NuActiv: Recognizing Unseen New Activities Using Semantic Attribute-Based Learning

Heng-Tze Cheng, Feng-Tso Sun, Martin Griss
Electrical and Computer Engineering
Carnegie Mellon University
{hengtze, fengtso}@cmu.edu,
martin.griss@sv.cmu.edu

Paul Davis, Jianguo Li, Di You
Applied Research Center
Motorola Mobility
{pdavis, jianguo.li, di.you}
@motorola.com

ABSTRACT

We study the problem of how to recognize a new human activity when we have never seen any training example of that activity before. Recognizing human activities is an essential element for user-centric and context-aware applications. Previous studies showed promising results using various machine learning algorithms. However, most existing methods can only recognize the activities that were previously seen in the training data. A previously unseen activity class cannot be recognized if there were no training samples in the dataset. Even if all of the activities can be enumerated in advance, labeled samples are often time consuming and expensive to get, as they require huge effort from human annotators or experts.

In this paper, we present *NuActiv*, an activity recognition system that can recognize a human activity even when there are no training data for that activity class. Firstly, we designed a new representation of activities using semantic attributes, where each attribute is a human readable term that describes a basic element or an inherent characteristic of an activity. Secondly, based on this representation, a two-layer zero-shot learning algorithm is developed for activity recognition. Finally, to reinforce recognition accuracy using minimal user feedback, we developed an active learning algorithm for activity recognition. Our approach is evaluated on two datasets, including a 10-exercise-activity dataset we collected, and a public dataset of 34 daily life activities. Experimental results show that using semantic attribute-based learning, NuActiv can generalize knowledge to recognize unseen new activities. Our approach achieved up to 79% accuracy in unseen activity recognition.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Design Methodology—*Classifier design and evaluation*; C.3 [**Special-Purpose and Application-Based Systems**]: Real-time and embedded systems

MobiSys'13, June 25-28, 2013, Taipei, Taiwan

General Terms

Algorithms, Design, Experimentation, Performance

Keywords

Activity recognition, zero-shot learning, semantic attributes, active learning, machine learning, mobile sensing, wearable computing, context-aware computing

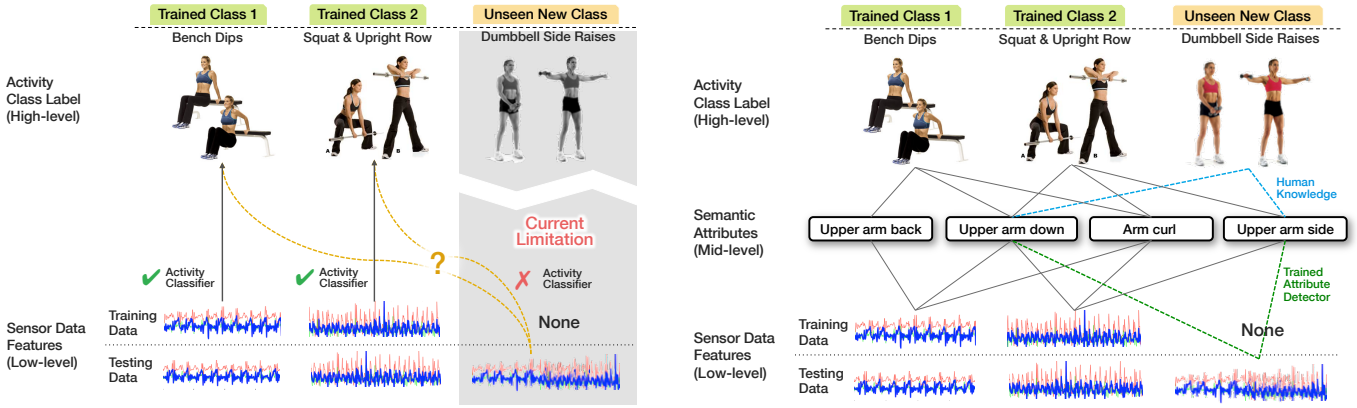
1. INTRODUCTION

The understanding of context and human activities is a core component that supports and enables various kinds of context-aware, user-centric mobile applications [10, 11, 14, 29]. Examples of application areas include user behavior modeling for marketing and advertising, health care and home monitoring, context-based personal assistants, context-enabled games, and social networks [14].

There has been extensive research on activity recognition using various sensors and various machine learning algorithms [2,18,21,25,40]. To recognize the context of a mobile phone user, most existing approaches require two steps: (1) collect and label a set of training data for every activity class that the system aims to detect, and (2) classify the current sensor readings into one of the pre-defined classes. However, labeled examples are often time consuming and expensive to obtain, as they require a lot of effort from test subjects, human annotators, or domain experts. Therefore, it has been reported that a fully supervised learning method, where labeled examples from different context are provided to the system, may not be practical for many applications [24,36,37]. More importantly, existing approaches to activity recognition cannot recognize a previously unseen new activity if there were no training samples of that activity in the dataset. According to the activity lexicon in the American Time Use Survey by U.S. Bureau of Labor Statistics [38], there are at least 462 different activities that people do in their daily lives. Considering the diversity of people and cultures that were not covered by the study, the actual number of activities is likely even larger. However, the fundamental problems in the existing activity recognition methods prevent the systems from recognizing any previously unseen activity and from extending to tens or hundreds of different human activity classes.

In light of existing problems, there are two major research questions we aimed to answer:

Q1. How to recognize a previously unseen new activity class when we have no training data from users?



(a) Existing supervised-learning approaches.

(b) Proposed semantic attribute-based learning.

Figure 1: Illustration of the comparison between existing supervised learning approach to activity recognition and the proposed semantic attribute-based learning approach.

Q2. If we have the opportunity to ask users for labeled training data, how to reinforce the recognition accuracy using minimal help from users?

In this paper, we present the *NuActiv* system to recognize human activity even when there are no training data for a particular activity class. NuActiv can generalize previously learned knowledge and extend its capability to recognize new activity classes. The design of NuActiv is inspired by the following observations:

- *Many human activities and context types share the same underlying semantic attributes*: For example, the attribute "`Sitting`" can be observed in the activities of both "having lunch in the cafeteria" and "working at a desk". Therefore, the statistical model of an attribute can be transferred from one activity to the another.
- *The limit of supervised learning can be overcome by incorporating human knowledge*: Rather than collecting sensor data and labels for every context, using nameable attributes allows humans to describe a context type even without the process of sensor data collection. For example, one can easily associate the activity "office working" with the motion-related attributes such as "`Sitting`," "`HandsOnTable`," and sound-related attributes such as "`PrinterSound`," "`KeyboardSound`," and "`Conversations`."

Based on these observations, we developed the NuActiv system to tackle the two research questions. The research question Q1 is often referred to as the *zero-shot learning* problem, where the goal is to learn a classifier that can recognize new classes that have never appeared in the training dataset [30]. While having been shown successful in the recent computer vision literature [13], zero-shot learning has been less studied in the area of human activity recognition.

There are several challenges when applying zero-shot learning to activity recognition. Firstly, while there exist some well-established attributes in the field of computer vision (such as shapes and colors), it has not been shown what kinds of representations or attributes are useful for recognizing human activities from sensor data. Secondly, most previous work on zero-shot learning focused on static image data, which is quite different from sequential sensor data in activity recognition.

To address these challenges, we designed a new representation of human activities by decomposing high-level activities into combinations of semantic attributes, where each attribute is a human readable term that describes a basic element or an intrinsic characteristic of an activity. The semantic attributes are detected based on the low-level features, which capture the temporal dynamics in the sequential sensor data. Using this representation, a two-layer attribute-based learning algorithm is developed for activity recognition. Figure 1 illustrates the difference between existing supervised-learning-based approach and our proposed semantic attribute-based learning approach, using exercise activities as examples.

For the research question Q2, to reinforce the activity recognition accuracy by leveraging user feedback, we extend the previous work in active learning [35] by designing an outlier-aware active learning algorithm and a hybrid stream/pool-based sampling scheme, which is suitable for the scenario of activity recognition using mobile or wearable devices. We integrated active learning in the framework of zero-shot learning for activity recognition, so that the system is able to not only recognize unseen activities but also actively request labels from users when possible.

The main contributions of the work include:

- The design and implementation of NuActiv, a new system to recognize human activity even when there are no training data for a particular activity class.
- The design and development of the semantic attributes as a representation for human activities, and the zero-shot learning algorithm to learn high-level human activities by decomposing them into combinations of semantic attributes.
- The design and development of the outlier-aware active learning algorithm to efficiently reinforce the recognition accuracy using minimal user feedback.
- The evaluation of the activity recognition system on two real-world human activity datasets, one in the daily life activity domain and the other in the exercise activity domain.

