Towards Less Supervision in Activity Recognition from Wearable Sensors

Tâm Huỳnh and Bernt Schiele
Computer Science Department, TU Darmstadt, Germany
{huynh, schiele}@mis.tu-darmstadt.de

Abstract

Activity Recognition has gained a lot of interest in recent years due to its potential and usefulness for context-aware wearable computing. However, most approaches for activity recognition rely on supervised learning techniques limiting their applicability in real-world scenarios and their scalability to large amounts of activities and training data. State-of-the-art activity recognition algorithms can roughly be divided in two groups concerning the choice of the classifier, one group using generative models and the other discriminative approaches. This paper presents a method for activity recognition which combines a generative model with a discriminative classifier in an integrated approach. The generative part of the algorithm allows to extract and learn structure in activity data without any labeling or supervision. The discriminant part then uses a small but labeled subset of the training data to train a discriminant classifier. In experiments we show that this scheme enables to attain high recognition rates even though only a subset of the training data is used for training. Also the tradeoff between labeling effort and recognition performance is analyzed and discussed.

1 Introduction

In recent years activity recognition has gained interest and convincing results have been reported on various databases (e.g. [1, 13, 15, 6, 20, 8]). Interestingly, while there seem to be but a few types of features employed, there exists quite a range of classification methods, going from generative and probabilistic models [1, 13, 20] to discriminative approaches [11, 10].

Generative models are quite appealing for various reasons in the context of activity recognition. For example those models can be learned incrementally or even in a fully unsupervised fashion [3], they can deal with missing data in a principled way, they allow for modular construction of composed solutions to complex problems and therefore lend themselves to hierarchical classifier design. Also, prior knowledge can be easily taken into account. However, the price for these favorable properties is that generative models tend to produce a significant number of false positives. This is particularly true for activities that are rather similar such as walking and walking upstairs. Therefore it is difficult to scale these approaches to a wide range of sometimes highly similar activities.

Discriminative methods enable the construction of flexible decision boundaries, resulting in classification performances often superior to those obtained by purely probabilistic or generative models [4, 12]. This allows for example to explicitly learn the discriminant features of one particular activity or between multiple activities.

While so far the activity recognition community has typically chosen one of these two modeling approaches, there has been an increasing interest in the machine learning community in developing algorithms which combine the advantages of discriminative methods with those of probabilistic generative models [4, 14], showing improvements in performance with respect to purely discriminative or generative approaches in information extraction (e.g. from biomedical text and gene finding [14]).

In this paper we integrate two different types of approaches into a single common framework to fully exploit their strengths while minimizing their weaknesses. More specifically, we combine a generative model (multiple eigenspaces) with SVM training on partially labeled training data. The idea of using a generative model inside a kernel function has been proposed before [4, 5, 19, 16] and a similar idea has been applied to activity recognition [11]. However, these approaches do neither address nor analyze the issue of reducing the amount of supervision and labeled training data.

The first main contribution of this paper is the combination of the generative model of multiple eigenspaces with with a discriminant SVM classifier into a single activity recognition framework. On the one hand the new integrated approach allows to significantly increase recognition accuracy w.r.t. to the multiple eigenspace approach by rejecting false positives more effectively. On the other hand the approach also enables to train discriminant classifiers on only

1-4244-0598-X /06/$20.00 ©2006 IEEE.

part of the data and therefore allows to substantially reduce the amount of supervision required. The second main contribution are experimental results which show the superiority of the new integrated approach with respect to the multiple eigenspace approach and to a baseline system using a naïve Bayes classifier. The performance is analyzed in particular with respect to the amount of labeled training data used and therefore the tradeoff between supervision and recognition accuracy is shown experimentally.

The rest of the paper is organized as follows. Section 2 describes the multiple eigenspace approach. Section 3 then introduces the integrated approach using multiple eigenspaces and discriminant training in oder to learn discriminant classifiers on partially labeled data. The experiments (section 4) consist of three sets of experiments to analyze the performance of the integrated approach as well as to analyze its performance when the amount of supervision is reduced.

