, which mostly consists of event based sensing data. However, this study is restricted towards one dimension of activity set. In many real world scenarios, multiple activities can be required for recognition task. This is commonly known as a multi-label problem.

Most of the recognition problems in past studies [6, 30, 16, 9, 20] addressed the methods of recognition from one label set. To provide meaningful contextual information, multi-label classification can be leveraged to facilitate such needs. The problem of multi-label classification is practical in real world scenarios as a subject can be associated with multiple annotations at a time. For example, a person "sitting" in a cafe while "drinking coffee" or "running" while "listening to music" from a smartphone. In the medical application, multi-label classification can be used for diagnosing diseases, such as diabetes and prostate cancer [36]. Furthermore, this can be leveraged for classifying several characteristics during real-time ECG (Electrocardiography) analysis of data streaming from on-body sensors. Moreover, the multi-label problem is not strictly limited to smart environment scenario. Consequently, it is also applicable for mobility modeling in a dynamic urban environment. For example, Read et al. [29] proposed multi-label oriented technique to preprocess sensing data for the purpose of producing labeled data that are reliable for human mobility modeling. Due to various benefits that can be attained for multi-label classification, the motivations of our studies are influenced to pursue multi-activity recognition from continuous streaming of sensor data. A very simplistic approach for solving multi-label classification is to construct a new label set that is composed of the possible combination of labels from predefined label sets.

Our paper focuses on the proposed step for finding reasonable parameter (window size) for time-interval based temporal segmentation that is robust for various scenarios such as multi-activity recognition. Apparently, we have not found any study or solution that is related to the problems that are defined in the following section.

3. PROBLEM DEFINITION

3.1 Impure Windows in Temporal Segments

In order to identify certain activity label on data streams from ubiquitous sensors, it is a common practice that the data points are annotated in a time-interval manner. In several scenarios, these sensor data could arrive in irregular manner at different point of time. Temporal segmentation is necessary in order to define the boundary of feature extraction process for recognition phase.

Let $S = \{S_1, S_2, S_3, S_4, ..., S_d\}$ be the d number of sensors and S_i $(1 \leq i \leq d)$ is a sensor identifer within the range of d sensor streams. Each arrival of the sensor reading instance I_{ji} can be associated to a unique timestamp t_j that is continuous in a sensor stream S_i.

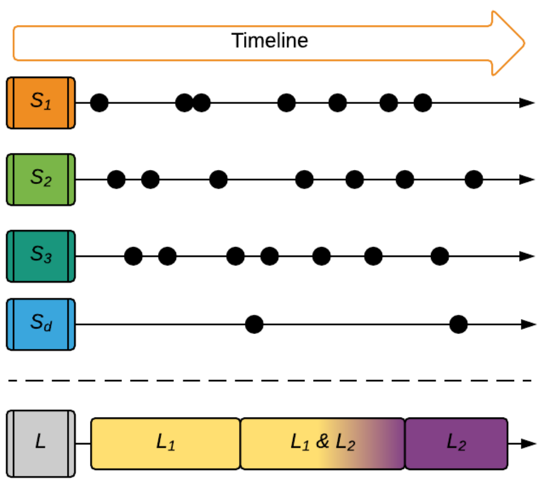

Figure 1: Non-overlapping temporal segmentation of sensor data streams with a given label set L. An impure segment is composed of more than one label.

Let us consider a temporal segmentation through sliding window technique where a data stream is processed in a continuous manner. It is apparent that temporal segmentation would produce several windows that are composed of mixtures of multiple annotations from a label set $L = \{L_1, L_2, ..., L_n\}$ (Figure 1). Inherently, the most dominant label would be selected as the final label for the corresponding window.

Assuming the segmentation is performed in time-interval based temporal segmentation with non-overlapping sliding window, the problem is formulated as how to find a reasonable time window given the impurity of segments in ubiquitous sensor streams. The complexity of the problem increases when these sensor streams need to be synthesised for segmentation purpose due to the variation of data arrival from heterogeneous sensors. Synthesis in this paper refers to synchronisation process to align the time segments of heterogeneous sensor streams in a format that is suitable for features extraction.

It is assumed that a complex streaming scenario involves different timestamp for the arrival of data from each sensor. A sensor may be inactive for a certain period of time given to certain predefined rules. For example, GPS sensor of smartphone user is activated only when the person is close to certain locations. When battery level of smartphone is low, the sampling frequency of sensor data may be reduced or scheduled in short time span in a controlled interval. Hence, a common approach such as frame based temporal segmentation would be inappropriate due to these irregular behaviours of sensor data streaming. Hence, the problem definition defined in this paper is constrained to time-interval based temporal segmentation for streaming ubiquitous sensor data.

A common practice is to assume a predefined fixed window size for segmentation process. For activity recognition on motion sensors (such as the accelerometer), one second window size is commonly assumed in many experiment settings [27, 41, 38, 12], including applications in the medical field such as monitoring for Parkinson's disease rehabilitation [5]. Another alternative method is to perform sensitivity analysis on various time-interval window size based on quality metrics of recognition results (e.g. accuracy of classifier validation results). In most cases, sensitivity analysis requires a significant amount of time to evaluate. In recognition phase such as classification tasks, another sensitivity analysis may need to be performed on each classifier for parameter tuning of machine learning algorithms.

3.2 Multi-label Problem of Temporal Segments

As described previously, real world applications may require multiple annotations to be associated with an instance. This is commonly addressed as multi-label problem. Assume that label vector L can be constructed with multiple label sets $L = \{L_1, L_2, ..., L_k\}$, which are derived from human annotations. Each label set L_k is composed of multiple labels $L_k = \{L_{k.1}, L_{k.2}, ..., L_{k.n}\}$. Therefore, it is clear that I_{ji} can be associated with annotations from each L_k label set as depicted in Figure 2.

