A Feature Selection-Based Framework for Human Activity Recognition Using Wearable Multimodal Sensors

Mi Zhang
Ming Hsieh Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089
mizhang@usc.edu

Alexander A. Sawchuk
Ming Hsieh Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089
sawchuk@sipi.usc.edu

ABSTRACT

Human activity recognition is important for many applications. This paper describes a human activity recognition framework based on feature selection techniques. The objective is to identify the most important features to recognize human activities. We first design a set of new features (called physical features) based on the physical parameters of human motion to augment the commonly used statistical features. To systematically analyze the impact of the physical features on the performance of the recognition system, a single-layer feature selection framework is developed. Experimental results indicate that physical features are always among the top features selected by different feature selection methods and the recognition accuracy is generally improved to 90%, or 8% better than when only statistical features are used. Moreover, we show that the performance is further improved by 3.8% by extending the single-layer framework to a multi-layer framework which takes advantage of the inherent structure of human activities and performs feature selection and classification in a hierarchical manner.

Categories and Subject Descriptors

I.5 [**Pattern Recognition**]: Applications

General Terms

Algorithms, Design, Experimentation, Performance

Keywords

Feature Design, Feature Selection, Human Activity Recognition, Wearable Multimodal Sensors, Pattern Recognition

1. INTRODUCTION

Human activity recognition using wearable sensors has received increasing attention due to a wide range of potential applications, including long term physical fitness monitoring, sleep quality monitoring, rehabilitation, and intelligent assistance for people with cognitive disorders [1] [2]. Recent work involves prototyping wearable sensor systems, building human activity datasets, and developing pattern recognition and machine learning algorithms to model and recognize human activities. In this paper, we focus on feature selection and pattern recognition algorithms that improve human activity classification performance.

It is well understood that high quality features are essential to improve the classification accuracy of any pattern recognition system. In human activity recognition, features such as mean, variance, correlation, and FFT coefficients computed from mechanical motion measurements are commonly used [3]. To perform classification, one naive idea is to pool all available features into one vector used as input to the classifier. The disadvantage here is that some features may be irrelevant or redundant, and do not provide new information to improve the classification accuracy. Some features might even confuse the classifier rather than help discriminate various activities. What is worse, due to the "curse of dimensionality", the performance may degrade sharply as more features are added when there is not enough training data to reliably learn all the parameters of the activity models [4]. Therefore, to achieve the best classification performance, the dimensionality of the feature vector should be as small as possible, keeping only the most salient and complementary features. In addition, keeping the dimensionality small could reduce the computational cost such that the recognition algorithms can be implemented and run on lightweight wearable devices such as mobile phones.

The two main techniques that are used to identify important features and reduce dimensionality in pattern recognition are: (1) feature transformation - creating new features based on transformations or combinations of the original extracted feature set; and (2) feature selection - selecting the best subset of the original extracted feature set [5]. Both of them have been used in the wearable sensor community for recognizing various human activities. One common strategy is to apply either feature transformation or feature selection to get a fixed set of features for the whole set of activities to be recognized. For example, in [6], a correlation-based feature selection method was used to select a subset of features. A 87% classification accuracy was achieved when using the top eight features selected to classify six basic human activities. In [7], researchers identified energy as the least significant feature among all the five available features by using a sequential backward elimination method. In [8], the authors applied three feature selection methods: Relief-F, Simba, and mRMR to assess the relevance of features for

BodyNets 2011, Beijing, China, 7-8 November 2011, ISBN (978-1-936968-29-9).
.

discriminating 15 different activities. All these three methods achieved similar performance. The other strategy assumes that different activities may be characterized by different sets of features. In [9], by performing cluster analysis, Huynh *et al.* showed that the classification performance could be improved by selecting features and window lengths for each activity separately. Lester *et al.* in [10] also demonstrated that a feature's usefulness depends on the specific activity to be inferred. They applied a modified version of AdaBoost [11] to select the top 50 features and then learn an ensemble of discriminative static classifiers based on the selected features for each activity. They found that the selected features were different for different activities.

