Discriminative Temporal Smoothing for Activity Recognition from Wearable Sensors

Jaakko Suutala, Susanna Pirttikangas, and Juha Röning

Intelligent Systems Group, Infotech Oulu
Computer Engineering Laboratory
90014 University of Oulu, Finland
{jaska,msp,jjr}@ee.oulu.fi

Abstract. This paper describes daily life activity recognition using wearable acceleration sensors attached to four different parts of the human body. The experimental data set consisted of signals recorded from 13 different subjects performing 17 daily activities. Furthermore, to attain more general activities, some of the most specific classes were combined for a total of 9 different activities. Simple time domain features were calculated from each sensor device. For the recognition task, we propose a novel sequential learning method that combines discriminative learning of individual input-output mappings using support vector machines (SVM) with generative learning to smooth temporal time-dependent activity sequences with a trained hidden Markov model (HMM) type transition probability matrix. The experiments show that the accuracy of the proposed method is superior to various conventional discriminative and generative methods alone, and it achieved a total recognition rate of 94% and 96% studying 17 and 9 different daily activities, respectively.

1 Introduction

Activity recognition from wearable sensors has become an important research topic in recent years [1], [2], [3]. Successful recognition of basic human activities based on sensing of body posture and motion can be used in different applications, such as health care, child care and elderly care, as well as in personal witness monitoring. In addition, it provides a mechanism for using the activities to control devices around us, for example to provide personalized services to assist those with physical disabilities or cognitive disorders.

In this paper we present a novel method for activity recognition from wearable sensors. It combines ideas from two major categories of supervised machine learning: discriminative and generative learning. Discriminative learning (e.g., kernel methods [4], [5], [6]) provides an effective framework for learning direct input-output mapping from a labeled training data set ($\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$, where $\mathbf{x}_i$ presents i:th input feature vector and y_i is i:th target class label) for particular applications, such as classification and regression to predict unknown examples. However, adapting a discriminative framework to more advanced learning problems, such as cases where input or/and output spaces can have a structure (a sequence, for example), is not straightforward.

H. Ichikawa et al. (Eds.): UCS 2007, LNCS 4836, pp. 182–195, 2007.

On the other hand, in generative learning methods, modeling of whole phenomena that generate the data is not efficient, and the discriminative properties are not modeled very powerfully, for example, in classification tasks [7], [8]. However, generative learning is easily extended to a structured domain and it is suitable for activity recognition, as the sequential nature of adjacent class labels can be modeled. Here, the idea that daily activities usually vary smoothly is applied and it is more likely that the same activity as the previous one is detected in a short time period. Moreover, it is useful to handle transitions between activities differently, because some transitions more probably will occur than others.

The most popular model of a generative learning category for sequential data is HMMs [9]. Conventional HMMs are typically trained in a non-discriminative manner, as they are not able to discriminate between different classes very well. Another problem is that multi-dimensional input vectors cannot be used directly and overlapping features are not allowed. The features have to be transformed into a sequence of discrete symbols using some quantization, clustering or static pre-classifier method, or by forming a continuous density model where each observation vector is modeled using some probability distribution, for example a Gaussian mixture [9].

To overcome the problems of discriminative and generative learning, we combine them in a novel way. First, we train a discriminative model (e.g., SVM) to predict confidence of activity labels from individual multi-dimensional input vectors in time-series sequences. Second, we use the conditional posterior probability outputs of a discriminative learning algorithm as the input observation to a generative model. The generative model has a HMM-type structure where observations are the predicted confidence measurements of different classes from the discriminative model. Then, a global transition matrix is trained by the well-known forward-backward (FB) algorithm [9]. Here, the temporal properties of different activities are modeled and individual predictions in sequence are smoothed to remove outliers. For example, in the activity sequence running, running, bicycling, running, the bicycling activity can be detected as an outlier. In the classification stage, the most probable state (i.e., label) sequence is recognized using Viterbi decoding [10]. This paper is an extension of the work [11] where sensor settings, data collection and feature selection along with classification of independent activity examples were studied. We use the same features calculated in the previous study. In addition, we compare our method with the earlier experiments along with other sequential learning methods, such as conventional HMMs and a SVM-HMM combination.