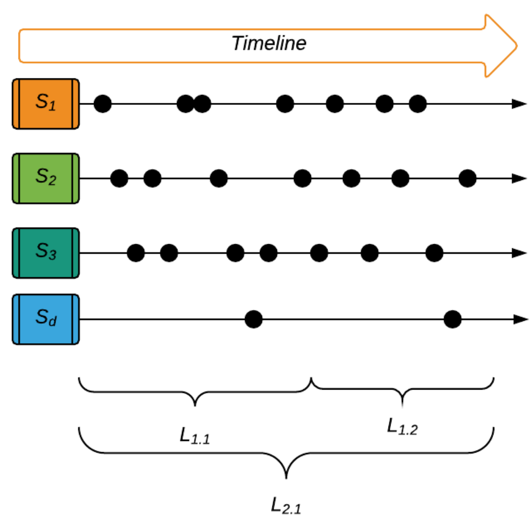

Figure 2: Sensor data streams (annotated with multiple labels).

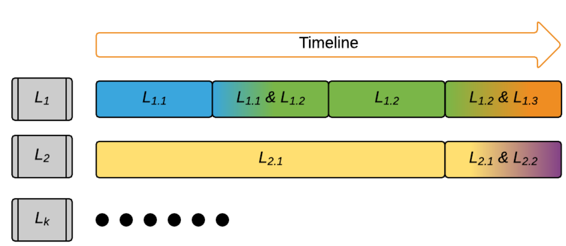

Figure 3: Impure segments in multi-label scenario.

Previously the problem of impure segments in temporal segmentation is explained. However, this problem is not limited to annotations from one label set. Hence, the impure segments can be scaled to several dimensions, given the annotations from heterogeneous label sets (as illustrated in Figure 3). Each label set may be associated with others or be independent according to certain contexts. Thus, finding a reasonable window size for temporal segmentation presents as a crucial problem in multidimensional scales.

Intuitively, it is feasible to find a balance for recommending optimal time-interval window size by minimising the impurity proportion of windows in temporal segmentation. Furthermore, the magnitude of complexity increases when sensor data streams are associated with annotations from multiple label sets. This is due to the assumption that each label set may have different optimal window size. For example, detecting a human activity from {standing, walking, sitting} may have smaller window size in comparison to recognising higher level activity from {cooking, exercising, relaxing on couch, eating} set. In addition, window size may be significantly different in the scenario where multiple activities must be predicted from given k label sets. In other words, the dynamic combination of items from heterogeneous label sets requires a balanced window size. Therefore, our contribution aims to improve and to accelerate part of processes in multi-activity recognition.

4. METHODOLOGY

Finding a reasonable window size for temporal segmentation requires a significant amount of time and effort from a typical sensitivity analysis of output quality of classifier results, especially in multi-activity scenarios. Hence, we present a data-driven approach of time-interval window size recommendation from annotated multivariate data streams. The objective of our algorithm is to achieve the balance between minimising impurity of segments in data streams and maximising factor for class separability based on given k number of label sets. Given the proposed multi-objective method of data-driven approach in finding optimal window size, we formalise the following metrics:

1. Impurity proportion of segments
2. Class separability

4.1 Impurity Proportion

In real world scenario, a window from temporal segmentation may contain a mixture of multiple annotations. Therefore, it is impractical to assume that a segment can contain only one annotation. We consider the portion of annotation mixture is important in the case of transition points between activities. In this paper, we define impure segments for the windows that contain more than one annotation from temporal segmentation.

The first objective of our method requires minimising the impurity proportion. Impurity proportion can be calculated via:

$$pimpure = \frac{m_{impure}}{m} \tag{1}$$

where m_{impure} is the count of impure segments over total m segments in data streams.

4.2 Class Separability

Class separability for data streams refers to the notion of representation for distinct separation between features in data streams with respect to each class label. Typically, the class separability score is calculated for each feature in a given data distribution. In this case, feature values of sensor streams are temporarily stored, which then be processed in order to identify which features are dominant towards a class label. In our method, Kullback-Leibler (KL) divergence [23] is used to calculate the class separability score. This is a common technique to measure the difference between two probability distributions P and Q.

It is assumed that the streaming of sensor data is constrained by numeric features. The overall values of each feature can be represented in histogram (frequency distribution) format. In other words, there would be one histogram per feature for each class. Hence, KL divergence can be calculated for each feature after the histogram representation has been attained from discretization using b equally-spaced bins. This histogram representation would fit into different scenarios where the numeric features are discrete or continuous. In other problem domain, class separability is used for feature selection [4, 13, 25, 26, 39, 40].

For each feature f, we calculate maximum score of the class separability (i.e. maximum divergence of f) as:

$$mdivergence_f = \max_{\substack{1 \le i \le n \\ 1 \le j \le n}} KL_f(i, j), \tag{2}$$

where n is number of class labels in a label set L_k and $KL_f(i, j)$ is the KL distance between two distributions (histograms) corresponding to class labels i and j:

$$KL_f(i, j) = \sum_{m=1}^{b} Pr_f(m \mid i) \, log\left(\frac{Pr_f(m \mid i)}{Pr_f(m \mid j)}\right) \tag{3}$$

where b is the number of bins in the histograms. Both probabilities $Pr_f(m \mid i)$ and $Pr_f(m \mid j)$ should never be 0; instead we define $0.0000001 \le Pr_f(m \mid l) \le 1$ heuristically where l is the class label index. In any case, a larger value of $mdivergence_f$ relates to greater separability of f over all labels in L_k. As KL divergence holds asymmetric property, it is practically useful to derive which top-k features provide significant contributions for a classifier by ranking class separability scores. However, this can be replaced with a symmetric solution such as Jensen-Shannon (JS) divergence [24] when a strict metric is required. Despite the distinction between symmetrical and asymmetrical divergence distances, it is prominent that the average of these pairwise measures is typically used for the extension to multi-class scenario [13]. In our case, the mean is derived from maximum divergences for all features. Maximum divergence is also used as scoring in [1] for feature selection.