In this paper, we focus on feature design and evaluation based on feature selection techniques. The rationale to use feature selection is that the selected features retain their original meanings that we believe are important for better understanding of human activities. And our goal is to identify the most important features to recognize various human activities. The contributions of this work are listed here: we first design a new set of features (called physical features) based on the physical parameters of human motion. We expect these features to represent motion more accurately and concisely than commonly used statistical features such as mean and variance. Then, we use both statistical and physical features in a single-layer feature selection and classification framework to systematically analyze and evaluate their impact on the performance of the recognition system for the whole set of activities to be recognized. To further improve the recognition performance, we follow the ideas from [9] and [10] to extend the single-layer framework to a multi-layer framework that selects the most important features for different activities in a hierarchical manner.

The paper is organized as follows. Section 2 introduces the sensing platform and dataset used in this study. Section 3 defines the statistical and physical features. Section 4 describes the feature selection techniques. Section 5 and 6 present the design and evaluation of the single-layer and multi-layer hierarchical feature selection and classification frameworks respectively. Finally, Section 7 summarizes this work and establishes directions for future work.

2. SENSING PLATFORM AND DATASET

For this work, data is recorded using an off-the-shelf multimodal sensing platform called MotionNode [12]. MotionNode is a 6-DOF inertial measurement unit (IMU) specifically designed for human motion sensing applications. It integrates a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis magnetometer. In this work, only the data sampled from the accelerometer and gyroscope is considered. The measurement range for each axis of accelerometer and gyroscope is $\pm 6g$ and $\pm 500dps$ respectively. The sampling rates for both accelerometer and gyroscope are set to 100 Hz.

Six participants with different gender, age, height, and weight are recruited to perform nine types of activities: walk forward, walk left, walk right, go upstairs, go downstairs, jump up, run, stand, and sit. These activities correspond to the most common activities in people's daily life and are useful for both elder care and personal fitness applications. During data collection, a MotionNode is attached firmly onto the participant's right front hip. Each participant performs five trials for each activity on different days at various indoor and outdoor locations without supervision.

3. FEATURE DESIGN

3.1 Activity Model

In this paper, we model each activity based on the sliding-window strategy. Specifically, we divide the continuous sensor streams into fixed length windows. By choosing a proper window length, we assume that all the information of each activity can be extracted from each single window. The information is then transformed into a feature vector by computing various features over the sensor data within each window. Here, a window of length 2 seconds with a 50% overlap is used. We now describe two sets of features we incorporated in our recognition framework.

3.2 Statistical Features

The first set of features are statistical features computed from each axis (channel) of both accelerometer and gyroscope. Table 1 shows the list of statistical features we consider in this work. Some of them have been intensively investigated in previous studies and proved to be useful for activity recognition [3] [7] [9]. For example, variance has been proved to achieve consistently high accuracy to differentiate activities such as walking, jogging, and hopping [9]. Correlation between each pair of sensor axes helps differentiate activities that involve translation in single dimension such as walking and running from the ones that involve translation in multi-dimension such as stair climbing [7]. We also consider statistical features that have been successfully applied in similar recognition problems. Examples are zero crossing rate, mean crossing rate, and first-order derivative. These features have been heavily used in human speech recognition and handwriting recognition problems.

3.3 Physical Features

The second set of features are called physical features, which are derived based on our physical interpretations of human motion. In this work, MotionNode is placed at the subject's front right hip, oriented so the x axis points to the ground and is perpendicular to the plane formed by y and z axes. We assume that the sensor location and orientation are known *a priori*. Some of our physical features are computed and optimized based on this prior knowledge. Although this assumption limits the generalization of our physical features to be applied to other locations and orientations to some extent, it simplifies the problem and allows us to focus on designing features with strong physical meanings so as to better describe human motion[1].

It should be noted that the way to compute physical features is different from statistical features. For statistical features, each feature is extracted from each sensor axis (channel) individually. In comparison, most of the physical features are extracted from multiple sensor channels. In other words, sensor fusion is performed at feature level for physical features. In the rest of this section, we explain the physical features in great detail.

[1]We have developed an algorithm to automatically identify the orientation of the sensing device. Based on this algorithm, any orientation can be virtually oriented to a predefined orientation. Since this algorithm is not the focus of this paper, we describe the algorithm elsewhere. Please contact the authors for details of the algorithm if interested.