The rest of the paper is organized as follows. Section 2 presents related work in activity recognition by wearable sensors and sequential learning methods and scenarios. Section 3 describes the details of the methods and data set used in this paper and section presents the experimental results. Finally, the conclusions of the work are given in Section 5.

2 Related Work

In ubiquitous computing, activity recognition has been realised using vision, audio, and different environmental and wearable sensing devices [12], [13]. To be able to recognize the actions of an individual person related to everyday tasks, the study of wearable accelerometer sensors has become dominant in the field. Wearable computing provides personalized services [14], which can be utilized by mobile devices or in clothing to assist in health care, fitness or work-related tasks, for example. The use of wearable acceleration sensors provides calm technology that is possibly not as obtrusive for the users compared with vision- and audio-based sensing.

The study of activity recognition using wearable sensors has concentrated on problems from hardware setups and sensor placement to feature extraction and classification methods. Activity recognition using wearable acceleration sensors attached to five different body parts was studied by [1]. Along with comprehensive related work in the field, they present useful features for recognizing everyday activities and the important aspect of the need of user-specific training data for some activities. An 84% accuracy rate for 20 different activities was achieved using user-annotated training data and a decision tree classifier. [15] used cluster analysis to examine which are the best features and time window lengths for discriminating between different activities. According to them, different features, such as Fourier coefficients, mean, and variance as well as different window lengths, are needed in the recognition.

Different features and sensor positions were examined by [16] using a single device with a dual axis accelerometer and a light sensor. They recognized six primary activities: sitting, standing, walking, ascending stairs, descending stairs, and running. To be able to compute features in real time on a wrist watch-like platform, they use only time domain features and feature selection. Wrist position was the best when the subset of features was optimized for it. In multiple sensor recognition, [17] studied the number and placement of devices. Naturally, in recognizing different activities the position of the sensor for a particular activity is important (e.g., lower and upper body motion when walking, upper body when typing with a keyboard).

The sequential nature of activity data has been considered, also. The most popular method is generative HMMs or related methods. Static and dynamic hand gestures of a mobile user were studied by [18] using acceleration sensors with self-organized maps (SOM) and HMMs. [19] combined vision and accelerometers and recognized the gestures of sign language using HMMs. Different daily activities, such as sitting, standing, walking, running, climbing stairs, and bicycling, were recognized by [20]. They combined unsupervised clustering (SOM) with supervised learning (k-nearest neighbors) and sequential modeling (Markov chain). [3] presented methods for recognizing assembly and maintenance work activities by hand motion and activities using an accelerometer and a microphone. Their case study of a wood workshop assembly task uses analysis of sound intensity detection to segment signals, and the classification is performed by the fusion of linear discriminant analysis (sound) and HMMs (acceleration sensors).

In activity recognition, a study most similar to our work is presented in [2]. It uses discriminative learning of multi-dimensional input-output mapping and feature selection of individual examples using boosting, which is then combined with HMMs to capture temporal properties. Compared with our approach, which uses a global transition probability matrix between activities, they trained a single HMM for each activity where a transition matrix models inner-class hidden state variation. They used a single sensor board equipped with an accelerometer, a microphone, two light sensors, barometric pressure, humidity, and temperature sensors, and a compass, and they initially extracted over 600 features. [21] applied another discriminative sequential learning approach to physiological activity data using conditional random fields. In classifying a physical activity (watching TV or sleeping) based on nine different sensor measurements, the method showed more accurate results compared with non-sequential methods, which only use information from individual input vectors. In a different application area [22], support vector machines and temporal smoothing were combined to classify audio sequences, which uses methodology quite similar to ours. However, they only applied it to a binary classification domain to detect speech and non-speech components from a video soundtrack, and they used a more ad-hoc technique to transform SVM outputs into confidence values for temporal modeling compared with our approach.