4.3 Multi-objective Function

Based on the above two critical metrics, multi-objective function is leveraged that aims to gain balance between minimising the impurity proportion in data streams and maximising the divergence in terms of class separability of activity recognition. The balance is found by looking for optimal window size that has impurity proportion measurement close to the intersection of metrics within the defined window size search range. The details of multi-objective operation would be elaborated in the following algorithm section.

4.4 Algorithm

Our proposed method is formalised as Optimal Window (OPTWIN) for time-interval based temporal segmentation. In order to recommend time-interval window size for segmentation, it requires finding the balance between the above impure proportion and class separability measure.

As described in Algorithm 1, the recommendation of window size include the initial parameters of:

1. Three main inputs for the search range of window size:
   (a) Smallest window size ws_{start}.
   (b) Biggest window size as the upper boundary ws_{end}.
   (c) Time-interval step ws_{step} for window size.
2. An additional input for data streams from ubiquitous sensors $D_{streams}$.

Algorithm 1 OPTWIN based window size recommendation

procedure *recommendWindowSize*($ws_{start}, ws_{end}, ws_{step}, D_{streams}$)
  global $ws_{current} \leftarrow ws_{start}$
  while $ws_{current} \le ws_{end}$ **do** ▷ Label purity and divergence derivation
    for each $L_k \in L$ **do**
      $D_{segments} \leftarrow temporalSegmentation(ws_{current}, L_k)$
      $pimpure_{L_k}[ws_{current}] \leftarrow impurityProportion(D_{segments})$
      $multiActivities \leftarrow isMultiActivities(L_k)$
      if *multiActivities* **then**
        $discore_{L_k}[ws_{current}] = \frac{1}{i}\sum_{i=1}^{n} discore_{L_i}[ws_{current}]$
      else
        for each f **in** $D_{segments}$ **do**
          $maxdivergences_{L_k}[f] \leftarrow maxKL(values_f, L_k)$
        end for
        $discore_{L_k}[ws_{current}] = \frac{1}{f}\sum_{i=1}^{f} maxdivergences_{L_k}$
      end if
    end for
    $ws_{current} \leftarrow ws_{current} + ws_{step}$
  end while

  for each $L_k \in L$ **do** ▷ Optimal window size recommendation
    $ws_{current} \leftarrow ws_{start}$
    while $ws_{current} \le (ws_{end}$ **do**
      $dideviation_{current} \leftarrow nrmlDvt(discore_{L_k}[ws_{current}], discore_{L_k})$
      $metrics_{current} \leftarrow (pimpure_{L_k}[ws_{current}],\ dideviation_{current})$
      $ws_{next} \leftarrow ws_{current} + ws_{step}$
      $dideviation_{next} \leftarrow nrmlDvt(discore_{L_k}[ws_{next}], discore_{L_k})$
      $metrics_{next} \leftarrow (pimpure_{L_k}[ws_{next}],\ dideviation_{next})$
      $ws_{optimal} \leftarrow intersect(metrics_{current},\ metrics_{next})$
      if $ws_{optimal}$ **then**
        $ws_{rec}[L_k] \leftarrow ws_{optimal}$
        break;
      end if
      $ws_{current} \leftarrow ws_{current} + ws_{step}$
    end while
  end for

  return ws_{rec}
end procedure

This algorithm requires scanning operation for impurity proportion and class separability measure of every window size (with incremental of ws_{step}) within the boundary of ws_{start} and ws_{end}. The computation of impurity proportion $pimpure_{L_k}$ and divergence score from all maximum divergence of features $discore_{L_k}$ are stored for each label set L_k. In this case, the average function of all maximum divergences of features is used to compute $discore_{L_k}$.

The process of window size recommendation is composed of two stages. Firstly, the necessary metrics (e.g. impurity proportion and divergence score) are computed and stored for each L_k from a given L label sets in data streams $D_{streams}$. This stage is referred as label purity and divergence derivation in Algorithm 1. Within iteration of every possible window size, the derivation begins with temporal segmentation procedure where time-interval segmentation is performed on the given $D_{streams}$. In temporal segmentation procedure, feature extraction is performed to derive the summary of data points in each time window. It should be noted that the results of feature extraction would be later used in calculating the class separability score. Afterwards, the impurity proportion of each window size $pimpure_{L_k}[ws_{current}]$ is calculated through Equation 1. The divergence score $discore_{L_k}[ws_{current}]$ is then calculated subsequently through the mean maximum divergence from $maxdivergences_{L_k}[f]$ of all f features. The corresponding $maxdivergences_{L_k}[f]$ is computed through Equation 2. All of these metrics are stored temporarily for optimal window size recommendation stage.

Furthermore, for any L_k that is defined as multi-activity (i.e. L_k is composed by combination of several other L_k in L label sets), mean maximum divergence can be computed from the following:

$$discore_{L_k}[ws_{current}] = \frac{1}{i}\sum_{i=1}^{n} discore_{L_i}[ws_{current}] \tag{4}$$

where L_k is identified as multi-activity label set and L_i is single-activity label set that is part of n label sets being associated with composition of corresponding L_k. For example, let us consider a multi-activity scenario where L_k is composed of combination between locomotion activity L_1 label set and gesture activity L_2 label set. Both of label sets L_1 and L_2 are considered as single-activity label sets. Therefore, the result $discore_{L_k}[ws_{current}]$ is derived from averaging divergence score $discore_{L_i}[ws_{current}]$ from L_1 and L_2 corresponding to the same window size.