Feature	Description
Mean	The DC component (average value) of the signal over the window
Median	The median signal value over the window
Standard Deviation	Measure of the spreadness of the signal over the window
Variance	The square of standard deviation
Root Mean Square	The quadratic mean value of the signal over the window
Averaged derivatives	The mean value of the first order derivatives of the signal over the window
Skewness	The degree of asymmetry of the sensor signal distribution
Kurtosis	The degree of peakedness of the sensor signal distribution
Interquartile Range	Measure of the statistical dispersion, being equal to the difference between the $75th$ and the $25th$ percentiles of the signal over the window
Zero Crossing Rate	The total number of times the signal changes from positive to negative or back or vice versa normalized by the window length
Mean Crossing Rate	The total number of times the signal changes from below average to above average or vice versa normalized by the window length
Pairwise Correlation	Correlation between two axes (channels) of each sensor and different sensors
Spectral Entropy	Measure of the distribution of frequency components

Table 1: Statistical features with symbols and brief descriptions

1. **Movement Intensity (MI)**: MI is defined as

$$MI(t) = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2}, \quad (1)$$

the Euclidean norm of the total acceleration vector after removing the static gravitational acceleration, where $a_x(t)$, $a_y(t)$, and $a_z(t)$ represent the t^{th} acceleration sample of the x, y, and z axis in each window respectively. This feature is independent of the orientation of the sensing device, and measures the instantaneous intensity of human movements at index t. We do not use MI directly. Instead, we compute the mean (AI) and variance (VI) of MI over the window and use them as two features given by

$$AI = \frac{1}{T}\left(\sum_{t=1}^{T} MI(t)\right) \quad (2)$$

$$VI = \frac{1}{T}\left(\sum_{t=1}^{T} (MI(t) - AI)^2\right) \quad (3)$$

where T is the window length.

2. **Normalized Signal Magnitude Area (SMA)**: SMA is defined as

$$SMA = \frac{1}{T}\left(\sum_{t=1}^{T} |a_x(t)| + \sum_{t=1}^{T} |a_y(t)| + \sum_{t=1}^{T} |a_z(t)|\right), \quad (4)$$

the acceleration magnitude summed over three axes within each window normalized by the window length. This feature has been used in previous studies and is regarded as an indirect estimation of energy expenditure [13] [14] [15].

3. **Eigenvalues of Dominant Directions (EVA)**: When a subject jumps, a large acceleration component along the vertical direction is expected. Likewise, when a subject runs forward, there should be a large acceleration component along the heading direction and a relatively large acceleration component along the vertical direction. To capture these effects, we calculate the covariance matrix of the acceleration data along x, y, and z axis in each window. The eigenvectors of the covariance matrix correspond to the dominant directions along which intensive human motion occurs. The eigenvalues measure the corresponding relative motion magnitude along the directions. In this work, we use the top two eigenvalues as our features, corresponding to the relative motion magnitude along the vertical direction and the heading direction respectively.

4. **Correlation between Acceleration along Gravity and Heading Directions (CAGH)**: Given the location and orientation of the sensing device described, the gravity direction is approximately parallel to x axis, and the subject's heading direction when walking is a combination of y and z axes. We first derive the Euclidean norm of the total acceleration vector along the heading direction, and then calculate the correlation coefficient between the acceleration in gravity direction and the derived acceleration along heading direction as our feature.

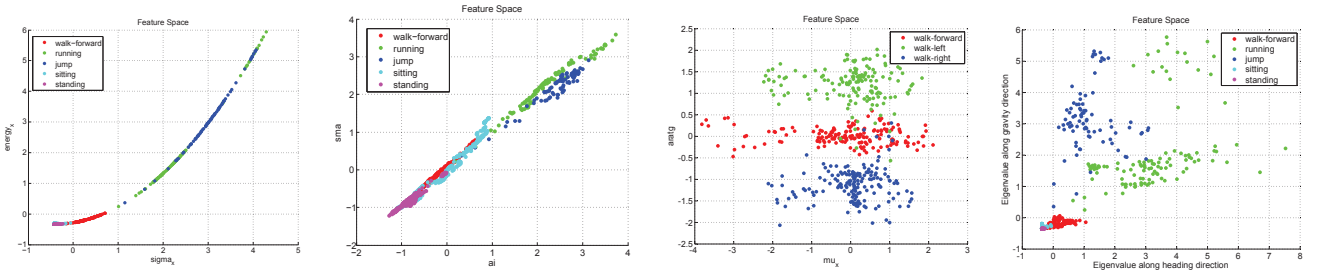
5. **Averaged Velocity along Heading Direction (AVH)**: AVH is approximated by first calculating the averaged velocities along y and z axes over the window, and then computing the Euclidean norm of those two velocities.