2 Multiple Eigenspaces

The multiple eigenspace algorithm is a general procedure to extract and represent low-dimensional structure from high-dimensional input data [9]. It is based on principal component analysis (PCA), a common technique in pattern recognition to reduce the dimensionality of feature spaces (see [2], e.g.). While PCA finds a single eigenspace that best represents all input features in a least-squares sense, the multiple eigenspace approach finds several of such eigenspaces, each representing a highly correlated subset of the input data. The advantage of this approach is that the dimensionality of the resulting eigenspaces can be much lower than when using a single eigenspace. More importantly, each of the eigenspaces can serve as a model for correlated subsets of the data. In previous work we have shown that such models can be used to detect and represent structure such as individual activities in accelerometer data [3].

In the following we will briefly describe the multiple eigenspace algorithm and give an example in which we apply the algorithm to features computed from acceleration sensors. (For a more detailed introduction the reader is referred to [9, 3].)

Overview. The problem which the algorithm solves is, given a set of feature vectors $\mathcal{G}$, to find sets $\mathcal{G}_j \subset \mathcal{G}$, eigenspaces $\mathcal{E}_j(\mathcal{G}_j)$ and dimensions p_j, so that each feature vector $\mathbf{x}_i \in \mathcal{G}_j$ can be approximated to a predefined degree of accuracy by its projection

$$\hat{\mathbf{x}}_i \quad = \quad \mathbf{e}_{0j} + \sum_{k=1}^{p_j} y_{kj}\mathbf{e}_{kj}. \qquad (1)$$

i.e., by a vector $\mathbf{e}_{0j} \in \mathbb{R}^n$ plus a linear combination of p_j (eigen-)vectors $\mathbf{e}_{1j}, \ldots, \mathbf{e}_{p_j j}$ ($p_j < n, \mathbf{e}_{kj} \in \mathbb{R}^n$).

The algorithm consists of three phases: initialization, eigenspace growing and eigenspace selection. During *initialization*, small subsets or *seeds* of data vectors are chosen from the input set $\mathcal{G}$, and their respective eigenspaces are calculated and initialized with dimension zero. During *eigenspace growing*, the initial sets are successively enlarged by adding data vectors and accepting or rejecting them based on reconstruction error. At the same time, the corresponding eigenspaces are recomputed and their dimension is adapted. As the growing process produces overlapping and thus redundant sets and eigenspaces, the final *eigenspace selection* phase applies an optimization procedure that finds a subset of eigenspaces that best represent the data with minimal redundancy. Importantly, the number of eigenspaces that are finally selected is determined automatically during eigenspace selection and does not have to be specified in advance.

Example. Figure 1 shows an example of applying the multiple eigenspace algorithm to features computed from twelve body-worn accelerometers (the features and dataset are described in detail in section 4.1). The upper plot of Figure 1 shows the features, and the middle plot shows the ground truth of the recording, consisting of eight different activities the user was performing while wearing the sensors.

When applying the multiple eigenspace algorithm to the features (each feature representing a seed), 16 eigenspaces are chosen during eigenspace selection. Each eigenspace is a representative model for a subset of the input features, and these models often correspond to activities of the user. This can be seen from the bottom plot of Figure 1, in which we assign to each original feature the eigenspace which has the lowest reconstruction error with respect to the feature. E.g., eigenspace 16 largely corresponds to activity 6, eigenspace 14 to activity 1, and eigenspace 2 to activity 2.

Even though there is no one-to-one correspondence between models and activities, one can clearly see that there is a high correlation between several models and activities. This can be used to turn the models into classifiers, by associating each model with the activity that occurs most often in the set of features that it covers.

3 Combining Multiple Eigenspaces with Support Vector Machines

In prior work we have shown that the approach of multiple eigenspaces can be used for unsupervised discovery of structure in activity data [3]. Quite interestingly, the system was able to build models that correspond to different activities without requiring any prior training, user annotation or