More generally, the idea of combining discriminative and generative learning has been studied much recently, mostly in the fields of natural language processing and computational biology. [23] give an overview of learning sequential data from simple sliding window techniques to generative methods such as HMMs, as well as discriminative sequential methods, e.g., maximum entropy Markov models and conditional random fields (CRF), which overcome some of the HMM's problems of feature presentation and non-discriminative learning with more expensive training. Additionally, kernel methods have been extended to sequential data through kernel design [24], and structured learning of support vector machines [25] and Gaussian process classification [26], which utilize the idea of HMMs and CRFs in dynamic programming style optimization and inference. Jebara [7] presents a framework for including generative models (e.g., HMMs) in large margin discriminative learning using maximum entropy discrimination.

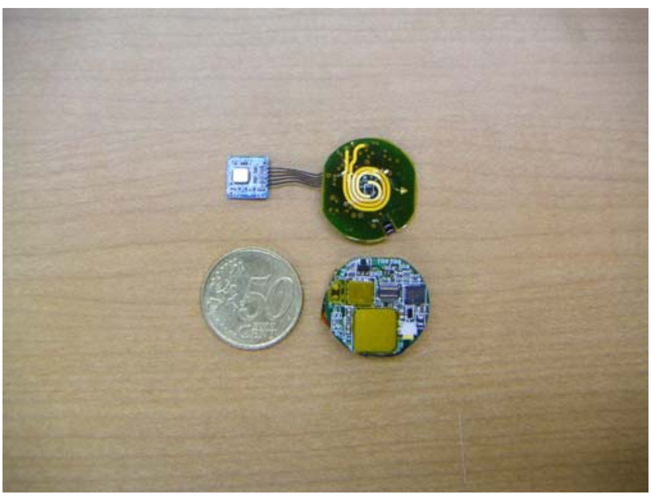

Fig. 1. Wearable sensor devices used in these experiments

Fig. 2. Attachment of sensor devices to the wrist

3 System for Sequential Learning of Activities

3.1 Activity Data Set and Feature Extraction

In this paper we used the data set collected in [11]. It includes activities recorded from 13 different subjects wearing four sensor nodes, which were attached to different parts of the body: the right thigh and wrist, the left wrist and a necklace. Each sensor node has a triaxial accelerometer that is sampled 64 times at 200 kHz, and the average values are sent every 100 msec to a data collecting terminal. The wearable sensor is presented in Figure 1, and the attachment of the sensor to the wrist is illustrated in Figure 2. The sensor was developed by the Nokia Research Center, Tokyo, in collaboration with the Distributed Computing Laboratory of Waseda University.

As presented in [11], each subject performed a sequence of 17 daily activities and annotated the starting and ending time of each activity using a touch screen or a wearable interface, depending on whether the particular activity was performed inside or outside. Each activity took at least one minute and altogether over 8 hours of data were collected. The 17 activities include *cleaning a whiteboard*, *reading a newspaper*, *standing still*, *sitting and relaxing*, *drinking*, *brushing teeth*, *sitting and watching TV*, *lying down*, *typing*, *vacuum cleaning*, *walking*, *climbing stairs*, *descending stairs*, *riding an elevator up*, *riding an elevator down*, *running*, and *bicycling*. Furthermore, some of the activities were combined into a single class, producing a data set of 9 general activities: *cleaning*, *standing*, *sitting*, *using stairs*, *brushing teeth*, *lying down*, *walking*, *running*, and *bicycling*. The *drinking* activity was left out because of its multimodal nature (i.e., the subjects were sitting or standing, etc.). Example activities in the data set are shown in Figure 3.