6. **Averaged Velocity along Gravity Direction (AVG)**: AVG is computed by averaging the instantaneous velocity along the gravity direction at each time t over the window. The instantaneous velocity at each time t is calculated through numerical integration of the acceleration along gravity direction.

7. **Averaged Rotation Angles related to Gravity Direction (ARATG)**: ARATG calculates the cumulative rotation angles around gravity direction. The cumulative sum is then divided by the window length for normalization. Since sensors are at the subject's front right hip, this feature captures the rotation movement of the human torso around gravity direction.

8. **Dominant Frequency (DF)**: The dominant frequency is defined as the frequency corresponding to the maximum of the squared discrete FFT component magnitudes of the signal from each sensor axis.

9. **Energy (ENERGY)**: Energy is calculated as the sum of the squared discrete FFT component magnitudes of the signal from each sensor axis. The sum is then divided by the window length for normalization.



(a) Sigma vs. Energy (b) AI vs. SMA (c) Mean vs. ARATG (d) Eigenvalue along heading direction vs. Eigenvalue along gravity direction

Figure 1: Scatter Plots in 2D Feature Space

The DC component of the FFT is excluded in this sum since it is already measured by the mean feature.

10. **Averaged Acceleration Energy (AAE)**: AAE is defined as the mean value of the energy over three acceleration axes.

11. **Averaged Rotation Energy (ARE)**: ARE is defined as the mean value of the energy over three gyroscope axes.

To explore the performance and correlation between features, a series of scatter plots in 2D feature space are shown in Figure 1. The horizontal and vertical axes represent two different features. Points in different colors represent different activities. In Figure 1(a), the relation between standard deviation and energy is described by a parabolic curve as expected. In Figure 1(b), the straight line along the diagonal means that SMA is highly correlated to the mean value of movement intensity (AI). This observation indicates that the l_1-norm (SMA) and l_2-norm (AI) of the acceleration signals in each window are quite equivalent. Thus using both SMA and AI is redundant. From the performance point of view, Figure 1(c) illustrates that the ARATG feature successfully partitions the data samples from walk-forward (red), walk-left (green), and walk-right (blue) into three isolated clusters, with each cluster containing data samples roughly from one single activity class. Finally, the scatter plot in Figure 1(d) demonstrates the discrimination power of the two eigenvalue features to differentiate walking (red), running (green), and jumping (blue). The vertical axis represents the eigenvalues along the gravity direction. As expected, the eigenvalues (relative motion magnitude) of jumping and running are larger than walking. The horizontal axis illustrates the eigenvalues along the heading direction. Since the speed of running is normally higher than walking, the relative motion magnitude of running along the heading direction is higher compared to walking. The eigenvalues of jumping are in the middle. This observation can be explained by the fact that normally people can not jump straight up and there is always a forwarding momentum exerted by human body when people jump. This forwarding momentum turns out to be larger when people jump than they walk.

3.4 Feature Normalization

Because the scale factors and units of the features described above are different, before we proceed to the feature selection stage, we normalize all the features to zero mean and unit variance using

$$f_{normalized} = \frac{f_{raw} - \mu}{\sigma} \tag{5}$$

where μ and σ are the empirical mean and standard deviation of a particular feature across all activity classes.

4. FEATURE SELECTION

The feature computation in the last section yields a total of 110 features including 87 statistical features and 23 physical features. To systematically assess the usefulness and identify the most important features for discriminating different activities, feature selection techniques are used. In general, feature selection techniques can be grouped into three categories: filter methods, wrapper methods and embedded methods [16]. This categorization is based on different ways to combine the feature selection search with the construction of the classification model. In our work, one filter method and two wrapper methods were investigated and are summarized here. These methods are selected due to their high popularity and usefulness in many pattern recognition and machine learning problems.