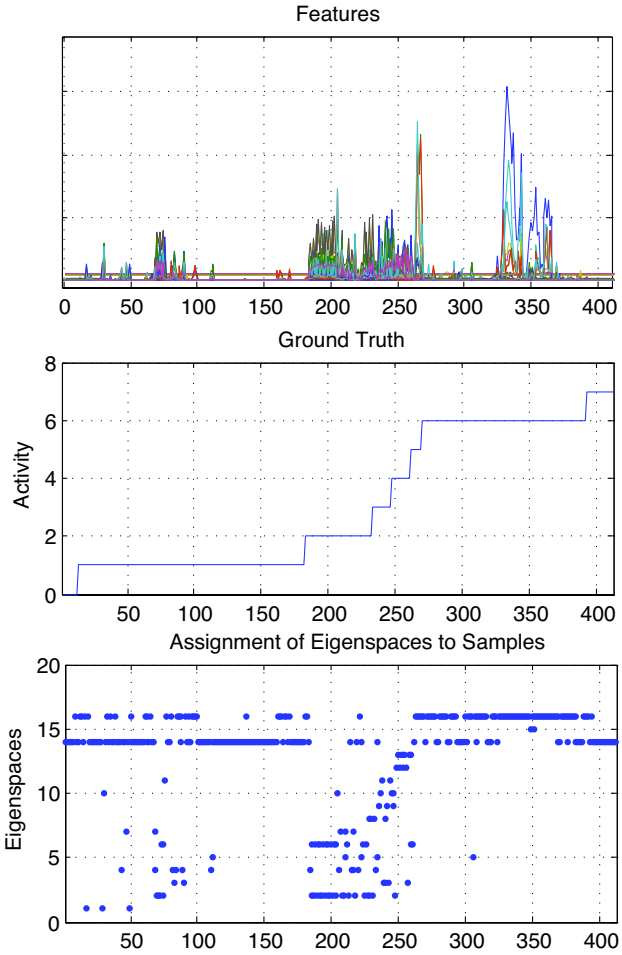

Figure 1. Application of the multiple eigenspace algorithm to features computed from twelve body-worn accelerometers. From top to bottom: mean and variance of the acceleration signals; ground truth (0 *Sitting*, **1** *Standing*, **2** *Walking*, **3** *Walking Upstairs*, **4** *Walking Downstairs*, **5** *Shaking Hands*, **6** *Writing on the Whiteboard*, **7** *Typing on a keyboard*)**; assignment of eigenspaces to samples based on reconstruction error**

information about the number of activities involved. When used for classification, the system showed recognition rates comparable to other, supervised techniques.

However, the approach of multiple eigenspaces suffered – as many other generative models – from a significant number of false positives. In order to improve recognition performance there is a clear desire to learn and incorporate discriminant information through a discriminant classification scheme. In this paper we adopt the support vector framework as it has shown competitive performance on a wide range of different classification tasks.

Obviously, the incorporation of a discriminant classifier requires to use labeled training samples. However, an important emphasis of this integrated approach is to keep the amount of required supervision to a minimum. Therefore, during training the approach leverages on the ability of multiple eigenspaces to learn in an unsupervised fashion an intermediate representation and structure description of the sensor data. This allows to use relatively small amounts of supervision while still obtaining competitive recognition performance.

3.1 Support Vector Machines

In the following we will briefly describe classification with Support Vector Machines (SVMs). Further details can be found in [18], e.g. Consider the problem of separating a set of training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_1, y_1)$ into two classes, where $\mathbf{x}_i \in \mathbb{R}^N$ is a feature vector and $y_i \in \{-1, +1\}$ its class label. If we assume that the classes can be separated by the hyperplane $\mathbf{w} * \mathbf{x}_i + b = 0$, and that we have no prior knowledge about the data distribution, then the optimal hyperplane (i.e., the one with the lowest bound on the expected generalization error) is the one with the maximum distance to the closest points in the training set. The optimal values for $\mathbf{w}$ and b can be found by solving the following constrained minimization problem:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \qquad (2)$$

$$\text{subject to } y_i(\mathbf{w} * \mathbf{x}_i + b) \geq 1, \forall i = 1, \ldots, m \qquad (3)$$

Solving it using Lagrange multipliers $\alpha_i (i = 1, \ldots, m)$ results in a classification function

$$f(x) = sign\left(\sum_{i=1}^{m} \alpha_i y_i \mathbf{w} * \mathbf{x} + b\right). \qquad (4)$$

where α_i and b are found using an SVM learning algorithm [18]. Most of the α_i take the value of zero. Those $\mathbf{x_i}$ with nonzero α_i are the so-called *support vectors*. In cases where the classes are non-separable, the solution is identical to the separable case with a modification of the Lagrange multipliers to $0 \leq \alpha_i \leq C, i = 1, \ldots, m$, where C is the penalty for misclassification.