[11] tested different features and time windows and they found out that using a short time window (e.g., 0.7 - 1 second) with simple features (the mean and the standard deviation) gave the most accurate recognition rates. In this study, we also use a 0.7 second window and the mean and the standard deviation calculated from all 3 acceleration

channels of each sensor device, providing a total of 24 features in every time step. The use of such simple features is justified in an application where only limited computational resources are available and a relatively short time window is applied to achieve a real-time response.

(a) Sitting and reading newspaper (b) Walking

Fig. 3. Example activities performed by the subjects

3.2 Discriminative Learning of Static Examples: SVM Approach

Discriminative learning is a very effective way to train mappings from multidimensional input feature vectors to class labels. Kernel methods in particular have become state-of-the-art, due to their superior performance in many real-world learning problems, clear mathematical foundations and generalization capabilities based on statistical learning theory.

In this study, we use the popular support vector machines (SVM) [27] as base classifiers in our recognition system. The SVM method has many favorable properties such as good generalization by finding the largest margin between classes, the ability to handle non-separable classes via soft-margin criteria, non-linearity modeling via explicit kernel mapping, sparseness by presenting data using only a small number of support vectors, and global convex optimization with given hyperparameters. Fast optimization and a sparse solution are very important in building real-time systems with large-scale data sets. Training can be done effectively, for example by using sequential minimization optimization [28]. After training, an unknown example $f(\mathbf{x})$ in a binary classification case can be labeled as follows,

$$f(\mathbf{x}) = \sum_{i \in SV} y_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b \tag{1}$$

where α_i is a non-zero Lagrange multiplier, y_i is the class label of the training set, $k(\mathbf{x}, \mathbf{x}_i)$ is kernel mapping between an unknown example $\mathbf{x}$ and a training example $\mathbf{x}_i$, and b is the bias of the learned solution. SV represents the group of support vectors.

One drawback of SVM is that it is directly applicable only in two-class problems. Thus, there have been different attempts to generalize it to multi-class classification. The simplest and most popular methods are based on multiple binary classifiers using one-vs-one or one-vs-rest approaches as well as error correcting output codes and directed acyclic graphs, to name a few [5]. We apply the one-vs-one strategy due to its simplicity, its good performance in practice, and its capability of extended hard decisions to output confidence values using different post-processing methods. Next, we present algorithms that are able to produce posterior probabilities from binary SVM outputs and combine them into a multiple class classification.

SVM cannot directly give a confidence measurement as an output, but gives only a decision as an unscaled distance from the margin in feature space. However, [29] proposed a very useful method for getting probabilistic outputs by performing another mapping function from the raw outputs to class probabilities. This is calculated through a parametric sigmoid function, as follows

$$P(y = 1|f(\mathbf{x})) = \frac{1}{1 + \exp(Af(\mathbf{x}) + B)} \tag{2}$$

The parameters A and B are found by minimizing the negative log-likelihood of the validation set

$$\begin{aligned} \min_{A,B} - \sum_{i=1}^{N} (t_i \log(P(y = 1|f(\mathbf{x}_i)) \\ + (1 - t_i) \log(1 - P(y = 1|f(\mathbf{x}_i))))), \end{aligned} \tag{3}$$

where

$$t_i = \begin{cases} \frac{N_+ + 1}{N_+ + 2}, & \text{if } y_i = 1 \\ \frac{1}{N_- + 2}, & \text{if } y_i = -1 \end{cases}$$

N_+ is the number of positive class labels and N_- presents the negative ones.

Based on one-vs-one classification, pairwise coupling (PC) is a method of combining multiple two-class probabilities to obtain multi-class estimates for C classes. The method was proposed by [30] and extended by [31]. Let r_{ij} be the probabilistic output of the classifier, obtained, e.g., using Platt's method, and p_i be the probability of the i:th class. Also, let p_i be presented by auxiliary variables $\mu_{ij} = p_i/(p_i + p_j)$. To estimate the values of p_i, the Kullback-Leibler divergence between r_{ij} and μ_{ij} can be determined as follows