The paper is organized as follows. The system design and the algorithms are presented in Sections 2 and 3. We present

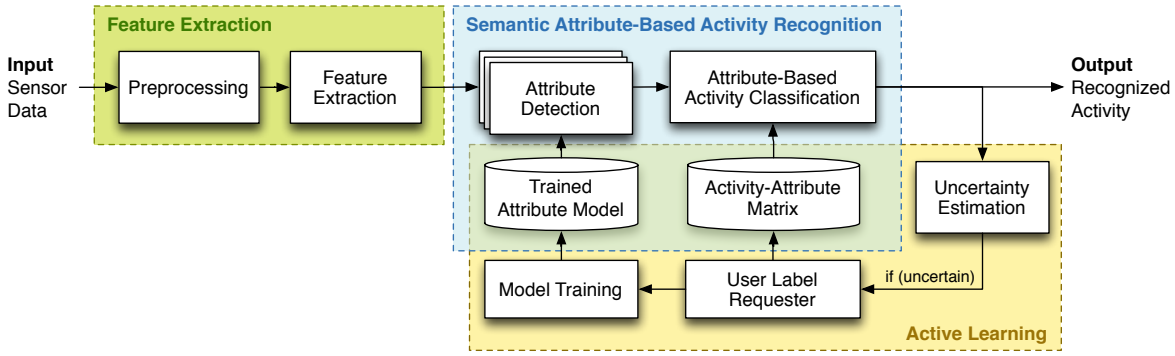


Figure 2: The architecture of the NuActiv activity recognition system.

the dataset collection, evaluation methodology, experimental results, and discussions in Section 4 and 5. In Section 6, we discuss and compare related work. The conclusion is presented in Section 7.

2. NUACTIV SYSTEM OVERVIEW

2.1 Scenarios and Design Considerations

NuActiv is designed for general human activity recognition in the field of mobile, wearable, and pervasive computing. The learning and recognition framework is independent of sensor data types or device types, so the source of sensor data does not limit to mobile phones but can also be wearable devices. Wearable mobile devices are becoming increasingly available in the commercial market [1]. Phones and devices can be worn as wrist watches (e.g. MotoACTV [27]), glasses (e.g. Google Glass [8]), and more. Advances in nanotechnology are further driving this trend by introducing flexible materials. These new wearable devices enable a wide range of context sensing, inference, and pervasive computing applications [14]. With these considerations in mind, in this work we choose phones and wristwatches with inertial sensors as examples to demonstrate two scenarios of activity domain: *daily life activities* and *exercise activities*.

The first scenario is daily life activity monitoring [9, 18]. Suppose we have the training data for two activities "`ReadingAtHome`" and "`Driving`". What if we want to detect if the user is "`ReadingOnTrain`"? Instead of hiring subjects to collect and label new sensor data, our goal is to directly recognize the new activity class "`ReadingOnTrain`" by reusing the model we already trained for "`ReadingAtHome`" and "`Driving`".

The second scenario is exercise activity detection. Detecting physical exercises and sports activities is useful for health and fitness monitoring applications [7, 28]. Through experiments on real-world sensor data, we will show that our semantic attribute-based learning applies well to this activity domain because many exercise activities are built up by the same underlying attributes, as illustrated in Figure 1.

Daily life activities are of bigger interest because they comprise the most part of people's lives. On the other hand, daily life activities are also arguably of much larger variation because different people do the same things differently. Even the same person can do one activity differently at different times. In this research, we started by testing our system and algorithms for the exercise activity scenario because the activities are well-defined, repeatable, and of lower variation among different people. After observing the effectiveness of our approach on exercise activities, we further generalized the approach to daily life activities.

2.2 System Architecture of NuActiv

The system architecture of NuActiv is shown in Figure 2. NuActiv consists of three main components:

(1) Feature Extraction: This component preprocesses the raw sensor data from various sensor inputs, and extract low-level signal features from the processed sensor data. (Section 3.1).

(2) Semantic Attribute-Based Activity Recognition: This component can be further divided into two part. The first part is *Attribute Detection*, which transform low-level features into a vector of human-readable semantic attributes. The second part is *Attribute-Based Activity Classification*, which classifies the detected attribute vector as one of the activity classes given the *activity-attribute matrix*, even if no training data exist for some of the target activity classes. (Section 3.2).

(3) Active Learning: Given the output recognized activity class, the active learning component estimates the uncertainty of the recognition result. Only when the result is estimated to be highly uncertain, the *user label requester* prompts the user for feedback or ground-truth labels. The labels are then used for re-training and updating models for attribute detection and activity classification. The function of this component is to reinforce activity recognition accuracy using minimal user feedback (Section 3.3).

All of the components can run on a mobile phone or sensor-enabled wristwatch in our system implementation. In cases where offline model training is needed, the attribute detection models can be pre-trained on a server and then be downloaded to a mobile device.

3. SYSTEM DESIGN AND ALGORITHMS

3.1 Feature Extraction

The NuActiv system is agnostic to input data type. Any kind of sensor data can be fed into the system for learning an activity recognition model. In this work, we select inertial sensor data from two activity domains—exercise activities and daily life activities—as examples to demonstrate the effectiveness of the system. The features we use include:

- The mean and standard deviation of sensor data in dimension x, y, and z.

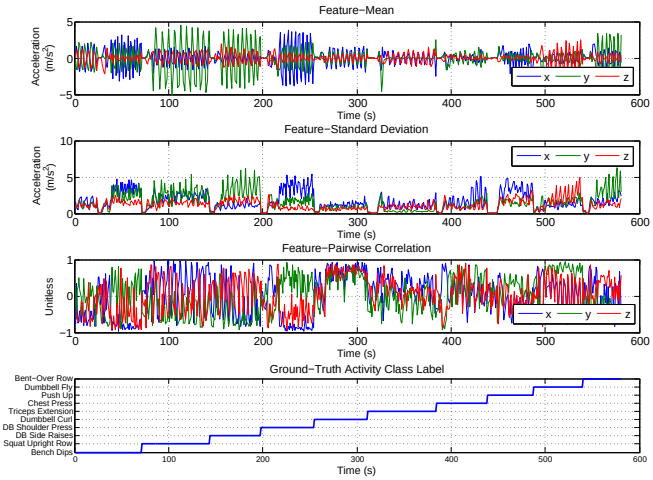


Figure 3: Examples of features extracted from acceleration data for each exercise activity.

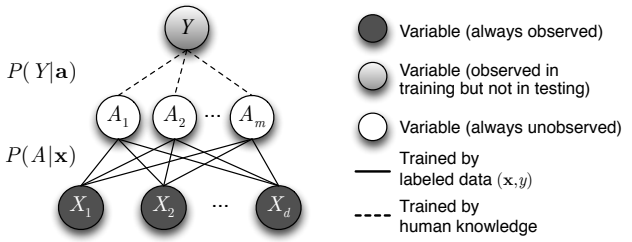


Figure 4: Graphical representation of semantic attribute-based activity recognition.

- Pairwise correlation between each pair of dimensions x, y, and z.
- Local slope of sensor data in dimension x, y, and z in using 1st-order linear regression.
- Zero-crossing rate in dimension x, y, and z.

Some examples of extracted features are shown in Figure 3. The sensor data and settings used in each dataset are described in Section 4.2. To capture the temporal changes of the features, we further include the n^{th}-order temporal features. Specifically, we concatenate the feature vector at time t with those at time $t-1, t-2, ..., t-n$ (n is empirically set to 2 using a 10-fold cross validation on the validation set in our experiments). For the daily life activity dataset [9], we also include time of day as an additional input feature, as it carries important information about the daily life routines of a user.

3.2 Semantic Attribute-Based Activity Recognition

3.2.1 Background and Problem Formulation

In this section, we describe the background and formulation of the problem of activity recognition with unseen classes. The formulation is inspired by and adapted from previous work in attribute-based classification and zero-shot learning [13, 30].

The problem of activity recognition is formalized as follows. Let Y be the class label, a random variable that can be one of the k classes in the activity class space $\mathcal{Y} = \{y_1, y_2, ..., y_k\}$. $\mathbf{x} = [X_1, X_2, ..., X_d]$ is a d-dimensional vector containing d input features in the feature space $\mathcal{X}$. We want to learn a classifier function $f : \mathcal{X} \rightarrow \mathcal{Y}$ where the function outputs an estimate or prediction of the most likely class label y given an input feature vector $\mathbf{x}$. Most of the existing approaches in activity recognition train the classifier f using a training dataset $D_{train} = \{(\mathbf{x}_i, y_i)|i = 1, 2, ..., N\}$, which contains N pairs of input features and ground-truth output class labels. If we have training instances for every class in $\mathcal{Y}$, we are able to train a classifier f. However, if there are no training data for a subset of classes $\mathcal{Y}$, we are not able to predict those classes.

In this work, we aim to solve the problem of recognizing previously unseen activity classes. Suppose $\mathcal{Y} = \{\{y_1, y_2, ..., y_s\}, \{y_{s+1}, ..., y_{s+u}\}\} = \mathcal{Y}_S \cup \mathcal{Y}_U$. $\mathcal{Y}_S$ is the set of *seen classes*, where there exists some training data for every class in $\mathcal{Y}_S$. $\mathcal{Y}_U$ is the *unseen classes* set where there are no training data for any class in $\mathcal{Y}_U$. The problem is: How to recognize an unseen class $y \in \mathcal{Y}_U$?

The idea is to first transform low-level features $\mathbf{x}$ into a vector of mid-level semantic attributes $\mathbf{a} = [A_1, A_2, ..., A_m]$ in the attribute space $\mathcal{A}$. Each attribute corresponds to an atomic physical motion or a specific characteristic of a complex activity. If every high-level activity in $\mathcal{Y}$ can be mapped to a point in the attribute space $\mathcal{A}$, then it is possible for us to recognize every activity class $y \in \mathcal{Y}$ given an accurately detected attribute vector $\mathbf{a}$. Since every semantic attribute in $\mathbf{a}$ is a human readable term, the mapping between y and $\mathbf{a}$ can be defined based on human knowledge without training data. Without this attribute layer, the direct mapping between y and $\mathbf{x}$ can only be trained with labeled sensor data, because the values in $\mathbf{x}$ low-level signal features that are hard for humans to interpret directly. The fundamental idea of semantic attribute-based learning is illustrated in Figure 4, where the edges represents $P(A|\mathbf{x})$, the probability of the attribute A given a feature vector $\mathbf{x}$, and $P(Y|\mathbf{a})$, the probability of the class label Y given an attribute vector $\mathbf{a}$. We will explain each step in details in the following sections.