To obtain a nonlinear classifier, one maps the data from the input space $\mathbb{R}^N$ to a high dimensional feature space $\mathcal{H}$ by $\mathbf{x} \rightarrow \Phi(\mathbf{x})$, such that the mapped data points of the two classes are linearly separable in the features space. Assuming there exists a kernel function K such that $K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) * \Phi(\mathbf{y})$, a nonlinear SVM can be constructed by replacing the inner product $\mathbf{w} * \mathbf{x}$ by the kernel function $K(x, y)$ in eqn. 4. This corresponds to constructing an optimal separating hyperplane in the feature space. Kernels commonly used include polynomials $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} * \mathbf{y})^d$, and the Gaussian Radial Basis Function (RBF) kernel $K(\mathbf{x}, \mathbf{y}) = exp\{-\gamma \|\mathbf{x} - \mathbf{y}\|^2\}$. For the features we use in our experiments (sec. 4), the RBF kernel showed the best performance.

The extension of SVMs from 2-class to N-class problems can be achieved e.g. by training N SVMs, each separating a single class from all remaining classes.

3.2 Combining Multiple Eigenspaces with SVMs

In order to train an SVM with minimal labeled training data we proceed in two steps. In the first step we use the multiple eigenspace approach to obtain a description of the data that captures the essential structure of the activity data. As the multiple eigenspace approach is fully unsupervised we can use all sensor data that is available to us regardless if it is labeled or not. The use of unsupervised training data is essential and a desired feature of the approach as this allows to use far more data than in the supervised case to derive a good representation and description of the overall sensor data. Essentially the approach of multiple eigenspaces provides a low-dimensional description of the sensor-data which lends itself for training SVMs.

In the second step we use the eigenspaces produced by the multiple eigenspace algorithm to construct features for training the discriminant SVMs. More specifically we calculate for each labeled training sample (that is for a subset of the entire training set) a vector $\mathbf{d} = d_1, ...d_n$ of reconstruction errors. The element d_i is the error of the sample with respect to the eigenspace i, introduced earlier. This corresponds to a soft assignment of samples to eigenspaces, as opposed to the hard assignment in the previous section, where each sample was assigned to exactly one eigenspace. By doing this we provide the SVMs with more information for their classification task.

Before feeding the error vectors and the corresponding sample labels to the SVM, we scale each component d_i of the vectors by

$$\hat{d}_i = e^{-\frac{{d_i}^2}{\sigma^2}} \tag{5}$$

The goal of this transformation is to get feature values between 0 and 1 with the property such that the distribution of small (and discriminant) errors is preserved and the errors beyond a certain threshold are mapped to zero. The threshold can be controlled by adjusting the value of σ in equation 5, which is related to the variance of the transformation function.

We use a radial basis function as kernel for the SVMs. The parameters C and γ (γ corresponds to the width of the RBF-kernel) are determined by sampling the parameter space on an exponential grid.

During classification, we assign to each test sample a vector $\mathbf{d}$ of reconstruction errors and transform this vector in the same fashion as done for training. We then pass the transformed vector to the SVMs for testing.

4 Experiments

The following experiments are designed to investigate two main questions. First, how does the performance of the new integrated approach of multiple eigenspaces and SVM-training compare to its predecessor, namely multiple eigenspaces, and to naïve bayes classification as a baseline. And second, what is the effect of decreasing the amount of labeled training data on the three different types of learning algorithms.

After introducing the employed data set in section 4.1, the following describes three sets of experiments, each set for a different classification scheme. The first experiment (section 4.2) describes and analyses the naïve Bayes approach as a baseline and example of a classical supervised approach. The second set of experiments (section 4.3) uses the unsupervised and generative approach of multiple eigenspaces described above. The third set of experiments (section 4.4) describes the results of the integrated approach that combines the generative nature of multiple eigenspaces with SVM learning on subsets of training data.

Each experiment is performed in five different configurations. In each configuration we change the amount of labeled training data while leaving the size of the test data unchanged. We start by dividing the entire dataset into 80% training and 20% test set. Then we gradually decrease the amount of labeled training data, from 80% down to 5% of the entire dataset. For the naïve Bayes classifier, this means that the size of the available training data is reduced in each iteration, as it cannot learn from unlabeled data. The multiple eigenspace approach however can still be trained on the unlabeled data. It uses the reduced set of labeled training data only for learning a mapping from labels (i.e. activities) to the models it has constructed from the unlabeled data. Similarly the integrated approach uses the unlabeled data to train the multiple eigenspace model and only uses the reduced set of labeled training data for SVM-training.