The second stage after metrics derivation refers to optimal window size recommendation in Algorithm 1. It involves finding optimal window size for each label set L_k with given metrics computed from the first stage. For each label set L_k, the optimal window size is derived from metrics intersection between current window size and next window size in an iterative operation. In order to decide the window size to be optimal, the metrics for impurity proportion and normalised squared deviation of divergence score are used. Given the divergence score ds_i and mean of divergence scores ($\overline{ds}$) from all possible window sizes, normalised squared deviation ndd_i can be computed via:

$$ndd_i = \frac{dd_i - \min(dd)}{\max(dd) - \min(dd)} \tag{5}$$

where squared deviation dd_i of ds_i is defined as:

$$dd_i = (ds_i - \overline{ds})^2 \tag{6}$$

As a result, divergence deviation of available window sizes would be scaled from 0 to 1 and can be used to find the intersection with impurity proportion. For temporal segmentation, larger time-interval window size corresponds to the following assumptions:

1. Greater impurity proportion is included.
2. Increasing mean of maximum divergences from all features.

These assumptions would be validated via the experiments in next section. In Algorithm 1, the intersection function would return the optimal window size $ws_{optimal}$. The $ws_{optimal}$ is non-empty in a condition of $range_{pimpure} \cap range_{dideviation}$ where

$$range_{pimpure} = (pimpure_{L_k}[ws_{current}],\ pimpure_{L_k}[ws_{next}]) \tag{7}$$

and

$$range_{dideviation} = (dideviation_{current},\ dideviation_{next}) \tag{8}$$

Given the ranges $range_{pimpure}$ and $range_{dideviation}$, the optimal window size is defined based on the first intersection occurrence.

Hence, the candidate of optimal window size (either $ws_{current}$ or ws_{next}) for $ws_{optimal}$ depends on closest absolute distance of impurity proportion (either $pimpure_{L_k}[ws_{current}]$ or $pimpure_{L_k}[ws_{next}]$) to average value (centroid) of S value set:

$$S = \begin{Bmatrix} pimpure_{L_k}[ws_{current}], \\ pimpure_{L_k}[ws_{next}], \\ dideviation_{current}, \\ dideviation_{next} \end{Bmatrix}$$

The final output of algorithm results in optimal window sizes ws_{rec} corresponding to all L_k in L label sets.

5. EXPERIMENTS AND EVALUATION

In this section, we present the experimental settings and evaluation. The method is validated with the benchmark OPPORTUNITY Activity Recognition Dataset [31] from UCI repository. It is a dataset that was collected from wearable, object and ambient sensors for activity recognition. This rich dataset contains 242 features in total. For the purpose of this study, we only use 101 features that are related to body sensors and wearables. The remaining unused features are associated with sensors that are attached to objects such as knife, spoon, plate and fridge.

There are several characteristics of this dataset that can be associated to our problem definition:

1. Dynamic sensor data from ubiquitous sensors in an **irregular manner**. In several occasions, feature values in a sensor stream can be empty due to unavailability of specific ubiquitous device. For example, unavailability of accelerometer sensor in certain time period would result in empty reading for three axes of accelerometer values. Therefore, the data contains inherent sparsity problem, which may result in less accurate and inconsistent performance of a classifier model.

2. The dataset contains **multi-activity label sets**, which is suitable for our evaluation in terms of scalability of the proposed method.

5.1 Data Preparation

The OPPORTUNITY Activity Recognition Dataset contains several activity sets including 5 high level activities, 4 locomotion activities, 17 gesture activities, low-level actions relating 13 actions to 23 objects. There are 4 users associated with the sensor data, 6 recordings for each user. In our experiments, we randomly selected 2 recordings for each user. Time-interval based temporal segmentation is performed for each recording in a streaming fashion. All temporal window instances from selected recordings of users are combined, producing a dataset that would be used for training and testing phases. Therefore, our method would be demonstrated and validated against this dataset for multi-activity recognition scenario. In this instance, we use the following label sets:

1. ***High level activity (HLA)*** label set:

$$L_1 = \begin{Bmatrix} \text{Relaxing,} \\ \text{Coffee time,} \\ \text{Early morning,} \\ \text{Cleanup,} \\ \text{Sandwich time,} \\ \text{None} \end{Bmatrix}$$

2. ***Locomotion activity (LA)*** label set:

$$L_2 = \{\text{Stand, Walk, Sit, Lie, None}\}$$

3. ***Gesture activity (GA)*** label set:

$$L_3 = \begin{Bmatrix} \text{Open Door 1, Close Door 1,} \\ \text{Open Door 2, Close Door 2,} \\ \text{Open Fridge, Close Fridge,} \\ \text{Open Dishwasher, Close Dishwasher,} \\ \text{Open Drawer 1, Close Drawer 1,} \\ \text{Open Drawer 2, Close Drawer 2,} \\ \text{Open Drawer 3, Close Drawer 3,} \\ \text{Clean Table, Drink from Cup, Toggle Switch,} \\ \text{None} \end{Bmatrix}$$

For the instance without class label L_k above, it is automatically assigned to "None". Furthermore, feature extraction is subsequently performed on each window that is produced by temporal segmentation process. The generated features include:

- Mean
- Minimum
- Maximum
- Standard deviation
- IQR (interquartile range)
- Median
- RMS (root-mean-square)

As a result, there are 707 features that can be used for training and testing a classifier. For recognition of activities, these following classifiers are used:

1. Naive Bayes (**NB**) classifier
2. Decision Tree (**J48**) classifier
3. Random Forests (**RF**) classifier

Throughout our experiments, the activity recognition models are built using a well-known data mining software: Weka 3.8 [15] with default parameters corresponding to each classifier. For multi-activity recognition, a new label set is constructed from the combination of three label sets. In other words, the new label set of multi-activity recognition L_4 contains labels in a given structure L_1:L_2:L_3 (denoted as **HLA:LA:GA**), e.g. "Coffee time:Sit:Drink from Cup".