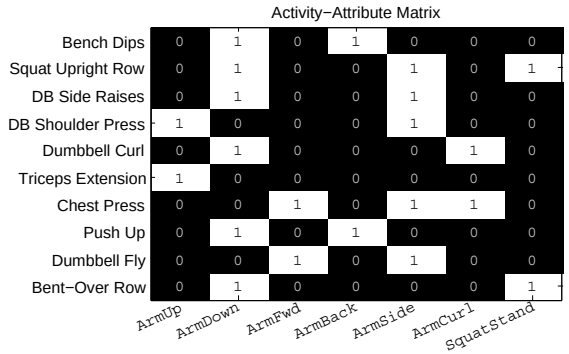


Figure 5: Activity-attribute matrix for exercise activities. The rows are the activities and the columns are the attributes.

3.2.2 Activity-Attribute Matrix

The Activity-Attribute Matrix encodes the human knowledge on the relationship between an activity and a set of semantic attributes that are associated with the activity. We designed the activity-attribute matrix by extending previous work in attribute-based object similarity and classification [12,13]. For M activities and N attributes, the activity-attribute matrix is an $M \times N$ matrix where the value of each

element a_{ij} represents the level of association between activity i and attribute j. We define each element as a binary value, indicating whether such an association exist ($a_{ij} = 1$) or not ($a_{ij} = 0$), although in general a_{ij} can be real-valued ($0 \leq a_{ij} \leq 1$), indicating the level or confidence of the association. An example of a binary activity-attribute matrix we manually defined for the exercise activity domain in our experiments is shown in Figure 5. In general, an activity-attribute matrix can be manually defined by common-sense knowledge or domain knowledge. Alternatively, the process can be automated using existing web text mining [31] or crowdsourcing platforms [34]. A user can also manually define a custom new activity by describing it using the attributes, which is equivalent to inserting a row in the matrix.

3.2.3 Attribute Detection

Given an activity-attribute matrix, the next step is to train a set of attribute detectors so that we are able to infer the presence/absence of an attribute from the sensor data features. However, collecting a separate training dataset for every attribute is not practical for several reasons. First of all, not all of the attributes are sub-activities themselves. Many attributes are descriptions, characteristics, or consequences of an activities rather than standalone sub-activities. Therefore, it may not be possible to collect data for an attribute "alone" without other interference or confounding factors. Furthermore, there can be a large number of possible attributes. If there were a need to collect many separate training dataset, the benefit of attribute-based learning would diminish significantly.

Since the goal is only to infer if an attribute is present or not given the feature vector (i.e. $P(A|\mathbf{x})$ in Figure 4), what we need is one set of positive samples and another set of negative samples. Therefore, to learn an attribute detector, we reuse the existing training data by merging the labeled data of all activity classes that are associated with the attribute as the positive set. Similarly, the negative set consists of the data of all activity classes that are not associated with the attribute.

After the training sets are constructed, a binary classifier is trained for each attribute. In general, any type of classifier can be used. We evaluated various classifiers and selected the Support Vector Machine (SVM) classifier [4] as the optimal implementation (the experiments are discussed in Section 4.4.3). SVM finds the hyperplane $\mathbf{w}^T\mathbf{x}_i + b = 0$ that maximizes the margin between the data points of different classes by optimizing the following Quadratic Programming problem:

$$\min_{\mathbf{w},b} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i \tag{1}$$

s.t. $a_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$, $\forall i$, where $\mathbf{x}_i$ and a_i are the feature vector and the attribute value for the i-th training sample, respectively. $\mathbf{w}$ and b controls the orientation and the offset of the hyperplane. The parameter C is a regularization term which controls overfitting and the tolerance on the degree of false classification ξ_i for each sample. After training phase, we have a trained attribute detector for each attribute specified in the activity-attribute matrix.

In some cases, we might only have positive or negative examples for an attribute. For example, this can happen when all of the seen classes in the training data exhibit a certain attribute. In such cases, we train the attribute detector using one-class SVM [23], which classifies a sample as one of the two classes given only training data of one class (positive or negative).

Algorithm 1 Hybrid Feature/Attribute-Based Activity Recognition Algorithm

```
Input: low-level feature vector x
Output: estimated activity class y
isUnseenClass ← unseenClassDetection(x);
if isUnseenClass = true then
    Keep only unseen classes in the attribute space;
    a ← attributeDetector(x);
    y ← attributeBasedActivityClassifier(a);
else
    Keep only seen classes in the feature space;
    y ← featureBasedActivityClassifier(x);
end if
return y;
```

3.2.4 Attribute-Based Activity Classification

After the attributes are detected, in the attribute space, a nearest-neighbor classifier is used to recognize the high-level activity given an attribute vector generated from attribute detectors [30]. Specifically, the activity recognizer takes an attribute vector $\mathbf{a} = [A_1, A_2, ..., A_m]$ as input and returns the closest high-level activity y^* represented in the attribute space $\mathcal{A}$ according to the activity-attribute matrix. In other words, the activity-attribute matrix essentially provides the information of $P(Y|\mathbf{a})$ shown in Figure 4.

3.2.5 Hybrid Feature/Attribute-Based Activity Recognition

While attributes are human readable and can be used to recognize previously unseen new classes, there are certain amounts of information in the low-level feature space that we do not want to discard. Transforming low-level features to mid-level attributes has the benefit for unseen class recognition, but there is an information loss associated with it.

Inspired by this thought, our idea is to keep the advantages of both feature-based and attribute-based activity recognition. Specifically, if we know that a sample belongs to a seen class where we had training data in the dataset, we can directly apply feature-based classifier to recognize the activity. On the other hand, if we think that a sample belongs to a new class that we have not had any training data associated with, we have to apply attribute-based activity recognition so that we can learn by reusing the known attributes.

Now the question is: How do we know if a sample belongs to a seen class or an unseen class? We draw an analogy between this problem and the problem of anomaly detection. A sample from a seen class is like a typical sample, which is similar to the other samples we had in the training data. In comparison, a sample from an unseen class is like an "anomaly" because it does not look like anything that the system has seen before. To approach this problem, we first train an unseen class detector using the one-class SVM classifier [6], where only the positive samples (all samples that belong to the seen classes) are given to the classifier. After using the unseen class detector, we then do a hybrid feature/attribute-based activity recognition using the algorithm described in Algorithm 1.

3.3 Active Learning: Reinforcing Activity Recognition Using Minimal User Feedback

So far we have focused on the scenario where no training data for the target class are available. What if we have the opportunity to acquire some ground-truth labeled data from users? Obviously, if we ask users to label every single sample, we can achieve the best recognition accuracy possible. However, it is impractical to ask a user to label his/her activity every single minute because it would be extremely intrusive. The more frequently we prompt the users for inputs, the more intrusive the system will be [35]. This observation motivates us to design a user feedback loop for the NuActiv system using active learning algorithms [35,37].

Our idea is simple: We ask a user for labels only when we are highly uncertain about our recognition result. To achieve this, we used the idea of uncertainty sampling in the field of active learning. The idea of active learning algorithms is that a machine learning algorithm can perform better with less training data if it is allowed to choose the data from which it learns [35].

3.3.1 Sampling Scheme

There are two types of selective sampling schemes in active learning [35]. The first one is *Stream-Based Sampling*, where an unlabeled instance is typically drawn one at a time from the input source, and the system must decide whether to query or discard it. The second scheme is *Pool-Based Sampling*, where a large pool of unlabeled data is available. Having observed all the unlabeled instances, the system can ask for the label of one instance at a time according to a certain decision criteria.

In this work, we use a hybrid stream/pool-based sampling scheme that is more suitable for the scenario of human activity recognition using mobile phones or wearable devices. The pool is not so big so that a user forgets what he/she did during the time interval asked by the system, yet large enough for the system to select a good sample to ask the user for a ground-truth label. The detailed settings are described in Section 4.6.

3.3.2 Uncertainty Sampling Metrics

In this work, we carefully evaluated several different widely used metrics [35, 37] that measure the uncertainty of a sample to the classifier in order to seek an optimal solution:

Least Confident: Ask the user for a ground-truth label when the confidence score of the classifier output $\hat{y}$ given the input feature x of a sample is minimum:

$$x^*_{LC} = \underset{x}{\operatorname{argmin}} P_\theta(\hat{y}|x) \tag{2}$$

Minimum Margin: Ask the user for a ground-truth label when the difference between the confidence of the first and the second likely classes ($\hat{y}_1$ and $\hat{y}_2$) is small:

$$x^*_M = \underset{x}{\operatorname{argmin}} \left[P_\theta(\hat{y}_1|x) - P_\theta(\hat{y}_2|x)\right] \tag{3}$$

Maximum Entropy: Entropy, in information theory, is measure of the uncertainty associated with a random variable. $H_\theta(Y|x) = -\sum_y P_\theta(y|x) \log P_\theta(y|x)$ means that given a sample x and classifier model θ, how uncertain the classifier is about the value of class label Y. Therefore, we can ask the user for a ground-truth label when the entropy over Y given a specific sample x is the largest among all x in consideration:

$$x^*_H = \underset{x}{\operatorname{argmax}} -\sum_y P_\theta(y|x) \log P_\theta(y|x) \tag{4}$$

The comparison between the performances using different metrics is reported in Section 4.