4.1 Feature Selection Methods

• **Relief-F**: Relief-F [17] is a popular filter method that estimates the relevance of features according to how well their values distinguish between the data points of the same and different classes that are near each other. Specifically, it computes a weight for each feature to quantify its merit. This weight is updated for each of the data points presented, according to the evaluation function:

$$w_i = \sum_{j=1}^{N} \left(x_i^j - nearmiss\left(x^j\right)_i\right)^2 - \left(x_i^j - nearhit\left(x^j\right)_i\right)^2 \tag{6}$$

where w_i represents the weight of the i^{th} feature, x_i^j represents the value of the i^{th} feature for data point x^j, N represents the total number of data points, $nearhit\left(x^j\right)$ and $nearmiss\left(x^j\right)$ denote the nearest point to x^j from the same and different class respectively. The higher the weight is, the more important is the feature. The major drawback of Relief-F is that it does not consider feature dependencies and therefore does not help remove redundant features.

• **SFC (Wrapper Method based on Single Feature Classification)**: In SFC [18], features are ranked based on

their individual classification performance. Features at the top of the ranking list are selected as the final feature subset. SFC is similar to Relief-F in the sense it can not capture redundant features either. However, unlike Relief-F, SFC uses the classifier's classification accuracy as the metric for feature evaluation.

• **SFS (Wrapper Method based on Sequential Forward Selection)**: In SFS [19], features are sequentially added one by one. Specifically, if one feature is needed, the feature with the best classification performance is selected. If more than one feature is needed, we add one feature at a time which in combination with the already selected features achieve the best classification performance. Compared to the two methods described earlier, the benefit of SFS is that it takes feature redundancy into consideration.

Let M represent the total number of features in the full feature set and N_f represent the number of features to be selected. Table 2 summarizes the computational costs of the three feature selection methods, with numerical examples for our case of $M = 110$. The computational cost is in the form of either the number of times of calling the feature evaluation function for filter methods, or the number of times of calling the classification algorithm for wrapper methods. In order to make this comparison more meaningful, the wrapper method based on exhaustive search is also included.

	$N_f = 1$	$N_f = 2$	$N_f = 3$	$N_f = n$
Relief-F	110	111	111	$M + 1$
SFC	110	111	111	$M + 1$
SFS	110	219	327	$\Theta(M^2)$
Exhaustive Search	110	5995	215820	$O(2^M)$

Table 2: Comparison of computational cost of feature selection methods with 110 features.

4.2 Classifier

The choice of classifier is critical to feature selection. Since the nature of feature selection problem is to select features from a high dimensional space, in this work, we choose Support Vector Machines (SVMs) with a linear kernel to be our learning machine. They have proved to be very effective in handling high dimensional data [20]. In addition, we use classification accuracy as our evaluation metric to assess the effectiveness of feature selection methods listed above.

5. SINGLE-LAYER FEATURE SELECTION AND CLASSIFICATION

As our first step to approach the problem of activity recognition using feature selection, we adopt a single-layer feature selection framework. That is, we take all activity classes into consideration at one time. Our goal is to find the best discriminating set of features for all activities.

5.1 Evaluation on Feature Selection Methods

To evaluate the effectiveness of the three feature selection methods, we adopt a leave-one-subject-out cross validation strategy. Specifically, we use the data from five subjects as training examples to select features and build activity models. Data from the left-out subject is used for testing. This process iterates for every subject. The final result is the averaged value across all the subjects.

Figure 2 shows the average testing classification error rates as a function of the number of features selected, ranging from 5 to 110 (full set), with interval equal to 5. Each line represents one feature selection method. The results show that across three feature selection methods, the classification errors taper off when 50 features are included, with approximately a 10% misclassification rate achieved on average. If we pick more features beyond the top 50, the performance only varies slightly. This matches the results in Lester's previous work [10]. If we look at each method individually, SFS performs the best in the sense that it achieves a 12% misclassification rate by using the first five features. In comparison, Relief-F is the worst since it gets a 47% misclassification rate with five features. SFC ranks in the middle with a 17% misclassification rate. Besides the classification performance, we record the averaged computational time of each method and list them in Table 3. As expected, SFS has the highest computational cost. SFC is computationally more expensive than Relief-F since the classifier is invloved.