The boundary to search for optimal window size is defined as the following:

- $ws_{start} = 600$ milliseconds (0.6 seconds).
- $ws_{end} = 5000$ milliseconds (5 seconds).
- $ws_{step} = 100$ milliseconds (0.1 seconds).

5.2 Observation of OPTWIN Metrics

In order to validate our assumptions on increasing window size for temporal segmentation towards OPTWIN metrics, impurity proportion and mean of maximum divergences (from all features) are observed in this section.

5.2.1 *Impurity Proportion*

As shown in Figure 4, the impurity proportion increases as window size used for temporal segmentation is larger. The decrease of purity is especially prominent for **LA**, **GA** and **HLA:LA:GA** label sets. However, it is less noticeable for **HLA** as its class labels have larger activity duration compared to other label sets. It is within the expectation that the impurity proportion for **HLA:LA:GA** label set would be significantly dominant. Essentially, this can be caused by the dynamic combination of classes from available label sets.

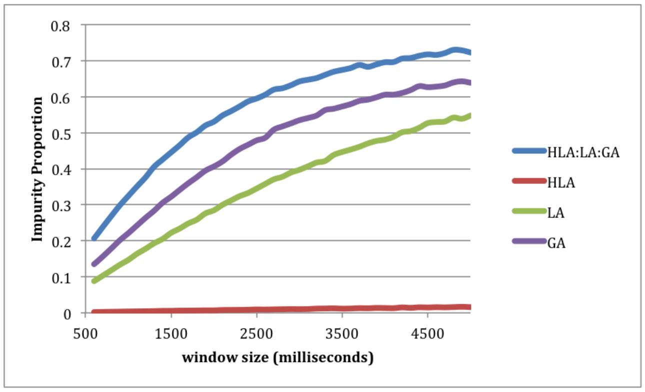

Figure 4: Impurity proportion

Furthermore, we noticed that the locomotive state changes frequently in OPPORTUNITY dataset (as described in Figure 5). These activities include instances without a class label, which are reassigned as "none". As a result, the frequent changes of locomotive state lead to impurity level exceeding 50% at 4.5 seconds (window size) for locomotion activities.

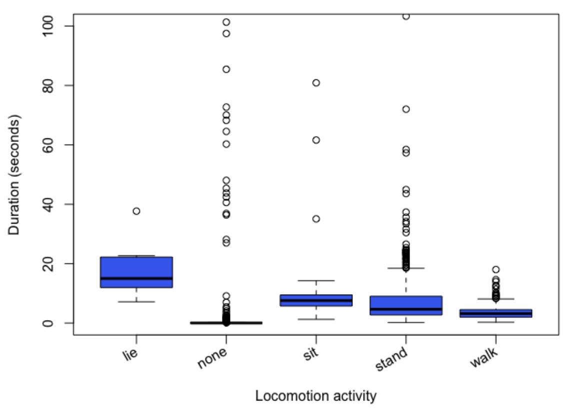

Figure 5: Box plot of locomotion activity duration (zoomed). Maximum locomotion activity duration is 224.994 seconds for "none" class label.

5.2.2 *Mean Maximum Divergence*

Before calculating maximum divergence for each feature, the feature values are converted to a histogram as explained in the methodology section. However, the sparsity problem is inherent in the OPPORTUNITY dataset since a sensor can be unavailable for a period of time in a stream. To solve this problem in relation to KL divergence computation, the empty feature values are allocated to the middle of defined b bins. In this case, 100 bins ($b = 100$) are used for the normalisation to a histogram. For the label set that is used for multi-activity recognition, $discore_{L_4}$ (mean maximum divergence of **HLA:LA:GA** label set) is computed by averaging the mean maximum divergences of associated label sets. In other words, $discore_{L_4} = \frac{1}{3}(discore_{L_1} + discore_{L_2} + discore_{L_3})$.

Similarly, the mean average divergence metrics appear to be increasing corresponding to larger time window size (up to 10 seconds) as shown in Figure 6. Moreover, the mean maximum divergence score is expected to degrade after reaching the peak at a certain time window size. This phenomenon is clearly shown in the sudden drop of $discore_{L_4}$ and $discore_{L_3}$ at 33 seconds window size. In many applications, this could be viewed as a convex optimisation problem. However, our observation revealed that larger window size would eventually reduce the number of unique class labels (especially for L_3 and L_4). This observation indicates that the optimal divergence score would not be convincing since the number of unique class labels will significantly decrease if activities are shifted rapidly. It should be noted that the range of divergence score to be used in our analysis is defined between 600 milliseconds and 5000 milliseconds for finding optimal window size.