Algorithm 2 Outlier-Aware Uncertainty-Sampling Active Learning Algorithm for Activity Recognition

Input: A sequence of initial unlabeled instances $\mathcal{U} = \{x_i|i = 1, ..., N_\mathcal{U}\}$; A set of initial labeled instances $\mathcal{L} = \{(x_i, y_i)|i = 1, ..., N_\mathcal{L}\}$; An initial classifier model θ; A pool window length L_{pwin}
Output: Updated activity classifier model θ
/* $N_\mathcal{U}$: the number of unlabeled samples available in the pool window */
while Activity Recognition Service is running **do**
  while $N_\mathcal{U} < L_{pwin}$ **do**
    $d \leftarrow$ getCurrentSensorData();
    $x \leftarrow$ extractFeatures(d);
    insert x into $\mathcal{U}$;
    $N_\mathcal{U} \leftarrow N_\mathcal{U} + 1$;
  end while
  $maxScore \leftarrow -\infty$; $x^* \leftarrow x_1$;
  for i **from** 1 **to** L_{pwin} **do**
    $score \leftarrow$ getOutlierAwareUncertainty(x_i);
    if $score > maxScore$ **then**
      $maxScore \leftarrow score$;
      $x^* \leftarrow x_i$
    end if
  end for
  $y^* \leftarrow$ queryForLabel(x);
  insert (x^*, y^*) to $\mathcal{L}$;
  $\theta \leftarrow$ trainClassifier($\mathcal{L}$);
  Remove all samples in pool $\mathcal{U}$; $N_\mathcal{U} \leftarrow 0$;
end while
return θ;

3.3.3 Outlier-Aware Uncertainty Sampling

Using uncertainty sampling, however, can run the risk of choosing outliers as samples to query [35]. The reason is that outliers are away from the other samples of the same class in the feature space; therefore, for most uncertainty metrics we use, outliers are likely to receive higher uncertainty scores than other samples. Unfortunately, knowing the label of outliers does not help training a classifier because outliers are exceptions rather than representative examples that a classifier should learn from. As a result, actively choosing outliers for training can even "mislead" the classifier and end up degrading the accuracy.

To mitigate the negative affect of outliers, we used Outlier-Aware Uncertainty Sampling in tandem with the uncertainty sampling metrics. The idea is to select samples that are uncertain but not outliers, i.e., samples that are representative of the underlying distribution (e.g. in dense region of the feature space). To determine whether a sample is representative of the underlying distribution, we calculate the mean similarity between this sample and all the other samples. If a sample is close to many other samples in the feature space, its mean similarity with all the other samples will be high; on the other hand, for an outlier that is far from most samples, the mean similarity will be low. Incorporat-

ing this constraint into the uncertainty sampling metric, the new objective function is:

$$x^*_{OA} = \underset{x \in \mathcal{U}}{\operatorname{argmax}} \left(\phi(x) \cdot \frac{1}{N_{\mathcal{U}}} \sum_{x' \in \mathcal{U}} S(x, x') \right) \quad (5)$$

The first term $\phi(x)$ refers to one of the uncertainty metrics we described in Section 3.3.2. To be consistent with the argmax objective, for Lease Confident uncertainty metric, $\phi(x)$ is defined as $\exp(-P_\theta(\hat{y}|x))$. Similarly, $\phi(x) = \exp(-(P_\theta(\hat{y}_1|x) - P_\theta(\hat{y}_2|x)))$ for minimum margin metric, and $\phi(x) = H_\theta(Y|x)$ for the maximum entropy metric. The second term $\frac{1}{N_{\mathcal{U}}} \sum_{x' \in \mathcal{U}} S(x, x')$ measures the mean similarity, $S(x, x')$, between a sample x and all other samples x' in the unlabeled sample pool $\mathcal{U}$, where $N_{\mathcal{U}}$ is the total number of samples in $\mathcal{U}$. The complete algorithm is shown in Algorithm 2.

4. EVALUATION

To evaluate our approaches, we investigated and answered the following questions through system implementation, dataset collection, and experiments:

- What is the overall precision/recall of unseen activity recognition using NuActiv? How does the performance vary among classes? (Section 4.4.1)
- How does the recognition accuracy change with the number of unseen classes? (Section 4.4.2)
- How does the performance vary with the use of different classification algorithms for attribute detectors? (Section 4.4.3)
- How to select attributes based on their importance to unseen activity recognition? (Section 4.4.4)
- What is the cross-user performance, i.e. when the users in the training set are different from those in the testing set? Is the system able to generalize from one or a few users to many new users? (Section 4.4.5)
- How does the attribute detection accuracy vary with the position and combination of the devices and sensors? (Section 4.4.6)
- How does NuActiv perform on recognizing unseen daily life activities? (Section 4.5.2)
- How efficiently can the system reinforce its performance using active learning? How does the performance vary with different active learning algorithms? (Section 4.6.1)
- What is the effect of outlier-aware uncertainty sampling on active learning algorithms? (Section 4.6.2)

4.1 System Implementation

We have implemented and tested the system on Nexus S 4G phones and MotoACTV wristwatches [27]. A picture of our system running on these two types of devices is shown in Figure 6. The Nexus S 4G phone has a three-dimensional accelerometer and a gyroscope. The MotoACTV has a three-dimensional accelerometer, Wi-Fi, and Bluetooth radio.

For the software part, we have implemented the code for feature extraction and the classification algorithm in the Java programming language. The code runs on the Android Operating System installed on the Nexus S 4G phones and the MotoACTV. For the classification algorithm, the Support Vector Machine classifier is implemented using the

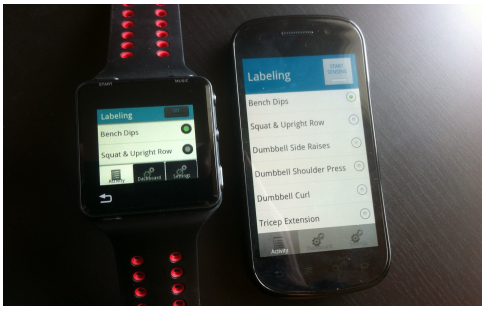


Figure 6: NuActiv running on MotoACTV wristwatch (left) and Nexus S 4G phone (right).

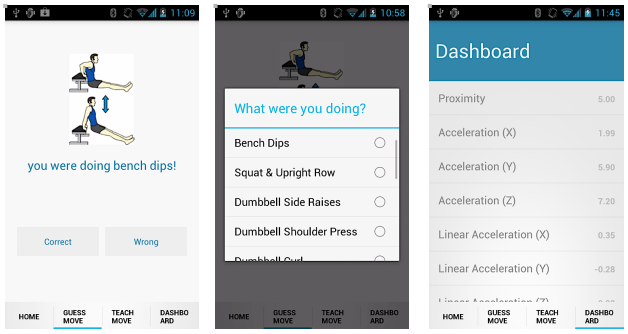


Figure 7: The screenshots of our mobile app running NuActiv activity recognition system.

LibSVM library [6]. The screenshots of several operations of the application are shown in Figure 7.

4.2 Datasets

4.2.1 *Exercise Activity Dataset*

We conducted an experiment involving exercise activities on 20 subjects. Each subject is asked to perform a set of 10 exercise activities as listed in Figure 5 with 10 iterations. Before the experiments, the subjects are given instructions on how to perform each of the 10 exercise activities. More information about these activities can be found in the literature [7,28]. During the experiment, each subject is equipped with three sensor-enabled devices: A Nexus S 4G phones attached to the arm using an armband, a MotoACTV wristwatch, and a second MotoACTV unit fixed at the hip position using a clip. A pair of three-pound dumbbells is also provided to the subject to perform some of the free-weight exercises (e.g. Dumbbell Side Raises, Dumbbell Curl, etc.). For sensor data collection, we collected accelerometer and gyroscope data using our mobile application with a sampling rate of 30 Hz. For feature extraction, the sliding window size is empirically set to 1 second with 50% overlap, based on a 10-fold cross-validation test on the validation dataset to find the optimal parameter.

4.2.2 *Public Dataset on Daily-Life Activities*

For the scenario of recognizing daily-life activities, we use a published and publicly-available dataset collected by Technische Universitat Darmstadt (TU Darmstadt) [9,36]. The dataset includes 34 daily life activity classes (including the *unlabeled* class) collected from one subject for seven days.

Table 1: Attribute list for daily life activities.

Type	Attribute Name
Basic	`Sitting`, `Standing`, `Walking`
Posture	`PostureUpright`, `PostureKneeling`
Hand/ Arm	`HandsOnTable`, `HandAboveChest`, `WristMovement`, `ArmPendulumSwing`
Motion Type	`TranslationMotion`, `CyclicMotion`, `IntenseMotion`
Relation	`WashingRelated`, `MealRelated`
Time	`TimeMorning`, `TimeNoon`, `TimeEvening`

The sensor data were collected using a wearable sensor platform with a three-axis accelerometer (ADXL330) worn on the wrist and the hip of the subject. The sampling rate is 100Hz, and the features are computed from a sliding window of 30 seconds with 50% overlap.