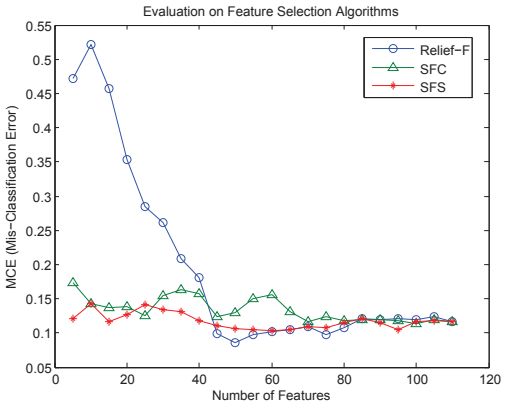


Figure 2: Testing classification error rates as a function of the number of features selected by different feature selection methods

Algorithm	Averaged Computational Time (second)
Relief-F	87.5
SFC	112.4
SFS	2780

Table 3: Averaged computational time of different feature selection methods

One interesting question to ask is whether the features selected by feature selection methods are truly important for our activity recognition problem. To answer this question, we first remove the top 50 features selected by feature selection methods in Figure 2. Then the same feature selection procedure is re-performed on the remaining feature set. Figure 3 shows the results. Across three methods, the classification errors are 13% to 23% higher compared to Figure 2 when 50 features are selected. This indicates that the top 50 features selected in Figure 2 contain more important information than the remaining feature set.

5.2 Feature Profiling on the Selected Features

The top 50 features selected by each feature selection method have been proven useful. To identify these features, we perform feature profiling. Since we use the leave-one-subject-out strategy on 6 subjects, not all the features selected in each iteration are the same. Thus we combine the

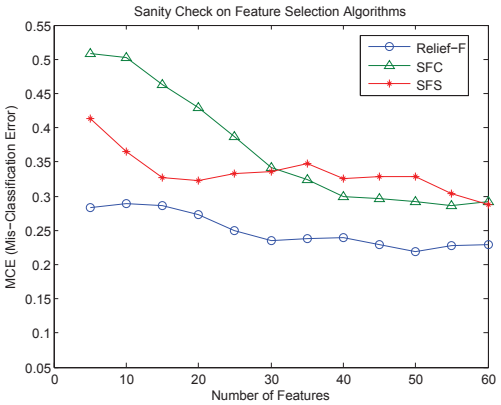


Figure 3: Sanity check on feature selection methods

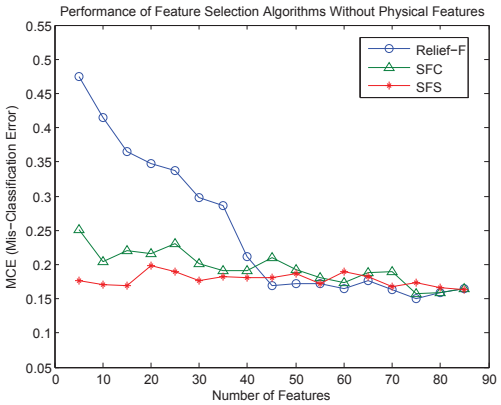


Figure 4: Testing classification error rates of feature selection methods without physical features

top 50 features selected in each iteration, which results in a total of 300 features with repetition for each method. The results are shown in Table 4. The second column lists the selected physical features. The third column lists the numbers and the percentages of physical features selected out of $23 \times 6 = 138$ physical features in 6 iterations. The corresponding results for statistical features are listed in the fourth column. For all three methods, although the numbers of selected physical features are less than statistical features, the corresponding percentages are higher. Among them, Relief-F selected the least number of physical features whereas SFC selects the most. This observation indicates that SFC considers physical features more critical in terms of classification performance. Compared to SFC, the number of physical features selected by SFS drops to 70. This is because some physical features are highly correlated to other features, so that SFS does not select the redundant ones. For example, SMA is not selected since it is highly correlated to AI (see Figure 1(b)). Likewise, the ENERGY features from all sensor channels are not selected because they are correlated to standard deviation (see Figure 1(a)).