$$l(\mathbf{p}) = \sum_{i<j} n_{ij} (r_{ij} \log \frac{r_{ij}}{\mu_{ij}} + (1 + r_{ij}) \log \frac{1 - r_{ij}}{1 - \mu_{ij}}) \tag{4}$$

where the weight n_{ij} is the number of examples of classes i and j in the training set. The weights n_{ij} can be set equal to one if there is no significant difference between class sizes. Minimizing the function in Eq. 4, can be computed using an iterative method [30]. Finally, p_i presents the conditional probability $P(c_i|f(\mathbf{x}))$ of recognizing class i. For example, [32] achieved encouraging multi-class classification results using the methods described above.

3.3 Temporal Smoothing of Sequences

Regardless of SVM's capability of classifying independently and identically distributed (i.i.d) data as presented in the previous subsection, it is not directly applicable to sequential data, such as activities where the data are rather dependent on the neighborhood labels. This subsection presents a general algorithm to train temporal smoothing to the confidence valued outputs of a discriminative (or generative) classifier trained on static independent examples.

The learning of sequential input-output pairs has usually been done with hidden Markov models (HMM) [9], which are generative graphical models with a Markov chain structure. HMMs have some limitations compared with kernel-based methods: they are trained in a generative manner (e.g., one model/class), they have some conditional independence assumptions, they need explicit feature presentation (e.g., suffering from the curse of dimensionality), and they cannot handle overlapping features. To overcome the limitations of HMMs, many discriminative variants have been proposed, including different discriminative training algorithm for HMMs (see section 2 for some of the related approaches).

We propose a simple algorithm that combines discriminative multi-class learning with generative smoothing of activity sequences, named discriminative temporal smoothing (DTS). DTS is a general algorithm in which you can use any base classifier that produces confidence output measurements. However, we applied SVM due to its accurate and efficient sparse solution. Once we have trained the SVM classifiers on the static examples and mapped them to confidence values, we can apply temporal smoothing. First, the probabilistic outputs of the static classifier from the training set is used as an observation input to estimate a global transition probability between class labels. Let $P(c_k|f(\mathbf{x}_1)), P(c_k|f(\mathbf{x}_2), \ldots, P(c_k|f(\mathbf{x}_t)$ be a sequence of conditional posterior probabilities of class k from the beginning of the sequence to a time step t estimated by SVMs and pairwise coupling. We collect these confidence values from every k class to observation matrix $\mathbf{B}$ as follows

$$\mathbf{B} = \begin{bmatrix} P(c_1|f(\mathbf{x}_1) & P(c_1|f(\mathbf{x}_2) & \ldots & P(c_1|f(\mathbf{x}_t) \\ P(c_2|f(\mathbf{x}_1) & P(c_2|f(\mathbf{x}_2) & \ldots & P(c_2|f(\mathbf{x}_t) \\ \vdots & \vdots & \vdots & \vdots \\ P(c_k|f(\mathbf{x}_1) & P(c_k|f(\mathbf{x}_2) & \ldots & P(c_k|f(\mathbf{x}_t) \end{bmatrix} \tag{5}$$

Then, a global transition matrix $\mathbf{A}$ with transition coefficients $a_{ij} = P(c_i^t|c_j^{t-1})$ (the probabilities between different classes i and j from the time $t-1$ to t) is calculated. The transition coefficients can be estimated with an iterative forward-backward algorithm, well-known from HMM training [9], over the observation matrix. Finally, an unknown sequence can be labeled from coupled probabilistic SVM confidence outputs with the use of a transition probability matrix and a Viterbi algorithm [10], resulting in smoothed class probabilities ($P_s(c_{1...k}|f(\mathbf{x}_t)) = P_s(c_1|f(\mathbf{x}_t)), P_s(c_2|f(\mathbf{x}_t)), \ldots, P_s(c_k|f(\mathbf{x}_t))$ for example $\mathbf{x}$ at time t). The final classification is made by choosing the most probable class from the smoothed confidence values, i.e., $\mathrm{argmax}[P_s(c_{1...k}|f(\mathbf{x}_t))]$. A diagram of different stages of the proposed activity recognition system based on DTS is presented in Figure 4.