To apply NuActiv to the TU Darmstadt daily-life activity dataset, we defined a list of 17 attributes (as shown in Table 1) and an activity-attribute matrix[1] based on the 34 daily life activities in the dataset. It is to be noted that the list is not mutually exclusive or collectively exhaustive. We show that these semantic attributes defined by human knowledge can enable unseen activity recognition using NuActiv in Section 4.5.

4.3 Evaluation Methodology

We used leave-two-class-out cross validation, the most widely used validation method used in the literature of zero-shot/zero-data learning [15, 30]. The validation scheme is used for recognizing unseen classes that do not have any sample in the training set. The traditional 10-fold cross validation is not applicable to unseen class recognition because it does not leave out all samples of certain "unseen" classes in the training step, so that every class will have some samples in the training set.

The leave-two-class-out cross validation works as follows. Suppose there are a total of N classes. Each time we first train our system on $(N-2)$ classes, and then test the discriminative capability of the classifier on the remaining 2 classes that were "unseen" by the system during the training process. We repeat this test for all $\binom{N}{2}$ unseen/seen class combinations. Finally, the average performance over all tests is reported.

The results are reported in precision, recall, and F1-score. These metrics show different aspects of the performance of an activity recognition system. Specifically, the metrics are defined as follows:

$$precision = \frac{TP}{TP+FP} \qquad recall = \frac{TP}{TP+FN} \tag{6}$$

$$F_1\text{-}score = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{7}$$

where TP, FP, TN, and FN denotes true positive, false positive, true negative, and false negative, respectively. Precision indicates the percentage of times that a recognition result made by the system is correct. Recall means the percentage of times that an activity performed by a user is detected by the system. F_1-*score* is a integrated measure that combines both. For overall performance across all classes, the *accuracy* is computed as the number of correctly recognized samples divided by the number of all samples in the test set.

[1]The activity-attribute matrix can be downloaded from the supplemental materials at `http://www.ece.cmu.edu/~hengtzec/data/DLActivityAttributeMatrix.pdf`

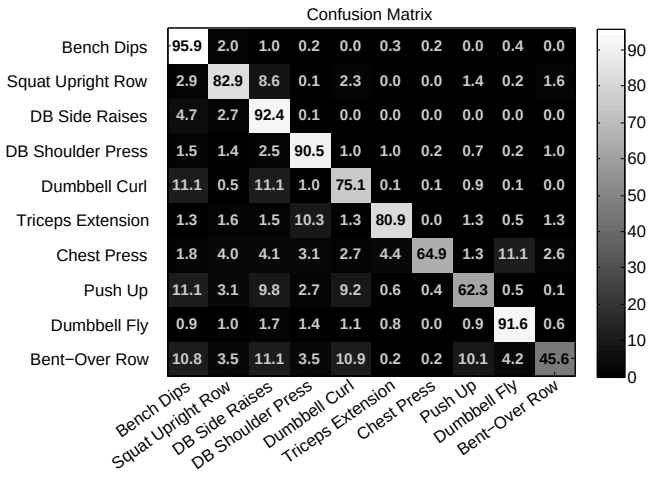


Figure 8: Confusion matrix of recognizing unseen activities using the 10-Exercise dataset. The rows are the ground-truth activity classes, and the columns are the estimated activity classes (classifier outputs). The numbers are shown as percentages.

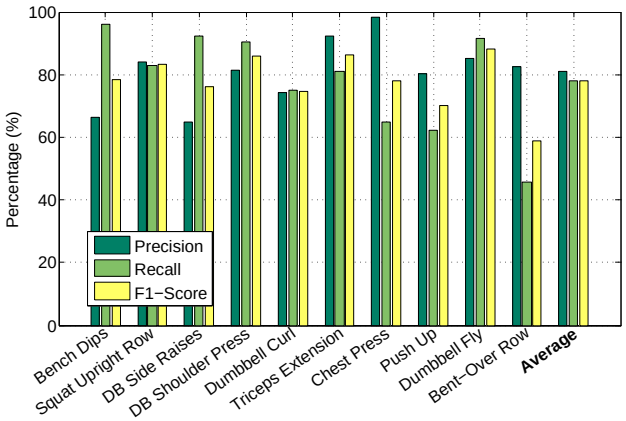


Figure 9: Precision and recall rate of recognizing unseen activities using the 10-Exercise dataset.

4.4 Case Study I: Exercise Activities

4.4.1 Unseen Activity Recognition Result

The confusion matrix of recognizing previously unseen exercise activities is shown in Figure 8. The average accuracy is 79% over all activities, among which the system achieved a promising recognition accuracy of 80-90% for five activity classes. It is to be noted that in these results, the target activities are recognized under the situation that no training data of any target activity class were given to or seen by the system during the training phase. The results support our hypothesis that unseen new human activities can be recognized with a reasonable accuracy using the proposed semantic-attribute-based learning approach in NuActiv.

One observation that we can draw from the experimental result is that misclassification usually happens when two activities only differ in one attribute. In this case, the success of recognition depends heavily on the detection accuracy

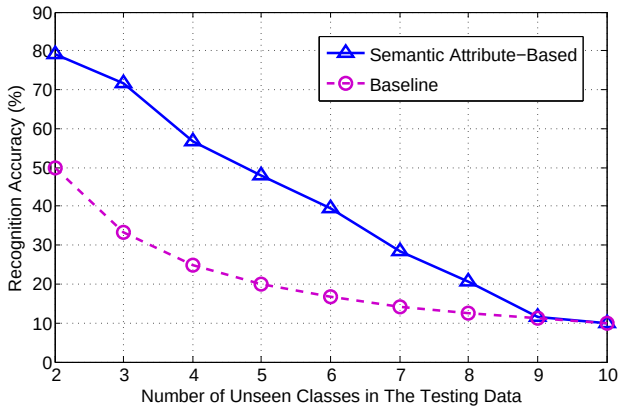


Figure 10: Accuracy vs. number of unseen classes in the testing dataset.

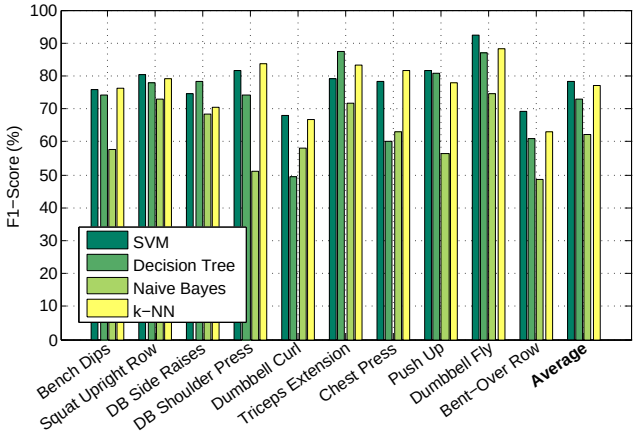


Figure 11: F1-score of unseen activity recognition vs. different classifiers for attribute detectors.

of the presence of that particular attribute. For example, "`DumbbellFly`" is classified as "`ChestPress`" because these two activities are inherently similar and are only different in the attribute "`ArmCurl`". The problem can potentially be overcome by including additional sensors/modalities so that other discriminative attributes can be used to further distinguish two similar classes.

4.4.2 The Impact of Number of Unseen Classes And Comparison with Baseline

The capability to recognize unseen new activity classes is built on the knowledge learned from the seen activity classes. As we can imagine, if all the classes were unseen, the system has nothing to learn from and thus is not able to recognize any activity with reasonable accuracy. To understand the capability and limitation of our approach, we conducted the following experiment: For a total of k classes, we vary the number of unseen classes (n_u) in the testing data from 2 to k, where the corresponding number of seen classes ($n_s = k - n_u$) in the training data varies from $k-2$ to 0. For each number of unseen classes, we repeat the test for all $\binom{k}{n_u}$ combinations and report the average results.

The result is shown in Figure 10. We observe that the recognition accuracy gradually degrades as the number of unseen classes in the testing data increases (i.e. the number of seen classes in the training data decreases). This is in accordance with the expectation, because it gradually becomes difficult for the system to generalize to a large number of unseen activity classes based on only a few seen classes. Furthermore, a successful unseen activity recognition relies on an accurate attribute detection. To accurately detect an attribute, it is important for the training classes to cover both positive and negative examples of the attribute. Therefore, the larger the seen-to-unseen class ratio is, the more likely that we can recognize unseen activities effectively.

We also compare our results with a baseline approach. The baseline approach is the random-guess prediction given the number of unseen classes, which is the best that a supervised learning-based activity recognition system can do. From Figure 10, we can see that our semantic-attribute-based approach is 20-30% better than the baseline for most cases, except for the cases where almost all the classes were unseen (when the system has seen zero to two classes in the training data). The results suggest that NuActiv is an viable approach to unseen activity recognition.