Feature profiling gives the impression that the physical features play a major role in differentiating different activities. To validate this point, we remove all the physical features from the full feature set and perform feature selection on statistical features only. The result is presented in Figure 4. For all three methods, the misclassification rates are approximately 18% when the top 50 features are selected. This is 8% higher than the rates when physical features are included (see Figure 2). Based on all the results shown in this section, we can conclude that our self-designed physical features make a strong contribution and improve the classification performance to a great extent.

Algorithm	Physical Features Selected	No.(%) of Physical Features Selected	No.(%) of Statistical Features Selected
Relief-F	AI, VI, AAE, SMA, EVA(2), CAGH, AVH, AVG, ARATG, DF	67 (48.6%)	233 (44.6%)
SFC	AI, VI, AAE, SMA, EVA, AVH, ARATG, ARE, ENERGY, DF	103 (74.6%)	197 (37.7%)
SFS	AI, VI, AAE, EVA, CAGH, AVH, AVG, ARATG, ARE, DF	70 (50.7%)	230 (44.1%)

Table 4: Top 50 feature profiling of different feature selection methods

6. HIERARCHICAL FEATURE SELECTION AND CLASSIFICATION

The major limitation of the single-layer framework studied in the previous section is that all activities are considered simultaneously. As a result, it may not scale well if the number of activities to be recognized is large. To achieve high scalability, we propose here a multi-layer feature selection and classification framework. Specifically, we first group activities into subsets based on our understanding of the problem domain and then perform feature selection and classification in a hierarchical manner. In this scenario, the classifier in each layer considers a smaller number of activities. In addition, we now have the flexibility to use different features for different classifiers/activity subsets, instead of using the same feature set for all activity classes.

Figure 5 illustrates the structure of the hierarchical feature selection and classification framework. Blue boxes represent meta-classes we create to group activity subsets. Green boxes represent the nine types of activities to recognize. Now the problem of recognizing nine activity classes is broken down to seven distinct classification problems. The classifier at the top layer distinguishes between two meta-classes: *static activity* vs. *dynamic activity*. The static activity meta-class includes standing and sitting while the dynamic activity meta-class includes the remaining seven activities. Then, on the second layer, walking-related activities (walk forward, walk left, walk right, go upstairs, and go downstairs) are differentiated from jumping and running. Finally, the classifiers at the third and fourth layer focus on recognizing different activities related to walking.

To determine the best feature sets for classifiers at each layer, SFS is used due to its good performance. We follow the same leave-one-subject-out cross validation strategy to perform feature selection and classification at each layer. For classifiers 1, 2, 3, and 6, the maximum classification accuracy achieved and the corresponding number of features selected (in parentheses) are shown in Figure 5 in red. For the remaining classifiers, the number of features selected when achieving the maximum accuracy is greater than 50. To lower the computational cost, we use instead the number of

features at which the classification accuracy reaches the first local maximum. These parameters are shown in red and in parentheses as before. Based on the structure and the features selected at each layer, the averaged testing classification accuracy of the overall multi-layer classifier is 93.1%. This result is 3.8% higher than the accuracy achieved by the single-layer classifier when the top 50 features are used.

Table 5 lists the selected physical features at each classifier that are the most important to differentiate different activity subsets. For example, as expected, features AI and VI are useful in differentiating *static activity* and *dynamic activity* at classifier 1 (C1). At C2, intensity-related features such as VI, AAE, ARE are selected. AVH and CAGH are also selected since walking, running, and jumping have different velocities and intensities along the heading direction. At C4, the eigenvalues along both gravity and heading directions are selected, matching the result in Figure 1(d). At C6, ARATG is selected, matching the result in Figure 1(c). Finally, at C7, both AVG and CAGH are selected since going upstairs and going downstairs exhibit different velocities and intensities along the vertical direction.