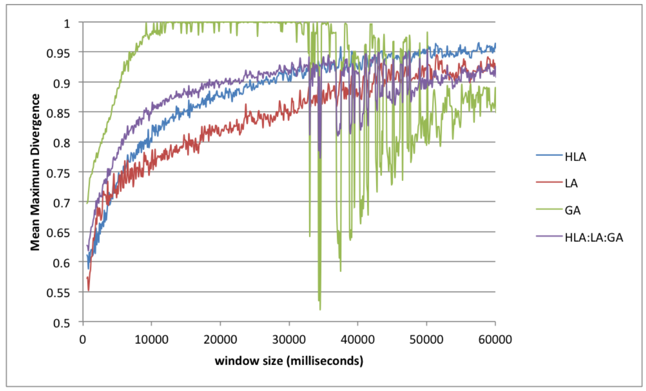

Figure 6: Mean of maximum divergences

5.3 Optimal Window Sizes

The OPTWIN algorithm finds the optimal window sizes for label sets based on the above metrics within a constrained search range. The optimal window size is set from the closest window size's impurity proportion to the intersection of impurity proportion and normalised deviation of divergence score. As an example, the optimal window size for **HLA:LA:GA** label set is found to be 1.2 seconds as depicted in Figure 7.

Performing sensitivity analysis of every label set would be a time consuming task in overall activity recognition operation, excluding parameter tuning for each classifier. The proposed method specifically operates as multi-objective function that intends to gain the balance between minimising the impurity proportion of time segments and maximising class separability measure. Moreover, the algorithm is scalable to tackle multi-activity recognition problem. For our experiments, 1-second window size is used as the baseline comparison. It was mentioned previously that many experiments leveraged 1-second as a heuristic way to select the window size, especially for activity recognition. The performances of classifiers are tested with F1 score (harmonic mean of precision and recall) via 10-folds CV (Cross Validation) on optimal and baseline window sizes.

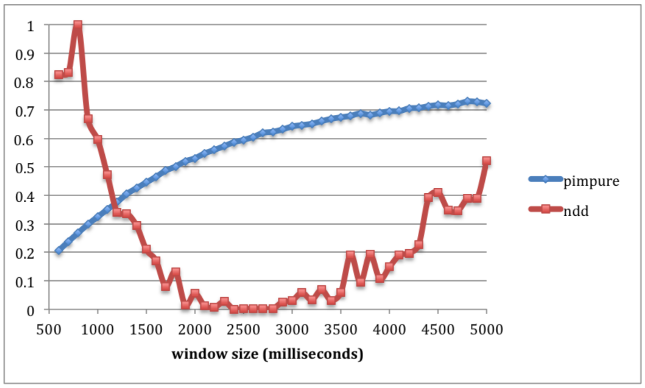

Figure 7: Intersection between impurity proportion (pimpure) and normalised squared deviation of divergence score (ndd)

The following $ws_{optimal}$ are returned from the window size recommendation:

1. **HLA**: 2.4 seconds - single-activity recognition
2. **LA**: 1.5 seconds - single-activity recognition
3. **GA**: 1.4 seconds - single-activity recognition
4. **HLA:LA:GA**: 1.2 seconds - multi-activity recognition

5.4 Sensitivity Analysis

In this section, the result of sensitivity analysis for multi-activity recognition is shown by the performance of classifiers on L_4. Thus, this brief analysis will verify that general classifier performance varies for each window size. The performances of classifiers are tested with both F1 score and accuracy via 10-folds CV. As shown in Figure 8 and Figure 9, fluctuation of quality performances on the classifiers are prominent for both F1 score and accuracy as the window size increases, especially on RF (best classifier). Thus, this is essentially aligned with the motivations and challenges in this research to find optimal window size for multi-activity recognition. It should be clear that multi-activity recognition in this paper aims to detect multiple activities (one activity from each label set) for a given time window instance.

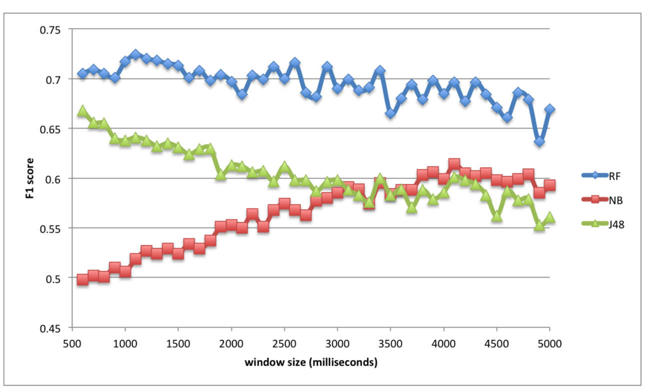

Figure 8: Sensitivity analysis - F1 score (OPPORTUNITY)

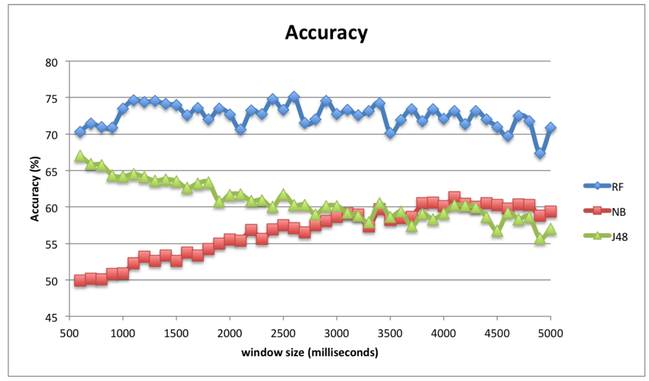

Figure 9: Sensitivity analysis - Accuracy (OPPORTUNITY)

5.5 Discussion for Optimal Window Size

From all performance comparisons of F1 scores shown in Figure 10 and Figure 11, NB classifier gains significant improvement for F1 scores for the defined label sets. This suggests that finding optimal window size would generally improve the performance of generative model based classifier such as NB in comparison to the baseline (1-second window size). It is important to note that in several applications of activity recognition, generative models can outperform discriminative approaches, especially when there are errors associated with activity labels (e.g. streaming data from crowdsensing [10]). However, the performances of F1 scores appear to be slightly worse for the optimal window size of single-activity recognition of **HLA** and **LA**. The slight degradation of these classifier performances can be found as the trade-offs to finding balanced window size for recognising multiple activities. However, an improvement of single activity recognition for the atomic activity (**GA**) is gained. In short, finding optimal window sizes through OPTWIN algorithm corresponds to better and balanced performance for multi-activity recognition (e.g. **HLA:LA:GA**) in comparison to single activity recognition from a single label set. Furthermore, the recommended window size can be used for real-time activity recognition according to user application context (either single-activity recognition or multi-activity recognition).