The recognition rates that we report represent the recall of the classifier with respect to an activity, i.e., the number

of correctly classified samples divided by the total number of samples for a given activity.

4.1 Data Set

For our experiments we use a dataset published by Kern et al. [6]. It consists of eight everyday activities, namely *sitting, standing, walking, walking upstairs, walking downstairs, shaking hands, writing on the whiteboard* and *typing on a keyboard*. The activities were recorded by twelve 3D acceleration sensor nodes distributed over the user's body. The sensors were attached to the ankles, knees, elbows, shoulders, wrists and to both sides of the user's hip. Each node consists of two 2D-accelerometers fixed at an angle of 90 degrees. The overall length of the dataset is 18.7 min, recorded at 92Hz.

The data was recorded in one consecutive run, and therefore each activity is represented by a different amount of data. To avoid bias in our recognition experiments, we use an equal (random) amount of data from each activity for constructing our test- and training sets.

The data samples are vectors of 48 acceleration values, from which we compute the running mean and variance over a window of 50 samples (i.e. about 0.5 seconds), which gives us a feature vector with 96 entries. The window is shifted over the data one sample at a time. In general one could use smaller overlaps between windows, but we found that smaller overlaps have a negative impact especially for very small amounts of training data. Since our experiments focus on reducing the amount of training data, we decided to use the maximum overlap for best performance.

Mean and variance of the acceleration signal are cheap to compute and have successfully been used for recognizing the activities we are considering (e.g. [1, 6, 7, 15]). We use these simple features for all approaches described in this work.

4.2 Naïve Bayes

We use a naïve Bayes classifier as a baseline for our experiments. It is a supervised approach which requires labeled training data for classification. Despite its simplicity, naïve Bayes has yielded high recognition rates for the activities and features we use [6, 17].

Bayes' rule states that the probability $p(a|\mathbf{x})$ of an activity a given an n-dimensional feature vector $\mathbf{x} = x_1, ...x_n$ can be calculated as

$$p(a|\mathbf{x}) = \frac{p(\mathbf{x}|a)p(a)}{p(\mathbf{x})}.$$

In this equation, $p(a)$ denotes the a-priori probability of the activity. The a-priori probability $p(\mathbf{x})$ of the data is only used for normalization. We ignore it in our experiments, since we are only interested in relative likelihoods and not absolute probabilities.

Assuming that the different components x_i of the feature vector $\mathbf{x}$ are independent, we can compute the likelihood $p(\mathbf{x}|a) = \prod_{i=1}^{n} p(x_i|a)$ from labeled training data.

Experiments. We represent each probability density function $p(x_i|a)$ by a 100 bin histogram. As stated, we use 96-dimensional vectors of running mean and variance over a window of 50 samples as features. We performed five experiments, each time reducing the amount of training data by a factor of two. We started with 80% training data and 20% test data, and gradually decreased the amount of training data to 5%. The amount of test data stayed constant for all experiments. Each experiment was repeated five times with different parts of the data for training and testing, and the average over all runs was taken as result. For the initial experiment (80% training, 20% test data), this corresponds to standard 5-fold crossvalidation.

Results. Table 1 shows the results of the experiment. Recognition of these activities obviously depends strongly on the amount of available training data. When looking at the average score over all activities, one observes that the score steadily drops by about 6% each time the amount of training data is halved, from 73.5% in the beginning to 50.1% percent in the end. This reduction in overall performance is to be expected and a main motivation to search for methods that can obtain high recognition scores using small amounts of training data.

Also, one can observe that the recognition rates vary greatly between the different activities. Stationary activities such as *sitting* and *standing* achieve higher rates than dynamic activities such as *walking upstairs* or downstairs. The highest rate is achieved for *standing* (95%), and the lowest for *walking upstairs* (51.4%). When comparing the different configurations, the rates for *standing* and *sitting* stay relatively stable as the amount of training data goes down. Obviously this is because there is not much variation in the features for these activities, which means that a small amount of samples is already enough to capture the characteristics of the activity. *Writing on a whiteboard* (which is similar to *standing*) has also high scores, but eventually drops from 93.6% in the beginning to 81.3% when only 5% of the data are used for training. The scores for *walking* drop significantly – from 86.5% to 40.8% – when the training data is reduced, as do the scores for *shaking hands* (from 53.7% to 9.8%). *Walking upstairs* and *walking downstairs* also both drop by about 30%.