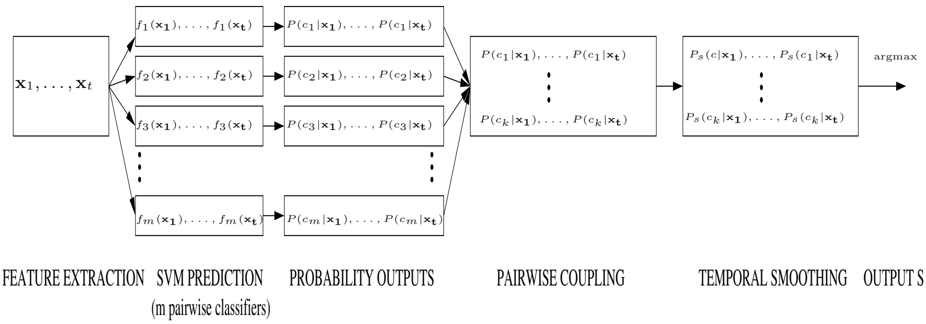

Fig. 4. Example diagram of the building blocks of a system for learning to recognize sequential activities

4 Results

This section presents the results of our experiments using the data sets of 17 and 9 activities, respectively. Both data sets consist of the data of 13 subjects. We removed 5 examples (i.e., 3.5 seconds) from the beginning of each activity due to outliers caused by the subject moving from the labeling screen to a spot (e.g., white board) to perform the particular activity. To compare our results with previous work by [11], we use a similar testing scheme, i.e., we presented the test results with four-fold cross-validation (i.e., we used independent test data sets not used for training). In addition, we compare the proposed method with conventional SVM and HMM classifiers as well as a SVM-HMM combination quite similar to the approach presented by [2]. However, they use a different discriminative method (i.e., ADABoost) as the base classifier.

For each data set we trained the ensemble of one-vs-one SVM classifiers (Eq. 1) with radial basis function (RBF) kernels ($K(\mathbf{x}, \mathbf{x}') = exp(-(1/2\sigma^2)||\mathbf{x} - \mathbf{x}'||^2)$). The regularization penalty term $C = \{0.5, 1, 10, 100\}$ and kernel hyperparameter $\sigma = \{0.6, 0.8, 1.0, 1.5, 2\}$ were found using four-fold cross-validation over the training data sets. Furthermore, the parameters of sigmoid mapping (Eq. 2) were estimated by cross-validation of each binary classifier. Pairwise classifiers were finally coupled to give confidence values for each class (Eq. 4). These conditional probabilities were used as an input to different methods: SVM-HMM and DTS, respectively. The structure of HMM (in conventional and SVM-HMM methods) included three hidden states, and the observation probability distributions were presented using a two-component Gaussian mixture model with a diagonal covariance matrix. A single HMM was trained for each class using five consecutive examples in a sliding window, and this sequence was classified as the highest likelihood value among the models. The models were implemented in a Matlab environment. The SVMs were trained using a Spider toolbox [33] with a libSVM optimizer [34], and the HMMs were trained using the HMM Matlab toolbox [35].

4.1 Recognition Results

Table 1 presents the total recognition accuracies of 17 activities using different classification methods as well average precision (true positive/(true positive + false positive)) and recall (true positive/(true positive + false negative)) values. The proposed method surpassed all other methods, presenting a 93.6% total recognition rate. Additionally, these experiments show the usefulness of the discriminative SVM classifier, as it gives superior accuracy compared with HMM, which is not able to model a high-dimensional input space accurately. Using the SVM-HMM combination gives a slightly better recognition rate compared with plain HMM, but it is not as effective as presented by [2]. This is related to the fact that besides accelerometers, they used different sensors and features such as audio, which usually includes a lot of temporal dynamics in intra-class variations. In addition, they used a much larger sliding window to extract features in which the usefulness of modeling the hidden dynamics of a single activity is justified. In our experiments, a simple global transition probability smoothing machine works well with simple statistical features and a small sliding window.