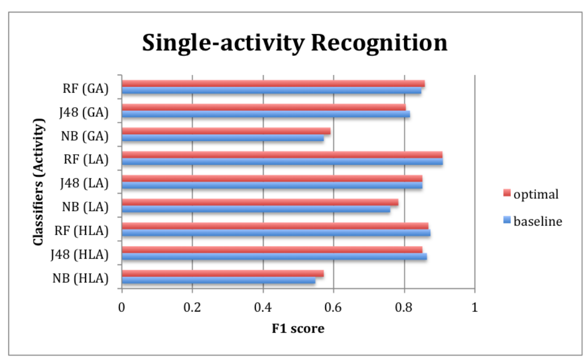

Figure 10: F1 score comparison between optimal and baseline window size for single-activity recognition (HLA, LA and GA)

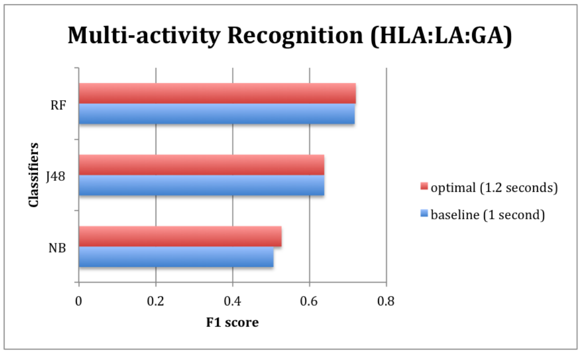

Figure 11: F1 score comparison between optimal and baseline window size for multi-activity recognition (HLA:LA:GA)

6. CONCLUSION

In this paper, we have proposed a specialised technique of finding optimal window sizes for time-interval based temporal segmentation. This technique is composed of multi-objective function that aims for the balance between minimising the impurity segments during temporal segmentation and maximising class separability measure. The scalability of proposed technique is demonstrated by the capabilities to produce optimal window size for each label set, including one for the combined label set (for multi-activity recognition). From the experiment results, the optimal window size for multi-activity recognition is found to be improving the quality of classifier performances from the baseline window size. In addition, the improvement appeared to be dominant for generative model based recognition techniques (such as Naive Bayes classifier). Nevertheless, OPTWIN can be used as an alternative method for automatic selection of window size instead of simple one second heuristic or a very time consuming sensitivity analysis. For future works, we would like to investigate a mechanism that allow window size to grow and shrink according to activity context.

The contribution of this paper is mainly targeted towards time-interval temporal segmentation. Therefore, we intend to extend this technique for dynamic window technique where segmentation size can expand and shrink in data stream processing. Moreover, another challenge is associated with the study of dependency between activity label sets. In this paper, the multi-label classification is achieved by constructing a new label set that consists of the combination of labels from available label sets. Unfortunately, this is not robust in terms for inferring the accuracy for each activity label. A more robust algorithm of multi-label classification is needed for multi-activity recognition in the scenario where label sets can be dependent or independent from each other.

7. REFERENCES

[1] T. Abeel, Y. V. d. Peer, and Y. Saeys. Java-ml: A machine learning library. *Journal of Machine Learning Research*, 10(Apr):931–934, 2009.

[2] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, 43(3):16, 2011.

[3] R. Bellman. On the approximation of curves by line segments using dynamic programming. *Communications of the ACM*, 4(6):284, 1961.

[4] L. Benoit, R. Benoit, É. Belin, R. Vadaine, D. Demilly, F. Chapeau-Blondeau, and D. Rousseau. On the value of the kullback–leibler divergence for cost-effective spectral imaging of plants by optimal selection of wavebands. *Machine Vision and Applications*, pages 1–11, 2015.

[5] J. Cancela, M. Pastorino, E. Moreno, and M. A. Waldmeyer. A mobile monitoring tool for the automatic activity recognition and its application for parkinson's disease rehabilitation. In *World Congress on Medical Physics and Biomedical Engineering, June 7-12, 2015, Toronto, Canada*, pages 1457–1460. Springer, 2015.

[6] L. Chen, J. Hoey, C. Nugent, D. Cook, and Z. Yu. Sensor-based activity recognition. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(6):790–808, Nov 2012.

[7] D. Cook, M. Schmitter-Edgecombe, A. Crandall, C. Sanders, and B. Thomas. Collecting and disseminating smart home sensor data in the casas project. In *Proceedings of the CHI Workshop on Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research*, pages 1–7, 2009.

[8] D. J. Cook, G. M. Youngblood, E. O. Heierman III, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja. Mavhome: An agent-based smart home. In *PerCom*, volume 3, pages 521–524, 2003.

[9] T. Feng and H. J. Timmermans. Transportation mode recognition using {GPS} and accelerometer data. *Transportation Research Part C: Emerging Technologies*, 37:118 – 130, 2013.