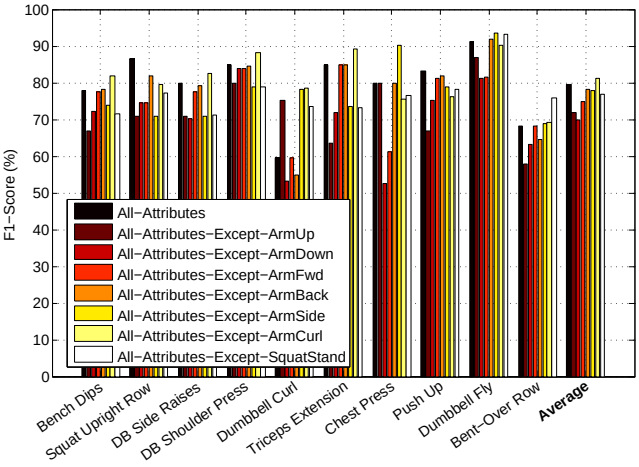


Figure 12: F1-score vs selected attributes. Each color represent an attribute that was unselected.

4.4.3 Comparison of Different Attribute Detectors

We also compare the SVM classifier with other classifiers that are widely used in the related work, including the Decision Tree classifier, Naive Bayes classifier, and k-Nearest Neighbor (k-NN) classifier [2, 18, 20]. For k-NN, the optimal result with $k = 3$ is reported.

The results are shown in Figure 11. SVM outperforms the Decision Tree and Naive Bayes classifier on average and for most of the classes if we break down the results by activity class. Overall, the accuracy using k-NN is comparable to the result using SVM. However, k-NN classification requires the storage and access to all the training data. Thus, k-NN is less scalable for a large dataset and less practical to run on mobile devices given their limited storage. Therefore, we used SVM for our experiments and implementation on the mobile devices.

4.4.4 Evaluation of The Importance of Attributes

We now investigate the importance of each semantic attribute and gain insights into attribute selection. The selection of attributes is important in two aspects. The first one is *discriminability*, meaning how well can an attribute discriminate between different high-level classes. The second

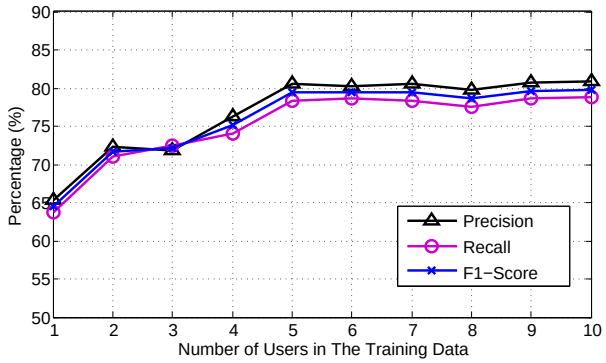


Figure 13: Cross-user recognition accuracy vs. number of seen users in the training data. The testing set includes 10 users that are different from those in the training data.

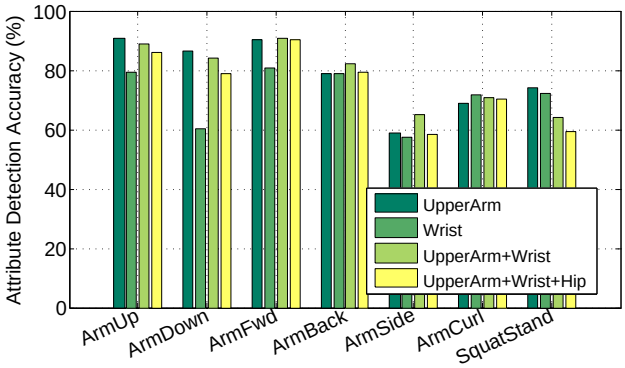


Figure 14: Attribute detection accuracy vs. device/sensor positions for each attribute.

one is *detectability*, namely how accurately can we detect the presence or absence of an attribute.

To test the *discriminability*, we conducted the following experiment: First, we run the unseen activity recognition test using all n_A semantic attributes ($n_A = 7$ for exercise activities). Then, we run n_A tests where each time we exclude one of the attributes and observe the change of the performance. If the performance drops significantly when an attribute is excluded, then being able to detect this attribute is important to accurately recognize the high-level activities. On the other hand, if the performance does not change much without an attribute, then the attribute is less likely to be important. To test the *detectability*, we compute the detection accuracy of each attribute. The accuracy is computed as the number of times an attribute is correctly detected divided by the number of times an attribute appears in the samples of the testing data.

The results of the discriminability and detectability test are shown in Figure 12 and 14, respectively. From the average F1-score (the rightmost bar group) in Figure 12, we can see that the attributes `ArmUp`, `ArmDown`, and `ArmFwd` have a higher impact on the activity recognition performance than other attributes. However, from the results broken down by activity class, we can see that an attribute may be important to some activity classes but not for other classes. Therefore, the selection of attributes also depends on the characteristics of the targeted classes. One reason for these phenomena is the inherent attribute-composition of an activity. Another possible reason is that some attributes are easier to detect than others. As shown in Figure 14, the system generally achieves higher accuracy detecting the first four attributes. These differences in detectability can be caused by the positions and types of the sensors used, the type of classifier used for detection, and the consistency of the presence of an attribute given an activity is performed.

4.4.5 Cross-User Activity Recognition Results

It is important for an activity recognition system to not only be able to recognize the activities of the users it has seen, but also be able to generalize to the activities of new users it has never seen before. To evaluate the generalizability and the limitation of our approach, we randomly divide the 20 users into two equal sets. While fixing the test set to be the 10 users in the first held-out set, we iterate the number of training users from 1 to 10, randomly chosen from the second set. For each test, we repeat the random choice for 100 times and report the average performance.

The results are shown in Figure 13. As we can see, the performance stays approximately constant when the number of seen users in the training data is equal to or greater than five. Furthermore, the precision, recall, and F1-score are almost the same as the case where the data of all the users exist in both the training set and the testing set, as shown previously in Figure 9. The F1-score decreases slightly when number of seen users in the training data falls below four, yet the system can maintain an F1-score of over 70% when having seen 2–4 users in the training data. The edge case happens when the system has only seen one user in the training set, where the F1-score is 60%. Overall, the system is able to achieve 70-80% accuracy after training on two or more users in the training set.

4.4.6 Impact of Device Position on Attribute Detection Accuracy

An attribute is often inherently associated with a characteristic of a human activity or a motion of a specific part of human body. Therefore, we conducted experiments to understand how the position or the set of positions at which the sensors are placed affects the attribute detection accuracy, which in turns affects the final activity recognition accuracy. When collecting the exercise activity dataset, we have placed sensor-enabled phones/devices on three different body positions of the users (as described in Section 4.2.1). The experimental results using sensor data from different body positions is shown in Figure 14. It is observed that while using the upper arm sensors (phone in an armband) usually achieves better and more stable accuracy than using the wrist sensors (wristwatch), combining these two data sources leads to improvement in accuracy in some cases. One possible reason is that while the movement of the wrist is of larger variation and less predictable, it complements the limitation of what can be observed by the upper arm sensors. Adding the hip sensor did not improve the accuracy, possibly because most attributes defined in our case study do not involve changes in lower-body postures. It is to be noted that the results do depend on the activity domains and sensor types.

4.5 Case Study II: Daily-Life Activities

4.5.1 Reproduction of Results in Previous Work

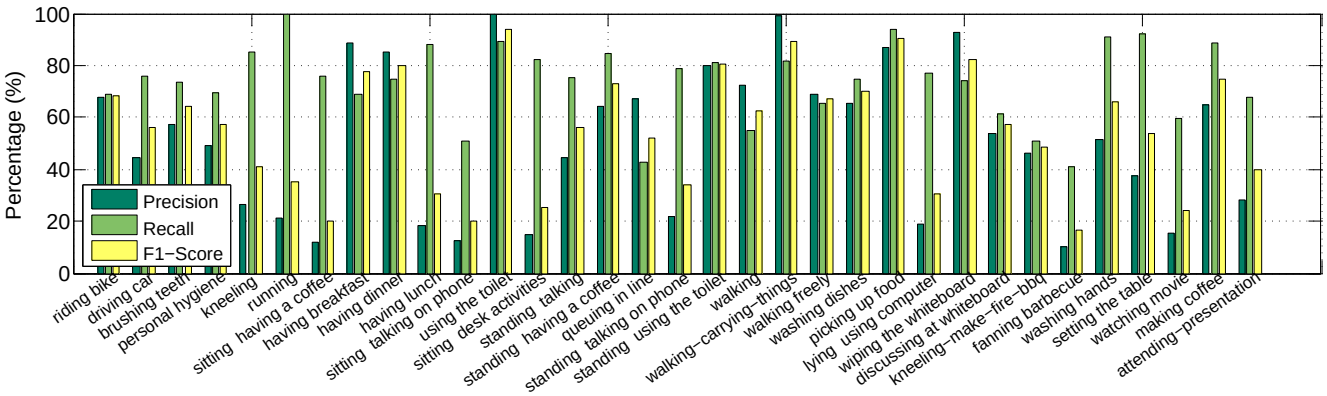


Figure 15: Precision, recall, and F1-score of recognizing unseen daily life activities in the TU Darmstadt dataset using NuActiv.

For the daily life activity recognition experiments, we used the public dataset of 34 daily life activities provided by TU Darmstadt [9], as described in Section 4.2.2. We first implement the supervised learning method closely following the experiment protocols in the paper of the dataset provider [9]. Given that all the classes were seen in the training set, our implementation achieved 71.2% accuracy, which is very close to 72.7% as reported in the previous work [9]. This reproduction of previous results confirms that our use and understanding of the dataset and features are valid.