Classifier	1	2	3	4	5	6	7
Physical Features Selected	AI, VI	VI, EVA(1), CAGH, AAE, AVH, ARE	AI, AAE	VI, EVA, CAGH	VI, AVG, ARATG, CAGH	ARATG, CAGH	CAGH, AVG

Table 5: The physical features selected at each layer

7. CONCLUSION AND FUTURE WORK

In this paper, a human activity recognition framework based on feature selection techniques is presented. We examined three feature selection methods and found that the sequential forward selection (SFS) method achieved the best performance compared to Relief-F and single feature classification (SFC) methods. In addition, we have demonstrated that our self-designed physical features make significant contributions to the recognition system. Finally, we have shown that the feature selection framework with a hierarchical structure improves the recognition performance compared to the single-layer framework. However, the structure of the hierarchical framework, i.e. how the different classes are arranged in meta-classes, is designed manually, based on domain knowledge. Although this type of knowledge may generally be helpful when designing recognition systems, in our future work, we will explore methods to learn the structure automatically in a data-driven manner.

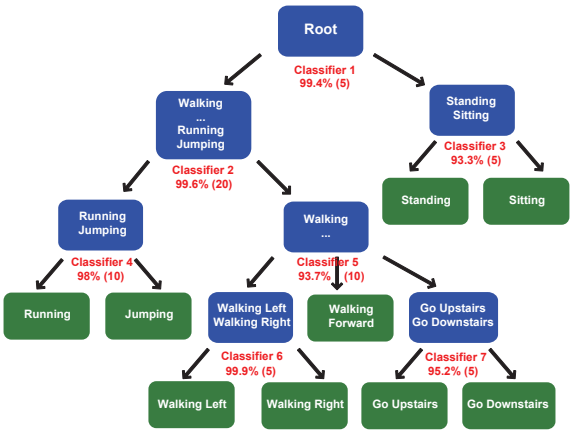


Figure 5: The structure and the performance of the hierarchical feature selection and classification framework

8. REFERENCES

[1] T. Choudhury et al. The Mobile Sensing Platform: An Embedded Activity Recognition System. *IEEE Pervasive Computing*, 7(2):32–41, April 2008.

[2] M. Zhang and A. Sawchuk. A customizable framework of body area sensor network for rehabilitation. In *ISABEL*, pages 1–6, Bratislava, Slovak Republic, 2009.

[3] L. Bao and S. Intille. Activity Recognition from User-Annotated Acceleration Data. In *Pervasive Computing*, pages 1–17, Linz/Vienna, Austria, 2004.

[4] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, Secaucus, NJ, USA, 2006.

[5] A. Jain, R. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(1):4–37, 2000.

[6] U. Maurer et al. Activity recognition and monitoring using multiple sensors on different body positions. In *BSN*, pages 113–116, Cambridge, MA, USA, 2006.

[7] R. Nishkam et al. Activity recognition from accelerometer data. In *IAAI*, pages 1541–1546, Pittsburgh, Pennsylvania, USA, 2005.

[8] L. Atallah et al. Sensor placement for activity detection using wearable accelerometers. In *BSN*, Biopolis, Singapore, 2010.

[9] T. Huynh et al. Analyzing features for activity recognition. In *sOc-EUSAI*, New York, USA, 2005.

[10] J. Lester et al. A hybrid discriminative/generative approach for modeling human activities. In *ICJAI*, pages 766–772, Edinburgh, Scotland, UK, 2005.

[11] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages 511–518, Kauai, HI, USA, 2001.

[12] http://www.motionnode.com.

[13] D. Karantonis et al. Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring. *IEEE Trans. Inf. Technol. Biomed.*, 10(1):156–167, 2006.

[14] A. Godfrey et al. Direct measurement of human movement by accelerometry. *Medical engineering & physics*, 30(10):1364–1386, 2008.

[15] J. Yang et al. Using acceleration measurements for activity recognition: An effective learning algorithm for constructing neural classifiers. *Pattern Recognition Letters*, 29:2213–2220, December 2008.

[16] Y. Saeys et al. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23:2507–2517, 2007.

[17] K. Kira and L. Rendell. A practical approach to feature selection. In *ICML*, pages 249–256, San Francisco, CA, USA, 1992.

[18] I. Guyon et al. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.

[19] A. Whitney. A direct method of nonparametric measurement selection. *IEEE Trans. Comput.*, 20:1100–1103, September 1971.

[20] C. Hsu et al. A practical guide to support vector classification. *Bioinformatics*, 1(1):1–16, 2010.