Activity	Amount of Training Data 80%	40%	20%	10%	5%
stand	95.0	94.8	94.5	94.1	93.5
sit	91.7	92.3	92.8	91.6	92.5
walk	86.5	79.9	70.6	55.6	40.8
upstairs	59.3	50.4	39.1	36.3	31.2
downstairs	51.4	35.4	26.0	23.2	19.9
shake hands	53.7	43.9	32.5	20.4	9.8
whiteboard	93.6	93.2	91.4	87.1	81.3
keyboard	57.0	50.8	56.0	45.6	31.8
Average	73.5	67.6	62.9	56.7	50.1

Table 1. Recognition Scores using naïve Bayes

Activity	Amount of Training Data 80%	40%	20%	10%	5%
stand	82.4	85.8	84.1	84.3	82.2
sit	43.1	43.4	52.1	54.8	47.1
walk	74.9	74.6	75.4	74.2	73.1
upstairs	68.7	67.8	68.6	64.0	64.0
downstairs	59.2	63.2	62.8	59.8	53.3
shake hands	52.6	53.0	59.5	53.9	53.6
whiteboard	81.8	78.6	77.2	78.9	78.8
keyboard	92.1	90.2	90.7	91.0	91.8
Average	69.4	69.6	71.3	70.1	68.0

Table 2. Recognition Scores using Multiple Eigenspaces

4.3 Multiple Eigenspaces

For this experiment we first trained the multiple eigenspace algorithm on the unlabeled features, i.e. on the mean and variance over running 50-sample windows of the acceleration data. In order to reduce the time and space complexity of the growing and selection phases of the algorithm, we performed a k-means clustering ($k = 100$) on the features and used the resulting cluster centers as seeds for the growing phase. Using other cluster numbers than 100 only had small effects on recognition performance.

Next we assigned activities to the models (i.e. eigenspaces) produced by the algorithm. First we assigned to each training sample the model with the lowest reconstruction error. Then, using the labels of the training samples, we counted which activity was associated most often with a given model. This activity was then assigned to the model, so that the model could be used later for classifcation. Again, we conducted five experiments in total, each time reducing the number of labeled samples used for finding the mapping between models and activities. We decreased the amount of labeled samples in the same way as for the previous experiment (from 80% to 5% of the entire dataset) and left the size of the test set unchanged.

For testing, we assigned to each test sample the activity of the model with the lowest reconstruction error. Each configuration was run five times with different parts of the data for training and testing, and the average over all five runs was taken as result.

Results. Table 2 shows the results of the experiment. The distribution of scores differs in various aspects from the naïve Bayes experiment. When using 80% training data, the mean recognition score is slightly lower than that of the naïve Bayes experiment. However, the strength of the approach becomes visible when looking at the runs with reduced training labels – other than in the supervised approach, the average score does not drop but consistently stays at around 70%. This tendency is visible for all individual activities – for none of them, the recognition rate drops by more than 6% when reducing the labeled training set from 80% to 5% of the dataset.

Throughout all configurations, *standing*, *writing on a whiteboard* and *typing on a keyboard* achieve the highest recognition scores, with *typing on a keyboard* consistently scoring over 90%. The lowest rates are achieved for *sitting* and *shaking hands*.

4.4 Multiple Eigenspaces combined with SVMs

In this experiment we first trained the multiple eigenspace algorithm with the same mean and variance features as in the previous experiment. Again, seeding of the eigenspace growing was performed using kmeans clustering ($k = 100$) on the training features. As in the previous experiments, each configuration of test and training sets was run five times, and the average over all five runs was taken as result.