4.5.2 *New Task: Recognizing Previously Unseen New Daily Life Activity*

After successfully reproducing the results in the previous work, we proceeded to a new problem—recognizing unseen daily life activities—which has not been addressed before in the activity recognition literature. We applied the NuActiv system and algorithms to the 34-daily-life dataset [9], and evaluated the performance using the evaluation methodology described in Section 4.3. The results are shown in Figure 15. We can see that for some classes the system can achieve high recall and lower precision, and vice versa for some classes. Overall, the system achieves 60-70% precision and recall rate for most classes. The mean precision and recall rate is 52.3% and 73.4%, respectively, averaged over all classes. Some classes, such as "sitting-desk-activities" or "sitting-talking-on-phone", do not have a clear difference in attributes since we only have inertial sensor data available in the dataset. Therefore, the system tends to have a low precision rate on these classes. This problem can be alleviated by incorporating extra sensing modalities such as ambient sound. While there is clearly room for improvement, the fact that our system is able to recognize an unseen daily life activity class with no training data with a reasonable accuracy is a new result in the field of activity recognition.

4.6 Active Learning Experiments

4.6.1 *Comparison of Active Learning Algorithms*

In addition to unseen activity recognition, we further evaluate how the system can improve itself using minimal user feedback. The active learning algorithms we used for the experiment are explained in Section 3.3.

Following the experiment protocols in the active learning literatures [35], our experiment setting is described as follows. The data set we used is the 34-daily-life dataset [9]. Each sample is a feature vector extracted from a window of 30 seconds of sensor data. An initial labeled set $\mathcal{L}$ of 50 samples from Day 1 of the dataset is provided to the system. Then, an unlabeled set $\mathcal{U}$ of 11087 samples from the rest of Day 1 to Day 5 is sequentially provided to the system in the order of time. Each time the system improves itself using the newly acquired labeled data, we evaluate its performance on a separate test set $\mathcal{T}$ of 5951 samples from Day 6 and 7. The active learning is performed as described in Algorithm 2, with $L_{pwin} = 100$ and SVM classifiers.

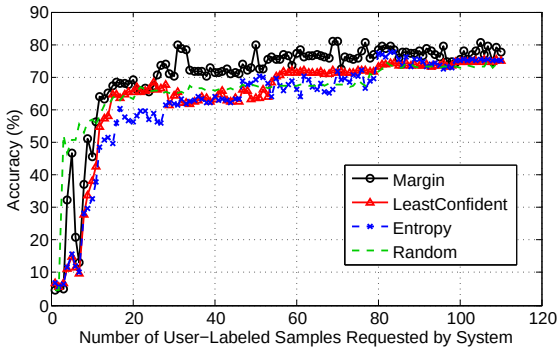
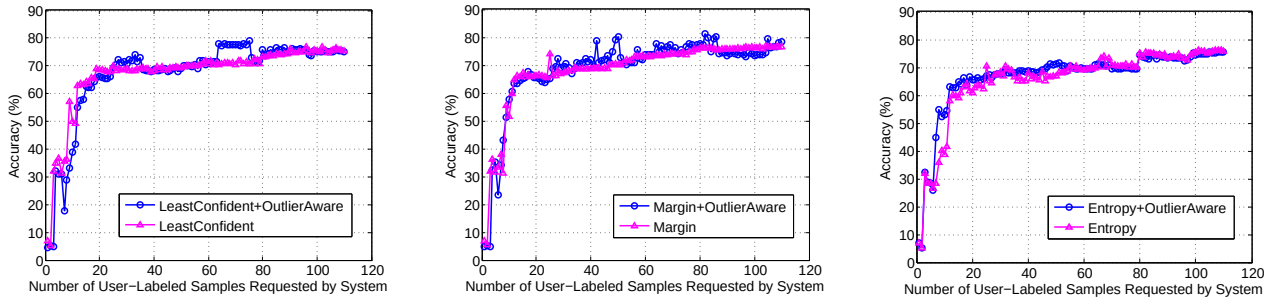


Figure 16: Recognition accuracy vs. user labels requested in the active learning experiment.

The results are shown in Figure 16. Using active learning, the classifier generally improves faster (using less labeled samples from the user) than the *random* baseline (randomly query the user without active learning). The margin-based uncertainty metric achieved 70% accuracy using only 30 labeled samples from the user and converged faster than other approaches. The entropy and least-confident metrics yielded comparable results.

4.6.2 *Outlier-Aware Uncertainty Sampling Results*

We further incorporate the outlier-aware uncertainty sampling as described in Section 3.3.3, and compare the results with those not using outlier-aware uncertainty sampling. The results are shown in Figure 17. It is observed that given the same amount of user-labeled samples requested by the system, using outlier-aware uncertainty sampling in general leads to comparable or better accuracy when compared to active learning algorithms without outlier-awareness. The



(a) Least-confidence-based sampling. (b) Margin-based sampling. (c) Entropy-based sampling.

Figure 17: Comparison between the learning curve of active learning algorithms with/without outlier-aware uncertainty sampling .

amount of improvement, of course, would depend on the relative weighting between the uncertainty term and the outlier-awareness term in Equation 5, and on whether there is a large number of outlier samples in the input data.

5. DISCUSSION

In this section, we discuss some assumptions and limitations of the current version of NuActiv, along with our ongoing work and potential future directions.

5.1 Semantic Attribute-Based Learning

In the current version of our semantic attribute-based learning approach, it is assumed that there exists a one-to-one mapping between an activity class label and a point in the attribute space, and that the associations between activities and attributes are fixed. This implies a fundamental limitation that the lower bound on the minimum number of attributes is $n_A \geq \log_2 n_Y$ for n_Y activity classes and n_A different attributes, assuming binary-valued attributes are used. As a future research direction, it is possible to overcome this limitation by incorporating continuous-valued attributes or relative attributes [32]. Further more, while we present an initial attempt to evaluate the attribute-based learning approach on two datasets, it would be beneficial to expand the study to more activity domains with a larger number of activities, attributes, and users in the future.

In the current implementation, the attributes are manually defined using common-sense knowledge and domain knowledge as an initial attempt towards zero-shot learning for activity recognition. To further reduce the effort of one-time manual definition per class, a potential future direction and part of our ongoing work is to automate the process using web text mining [31] or crowdsourcing [34] as explored in the zero-shot learning literature.

Our results suggest that the performance of the zero-shot learning model varies depending on the selected semantic attributes. Therefore, another future direction is to develop a systematic way for semantic attribute selection based on the discriminability and detectability of the attributes. Further, to truly exploit the advantages of both low-level features and mid-level attributes, future work and experiments are to be done to explore and compare various kinds of algorithms for hybrid feature/attribute-based activity recognition.

5.2 Active Learning for Activity Recognition

For the active learning experiment, it is assumed that the users are willing to provide the label and all the labels provided by the user are correct. Related study or future work on usability and interruptibility [33] can be further leveraged to adjust the frequency of requesting labels from users based on their preferences, and to improve the effectiveness of active learning in real practice. It would also be beneficial to study the ideal way (e.g. haptic, gestural, or audio-based interfaces) to engage users to provide labeled data for activity recognition using wearable and mobile devices.

6. RELATED WORK

6.1 Activity Recognition

6.1.1 *Supervised Learning*

In the field of mobile, wearable, and pervasive computing, extensive research has been done to recognize human activities (e.g. sitting, walking, running) [2, 5, 14, 21, 25, 36, 37, 40]. In terms of the learning method, the majority of the research in this field used supervised learning approaches, including discriminative classifiers (e.g. Decision Trees, SVM) and generative models (e.g. Naive Bayes, Hidden Markov Model), where a classifier is trained on a large set of labeled examples of every target activity. [2, 3, 18, 21, 25, 40]. There has also been prior study of representing high-level activities as a composite of simple actions, using a supervised layered dynamic Bayesian network [39]. While many promising results have been reported, a widely acknowledged problem is that labeled examples are often time consuming and expensive to obtain, as they require a lot of effort from test subjects, human annotators, or domain experts [36, 37].

6.1.2 *Semi-Supervised and Transfer Learning*

To lessen the reliance on labeled training data and to exploit the benefit of abundant unlabeled data, previous work has incorporated semi-supervised learning into activity or context recognition systems [19, 22, 24, 36, 37]. Semi-supervised learning approaches can improve the recognition accuracy by refining the decision boundary based on the distribution of the unlabeled data, or by assigning highly-confident estimated labels to the unlabeled data. Recently, transfer learning has also been explored so that the model learned for one target class can be transferred to improve the recognition accuracy of another target class [5, 41]. As a result, the amount of training data required for new applications can be reduced. While many promising results

have been reported, most of the existing approaches can only recognize activity classes that were included in the training data. Inspired by previous study, our work presents an early attempt to recognize unseen human activities with no training data using attribute-based zero-shot learning.

6.1.3 Active Learning

Active learning has been used to improve the accuracy of human activity recognition [17, 19, 37] or to model the interruptibility of a mobile phone user [33]. We extend the previous work by incorporating active learning in the framework of zero-shot learning for activity recognition, so that the system is able to not only recognize unseen activities but also actively request labels from users when possible.

6.1.4 Unsupervised Learning

Another related research direction is unsupervised learning. Unsupervised learning focuses on clustering or pattern discovery rather than classification [9, 26]. Although labels are not required for unsupervised learning approaches, the output is a set of unnamed clusters which cannot be used for classification purposes. To perform classification, labels are still needed to connect the discovered patterns to the actual classes.