[10] R. K. Ganti, F. Ye, and H. Lei. Mobile Crowdsensing : Current State and Future Challenges. *IEEE Communications Magazine*, 49(11):32–39, 2011.

[11] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.

[12] D. Gordon, J. Czerny, and M. Beigl. Activity recognition for creatures of habit. *Personal and ubiquitous computing*, 18(1):205–221, 2014.

[13] B. Guo, R. I. Damper, S. R. Gunn, and J. D. Nelson. A fast separability-based feature-selection method for high-dimensional remotely sensed image classification. *Pattern Recognition*, 41(5):1653–1662, 2008.

[14] T. Guo, Z. Yan, and K. Aberer. An adaptive approach for online segmentation of multi-dimensional mobile data. In *Proceedings of the Eleventh ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 7–14. ACM, 2012.

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, Nov. 2009.

[16] S. Hemminki, P. Nurmi, and S. Tarkoma. Accelerometer-based transportation mode detection on smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, SenSys '13, pages 13:1–13:14, New York, NY, USA, 2013. ACM.

[17] J. Himberg, K. Korpiaho, H. Mannila, J. Tikanmaki, and H. T. Toivonen. Time series segmentation for context recognition in mobile devices. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 203–210. IEEE, 2001.

[18] S. S. Intille, K. Larson, E. M. Tapia, J. S. Beaudin, P. Kaushik, J. Nawyn, and R. Rockinson. Using a live-in laboratory for ubiquitous computing research. In *International Conference on Pervasive Computing*, pages 349–365. Springer, 2006.

[19] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. Mynatt, T. E. Starner, and W. Newstetter.

The aware home: A living laboratory for ubiquitous computing research. In *International Workshop on Cooperative Buildings*, pages 191–198. Springer, 1999.

[20] Y. Kim, F. Pereira, F. Zhao, A. Ghorpade, P. Zegras, and M. Ben-Akiva. Activity recognition for a smartphone based travel survey based on cross-user history data. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 432–437, Aug 2014.

[21] N. C. Krishnan and D. J. Cook. Activity recognition on streaming sensor data. *Pervasive and Mobile Computing*, 10, Part B:138 – 154, 2014.

[22] B. Kröse, T. Kasteren, C. Gibson, T. Dool, et al. Care: Context awareness in residences for elderly. In *Sixth International Conference of the International Society for Gerontechnology*, 2008.

[23] S. Kullback and R. A. Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[24] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

[25] D. Liu, D.-M. Sun, and Z.-D. Qiu. Feature selection for fusion of speaker verification via maximum kullback-leibler distance. In *IEEE 10th INTERNATIONAL CONFERENCE ON SIGNAL PROCESSING PROCEEDINGS*, pages 565–568. IEEE, 2010.

[26] I.-S. Oh, J.-S. Lee, and C. Y. Suen. Analysis of class separation and combination of class-dependent features for handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1089–1094, 1999.

[27] S. Pirttikangas, K. Fujinami, and T. Nakajima. Feature selection and activity recognition from wearable sensors. In *International Symposium on Ubiquitious Computing Systems*, pages 516–527. Springer, 2006.

[28] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas. Emotionsense: a mobile phones based adaptive platform for experimental social psychology research. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 281–290. ACM, 2010.

[29] J. Read, I. Žliobaitė, and J. Hollmén. Labeling sensing data for mobility modeling. *Information Systems*, 57:207–222, 2016.

[30] D. Riboni and C. Bettini. Cosar: hybrid reasoning for context-aware activity recognition. *Personal and Ubiquitous Computing*, 15(3):271–289, 2011.

[31] H. Sagha, S. T. Digumarti, J. d. R. Millán, R. Chavarriaga, A. Calatroni, D. Roggen, and G. Tröster. Benchmarking classification techniques using the opportunity human activity dataset. In *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, pages 36–40. IEEE, 2011.

[32] D. Sánchez, M. Tentori, and J. Favela. Activity recognition for the smart hospital. *IEEE intelligent systems*, 23(2):50–57, 2008.

[33] X. Su, H. Tong, and P. Ji. Activity recognition with smartphone sensors. *Tsinghua Science and Technology*, 19(3):235–249, June 2014.

[34] D. Tacconi, O. Mayora, P. Lukowicz, B. Arnrich, C. Setz, G. Troster, and C. Haring. Activity and emotion recognition to support early diagnosis of psychiatric diseases. In *2008 Second International Conference on Pervasive Computing Technologies for Healthcare*, pages 100–102. IEEE, 2008.

[35] R. L. S. Torres, D. C. Ranasinghe, and Q. Shi. Evaluation of wearable sensor tag data segmentation approaches for real time activity classification in elderly. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 384–395. Springer, 2013.

[36] G. Tsoumakas and I. Katakis. Multi-label classification: An overview. *Dept. of Informatics, Aristotle University of Thessaloniki, Greece*, 2006.

[37] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488, 2008.

[38] L. Wang, T. Gu, X. Tao, and J. Lu. Sensor-based human activity recognition in a multi-user scenario. In *European Conference on Ambient Intelligence*, pages 78–87. Springer, 2009.

[39] Y. Zhang, S. Li, T. Wang, and Z. Zhang. Divergence-based feature selection for separate classes. *Neurocomputing*, 101:32–42, 2013.

[40] Y. Zhou, U. Porwal, C. Zhang, H. Q. Ngo, L. Nguyen, C. Ré, and V. Govindaraju. Parallel feature selection inspired by group testing. In *Advances in Neural Information Processing Systems*, pages 3554–3562, 2014.

[41] C. Zhu and W. Sheng. Human daily activity recognition in robot-assisted living using multi-sensor fusion. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 2154–2159. IEEE, 2009.