Results. Table 3 shows the results of this third experiment. When using all annotations and as expected from discriminant training of the SVMs, the recognition scores are significantly increased w.r.t. to both other approaches. The average performance is more than 88% with most scores above 85% and a maximum of 98% for *typing on a keyboard*. For example *Walking upstairs* and downstairs both reach over 90%, which is considerably higher than in both of the previous experiments. *Shaking hands* also has rates which are 30% to 40% higher than in the previous experiments. As the number of annotations is reduced, the average recognition score drops from 88.8 to 64.7%, which

Activity	Amount of Training Data 80%	40%	20%	10%	5%
stand	88.0	82.3	80.6	75.5	66.2
sit	83.2	78.6	73.8	69.2	48.6
walk	79.7	68.8	71.1	60.9	51.1
upstairs	91.7	89.7	82.1	78.8	74.9
downstairs	95.0	92.1	88.3	75.2	78.8
shake hands	85.6	85.2	75.9	72.1	67.3
whiteboard	85.8	76.7	75.6	59.9	50.5
keyboard	98.0	94.1	94.9	87.3	80.2
Average	88.3	83.4	80.3	72.4	64.7

Table 3. Recognition Scores using Multiple Eigenspaces combined with an SVM

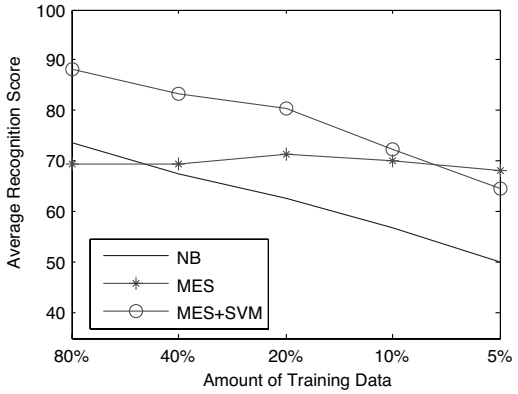

Figure 2. Comparison of recognition performance of the three approaches

is a similar dropoff compared to the naïve Bayes approach. However, the absolute scores are about 15-20% above those of the naïve Bayes experiment.

4.5 Discussion

Figure 2 shows the performance of all three approaches in one plot. The plot shows the average performance as the size of the labeled training data is successively reduced by a factor of two. On this exponential scale, the performance of the supervised naïve Bayes approach decreases almost linearly. In contrast, the unsupervised multiple eigenspace approach has an almost constant performance throughout all configurations. While starting slightly lower (69% compared to 73%) than naïve Bayes, it outperforms the supervised approach already when the labeled training data is cut in half. The recognition performance of multiple eigenspaces stays stable until the last configuration, in which the labels are reduced to one sixteenth of the original amount. This clearly shows the advantages of this unsupervised and generative approach. The information it extracts from the unlabeled training data helps it to maintain its classification performance as the amount of labeled data goes down. More importantly, this implies that the multiple eigenspace approach can help to reduce the amount of supervision – and thus the amount of manual work by users – which is required for activity recognition. In contrast, since the supervised approach can only learn from labeled data, it is strongly dependent on annotation, as can be seen from the performance drop in Figure 2.

The last curve of the plots in figure 2 shows the performance of the integrated approach. Clearly, the overall performance when using 80% training data is far above the performance of the two other approaches. Even though the performance drops as the amount of training data is reduced, the performance using 20% of training data is still clearly above the performances of both approaches. Only when the amount of training data is further reduced to 10% the performance gain using SVM training becomes negligible. When using only 5% of the training data the performance stays behind the multiple eigenspace approach as the SVM training aims to generalize from the rather limited set of training samples.

Overall we can make three observations. First, the use of multiple eigenspaces can reduce the effort of supervision to very small amounts of training data while still preserving a good and constant level of recognition performance. Second, the integration of SVM-training on subsets of the training data can increase the overall recognition performance substantially assuming a sufficient amount of labeling data. Third, when labeling and amount of supervision are further reduced, the discriminant nature of SVM learning may not help anymore or can even hurt recognition performance.

5 Conclusion

This paper has introduced an integrated approach combining the advantages of generative modeling and discriminant learning. More specifically the generative approach of multiple eigenspaces was used to obtain a low-dimensional representation of sensor data in a fully unsupervised fashion. In particular the approach allows to model effectively different activities without prior training, user annotation, or any information about the number of activities involved. Support vectors machines are then trained on labeled subsets of the training data to boost the recognition accuracy of the purely unsupervised approach of multiple eigenspaces.