6.1.5 Human Activity Domain

In terms of the activity domain of interest, some previous work in the area of human activity recognition focused on daily life activities [9, 18, 36] and some focused on sports and exercise activities [7, 28]. In this work, we evaluated our system in both domains to validate its effectiveness in general unseen activity recognition.

6.2 Zero-Shot Learning

The idea of semantic or human-nameable visual attributes and zero-shot learning has recently been explored in the field of computer vision such as object recognition and has been shown to be useful [13, 16, 34]. Palatucci et al. presented an early study on the problem of zero-shot learning [30], where the goal is to learn a classifier that can predict new classes that were omitted from the training dataset. A theoretical analysis is done to study the conditions under which a classifier can predict novel classes. For case study, the authors study the problem of decoding the word that a human is thinking from functional magnetic resonance images (fMRI). There has also been some previous work on using zero-shot learning to solve computer vision problems [13, 34]. In [13], the system does object detection based on a human-specified high-level description (such as shapes or colors) of the target objects instead of training images. Compared to our work, these problem domains are inherently different from activity recognition because image data are static rather than sequential. Furthermore, the image attributes and description for visual objects cannot be directly applied to activity recognition. Inspired by these previous studies, we designed and implemented a new activity recognition system using the concept of attribute-based zero-shot learning.

7. CONCLUSION

In this paper, we have presented the design, implementation, and evaluation of NuActiv, a new activity recognition system. Existing machine-learning-based approaches can only classify sensor data as one of the pre-trained classes in the training data, and thus cannot recognize any previous unseen class without training data. NuActiv uses semantic attribute-based learning to recognize unseen new activity classes by reusing and generalizing the attribute model learned for other seen activities. An outlier-aware active learning algorithm is also developed to efficiently improves the recognition accuracy of the system using minimal user feedback. The system achieved up to an average of 79% recognition accuracy on the unseen activity recognition problem, which could not be solved using existing supervised or semi-supervised approaches. Experimental results also support our hypothesis that using outlier-aware active learning the system can converge faster to optimal accuracy using fewer labels from the user.

8. ACKNOWLEDGMENTS

This research is supported in part by Applied Research Center, Motorola Mobility. The views and conclusions contained in this paper are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of any funding body. We are grateful for the advice and help from Dr. Ying Zhang, Dr. Ole Mengshoel, Senaka Buthpitiya, Faisal Luqman, and Zheng Sun. We sincerely appreciate the suggestions and feedback from our shepherd, Dr. Junehwa Song, and from anonymous reviewers.

9. REFERENCES

[1] M. Azizyan, I. Constandache, and R. Roy Choudhury. SurroundSense: Mobile phone localization via ambience fingerprinting. In *Proc. Int'l Conf. Mobile Computing and Networking*, pages 261–272, 2009.

[2] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In *Proceedings of The International Conference on Pervasive Computing*, pages 1–17. Springer, 2004.

[3] E. Berlin and K. Van Laerhoven. Detecting leisure activities with dense motif discovery. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 250–259, New York, NY, USA, 2012. ACM.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag Inc., 2006.

[5] U. Blanke and B. Schiele. Remember and transfer what you have learned - recognizing composite activities based on activity spotting. In *International Symposium on Wearable Computers (ISWC)*, pages 1 –8, Oct. 2010.

[6] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] K.-H. Chang, M. Y. Chen, and J. Canny. Tracking free-weight exercises. In *Proc. Int'l Conf. Ubiquitous computing*, UbiComp '07, pages 19–37, 2007.

[8] Google Inc. Google Glass, Apr. 2013. `http://www.google.com/glass/`.

[9] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. In *Proceedings of The International Conference on Ubiquitous Computing*, UbiComp '08, pages 10–19, New York, NY, USA, 2008. ACM.

[10] S. Kang, J. Lee, H. Jang, H. Lee, Y. Lee, S. Park, T. Park, and J. Song. Seemon: scalable and energy-efficient context monitoring framework for sensor-rich mobile environments. In *Proceedings of The 6th International Conference On Mobile Systems, Applications, And Services*, MobiSys '08, pages 267–280, New York, NY, USA, 2008. ACM.

[11] S. Kang, Y. Lee, C. Min, Y. Ju, T. Park, J. Lee, Y. Rhee, and J. Song. Orchestrator: An active resource orchestration framework for mobile context monitoring in sensor-rich mobile environments. In *International Conference on Pervasive Computing and Communications (PerCom)*, pages 135–144, 2010.

[12] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda. Learning systems of concepts with an infinite relational model. In *Proceedings of the 21st national conference on Artificial intelligence - Volume 1*, AAAI'06, pages 381–388. AAAI Press, 2006.

[13] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958, 2009.

[14] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell. A survey of mobile phone sensing. *IEEE Comm. Magazine*, 48:140–150, September 2010.

[15] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *Proc. Conf. on Artificial intelligence*, AAAI'08, pages 646–651, 2008.

[16] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *Conf. Computer Vision and Pattern Recognition*, pages 3337–3344, 2011.

[17] R. Liu, T. Chen, and L. Huang. Research on human activity recognition based on active learning. In *International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 1, pages 285–290, 2010.

[18] B. Logan, J. Healey, M. Philipose, E. M. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Proc. Int'l Conf. Ubiquitous computing*, UbiComp '07, pages 483–500, 2007.

[19] B. Longstaff, S. Reddy, and D. Estrin. Improving activity classification for health applications on mobile devices using active and semi-supervised learning. In *International Conference on Pervasive Computing Technologies for Healthcare*, PervasiveHealth'10, pages 1–7, 2010.

[20] H. Lu, W. Pan, N. Lane, T. Choudhury, and A. Campbell. SoundSense: Scalable sound sensing for people-centric applications on mobile phones. In *Proc. Int'l Conf. Mobile systems, applications, and services*, pages 165–178, 2009.

[21] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The Jigsaw continuous sensing engine for mobile phone applications. In *Proc. ACM Conf. Embedded Networked Sensor Systems*, SenSys '10, pages 71–84, 2010.

[22] M. Mahdaviani and T. Choudhury. Fast and scalable training of semi-supervised CRFs with application to activity recognition. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, pages 977–984. MIT Press, Cambridge, MA, 2007.

[23] L. M. Manevitz and M. Yousef. One-class SVMs for document classification. *Journal of Machine Learning Research*, 2:139–154, Mar. 2002.

[24] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: the evolution of sensing and inference on mobile phones. In *Proc. Int'l Conf. Mobile systems, applications, and services*, pages 5–20, 2010.

[25] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application. In *Proc. ACM Conf. Embedded network sensor systems*, SenSys '08, pages 337–350, 2008.

[26] D. Minnen, T. Starner, I. Essa, and C. Isbell. Discovering characteristic actions from on-body sensor data. In *In Proc. International Symposium On Wearable Computing*, pages 11–18, 2006.

[27] Motorola Mobility. MotoACTV, Apr. 2013. `https://motoactv.com/`.

[28] M. Muehlbauer, G. Bahle, and P. Lukowicz. What can an arm holster worn smart phone do for activity recognition? In *Proc. International Symposium on Wearable Computers*, ISWC '11, pages 79–82, 2011.

[29] S. Nath. ACE: exploiting correlation for energy-efficient and continuous context sensing. In *Proc. Int'l Conf. Mobile systems, applications, and services*, MobiSys '12, pages 29–42, 2012.

[30] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *Proceedings of the Neural Information Processing Systems (NIPS)*, pages 1410–1418, 2009.

[31] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *Proc. Int'l Conf. Computer Vision and Pattern Recognition*, pages 1681–1688, 2011.

[32] D. Parikh and K. Grauman. Relative attributes. In *IEEE International Conference on Computer Vision (ICCV)*, pages 503–510, 2011.

[33] S. Rosenthal, A. K. Dey, and M. Veloso. Using decision-theoretic experience sampling to build personalized mobile phone interruption models. In *Proc. International Conference on Pervasive Computing*, Pervasive'11, pages 170–187, Berlin, Heidelberg, 2011. Springer-Verlag.

[34] O. Russakovsky and L. Fei-Fei. Attribute learning in large-scale datasets. In *Proceedings of the 11th European Conference on Trends and Topics in Computer Vision*, ECCV'10, pages 1–14, Berlin, Heidelberg, 2012. Springer-Verlag.

[35] B. Settles. *Active Learning*. Morgan & Claypool, 2012.

[36] M. Stikic, D. Larlus, S. Ebert, and B. Schiele. Weakly supervised recognition of daily life activities with wearable sensors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(12):2521–2537, 2011.

[37] M. Stikic, K. Van Laerhoven, and B. Schiele. Exploring semi-supervised and active learning for activity recognition. In *Proc. International Symposium on Wearable Computers (ISWC)*, pages 81–88, 2008.

[38] U.S. Bureau of Labor Statistics. American time use survey activity lexicon. *American Time Use Survey*, 2011.

[39] S. Wang, W. Pentney, A.-M. Popescu, T. Choudhury, and M. Philipose. Common sense based joint training of human activity recognizers. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 2237–2242, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[40] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proc. Int'l Conf. Mobile systems, applications, and services*, MobiSys '09, pages 179–192, 2009.

[41] V. W. Zheng, D. H. Hu, and Q. Yang. Cross-domain activity recognition. In *Proc. International Conference on Ubiquitous Computing*, Ubicomp '09, pages 61–70, New York, NY, USA, 2009. ACM.