Experimental results showed several things. First of all the experiments showed that the multiple eigenspace approach can achieve a comparable performance to a baseline system using naïve Bayes classification. However, the performance of the multiple eigenspace approach remains high even when the amount of supervision is reduced

substantially from 80% to only 5%. Second, the experiments showed that the combined approach does indeed increase recognition performance substantially w.r.t both the purely unsupervised approach of multiple eigenspaces and the baseline recognition system based on naive Bayes.

Interestingly, the experiments and the discussion of the previous section also suggest that neither the multiple eigenspace approach nor the discriminant learning approach are sufficient. While the generative eigenspace approach obtains a constant performance even in the presence of decreased supervision, the discriminant learning using SVMs clearly obtains the best recognition performance when enough training data is available. However, when the amount of training data is reduced the performance gain is negligible and might be even reversed. These observations support in our opinion the claim that in order to obtain scalable activity recognition for real world scenarios we should aim to optimally combine generative with discriminant models. In the future we therefore want to investigate and extend the possibilities of discriminant learning from very small labeled training data leveraging on the ability to model and represent unlabeled data through methods such as multiple eigenspaces.

References

[1] L. Bao and S. Intille. Activity recognition from user-annotated acceleration data. In *Proc. Pervasive*, volume 3001 of *Lecture Notes in Computer Science*, pages 1–17, Vienna, Austria, April 2004. Springer.

[2] R. Duda, P. Hart, and D. Stork. *Pattern Recognition*. Wiley and Sons, 2004.

[3] T. Huynh and B. Schiele. Unsupervised discovery of structure in activity data using multiple eigenspaces. In *2nd International Workshop on Location- and Context-Awareness (LoCA)*, Dublin, Ireland, May 2006. Springer. To appear.

[4] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems*, 11:487–493, 1998.

[5] T. Jebara, R. Kondor, and A. Howard. Probability Product Kernels. *The Journal of Machine Learning Research*, 5:819–844, 2004.

[6] N. Kern, B. Schiele, and A. Schmidt. Multi-sensor activity context detection for wearable computing. In *European Symposium on Ambient Intelligence*, Eindhoven, The Netherlands, November 2003.

[7] A. Krause, D. Siewiorek, A. Smailagic, and J. Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, pages 88–97, 2003.

[8] K. V. Laerhoven and H.-W. Gellersen. Spine versus porcupine: A study in distributed wearable activity recognition. In *Proceedings of the 8th IEEE International Symposium on Wearable Computers*, pages 142–149, 2004.

[9] A. Leonardis, H. Bischof, and J. Maver. Multiple eigenspaces. *Pattern Recognition*, 35(11):2613–2627, 2002.

[10] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. In *Pervasive*, 2006. To appear.

[11] J. Lester, T. Choudhury, N. Kern, G. Borriello, and B. Hannford. A hybrid discriminative/generative approach for modeling human activities. In *International Joint Conference on Artificial Intelligence*, pages 766–772, Edinburgh, United Kingdom, August 2005.

[12] A. Ng and M. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems*. MIT Press, 2002.

[13] D. Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. *Proceedings of the 9th IEEE International Symposium on Wearable Computers*, pages 44–51, 2005.

[14] F. Perreira, A. Bernal, K. Crammer, and R. McDonald. Linear models for structure prediction. In *NIPS Workshop on Graphical Models and Kernels*, 2004.

[15] N. Ravi, N. Dandekar, P. Mysore, and M. Littman. Activity recognition from accelerometer data. *Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence (IAAI)*, 2005.

[16] K. Tsuda, M. Kawanabe, G. Rätsch, S. Sonnenburg, and K.-R. Müller. A new discriminative kernel from probabilistic models. *Neural Computation*, 14(10):2397–2414, 2002.

[17] K. Van Laerhoven, N. Kern, H. Gellersen, and B. Schiele. Towards a wearable inertial sensor network. *In Proc. of the IEE Eurowearable*, pages 125–130, 2003.

[18] V. Vapnik. *Statistical learning theory*. Wiley and Sons, NY, 1998.

[19] N. Vasconcelos, P. Ho, and P. J. Moreno. The kullback-leibler kernel as a framework for discriminant and localized representations for visual recognition. In *European Conference on Computer Vision*, pages 430–441. Springer, 2004.

[20] J. Ward, P. Lukowicz, and G. Tröster. Gesture spotting using wrist worn microphone and 3-axis accelerometer. *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence*, pages 99–104, 2005.