Table 1. Total recognition accuracies as well as average precision and recall values of 17 activities using different methods

	SVM	HMM	SVM-HMM	DTS
Accuracy (%)	90.65 (4.53)	84.26 (4.66)	84.39 (5.65)	**93.58 (4.15)**
Precision (%)	88.00 (4.68)	75.69 (3.04)	77.82 (5.36)	**93.88 (3.69)**
Recall (%)	87.74 (3.21)	79.74 (3.76)	81.17 (3.90)	**90.58 (3.55)**

Table 2 presents the total recognition accuracies of 9 activities using different classification methods as well average precision and recall values. Also, in this case the DTS method outperformed the other methods, showing a 96.4% success rate. Similar conclusions can be made with a data set of 17 activities.

Table 2. Total recognition accuracies as well as average precision and recall values of 9 activities using different methods

	SVM	HMM	SVM-HMM	DTS
Accuracy (%)	94.15 (2.62)	88.75 (2.93)	90.42 (4.75)	**96.36** (2.13)
Precision (%)	92.12 (2.98)	82.32 (4.50)	85.77 (3.14)	**96.76** (2.06)
Recall (%)	92.10 (1.80)	86.77 (3.74)	87.89 (7.20)	**94.53** (1.05)

Finally, we examined the individual activities in the data set of 9 activities. Table 3 presents an example confusion matrix of a total number of 4405 test examples of 9 activities performed by 13 subjects recognized by a DTS algorithm. All the activities, except using stairs, are recognized at over a 90% success rate, where the most distinguished ones are: *sitting*, *walking*, *running*, and *bicycling*. The *using stairs* activity is naturally most often confused with *walking*, which is not the case the other way around.

Table 3. Confusion matrix of recognizing 9 different activities with a discriminative temporal smoothing algorithm

%	clean	sit	stand	use stairs	brush teeth	lie down	walk	run	cycle
clean	**94.3**	1.5	1.2	0.0	0.0	0.0	2.4	0.0	0.6
sit	0.0	**99.4**	0.4	0.0	0.0	0.0	0.2	0.0	0.2
stand	3.1	2.6	**94.1**	0.0	0.2	0.0	0.0	0.0	0.0
use stairs	0.0	0.0	0.0	**70.9**	0.0	0.0	29.1	0.0	0.0
brush teeth	1.7	0.7	0.0	0.0	**97.2**	0.4	0.0	0.0	0.0
lie down	3.4	3.4	0.0	0.0	0.0	**92.7**	0.0	0.0	0.5
walk	0.0	0.0	0.0	0.2	0.0	0.0	**99.8**	0.0	0.0
run	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0
cycle	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**99.6**

In comparison, using the same data sets and features, the k-nearest neighbor classifier used by [11] gives total recognition accuracies 89.47% (4.64) for the data set of 17 subjects and 93.02% (2.64) for the data set of 9 subjects, respectively. In both dataset it is more accurate than the HMM and SVM-HMM methods, but DTS also outperforms those methods.

5 Conclusions

We presented a novel approach to activity recognition by wearable sensors. The proposed algorithm combines effective discriminative classification with a smoothing of adjacent class label estimates in an activity sequence. In activity recognition, it is very useful to extend conventional i.i.d data assumption-based classifiers to the sequential learning domain to be able to take advantage of the smoothing changes of the targets and the probabilities of transition between different activities.

We used a SVM classifier to recognize individual activity examples, which were then mapped to class confidence values. At the post-processing stage we trained a global transition probability matrix from the confidence values using a forward-backward algorithm. Final classification was then performed with the confidence values and the transition probability matrix using a Viterbi algorithm.

Using a data set of 13 subject performing 17 daily activities, we were able to achieve a total accuracy of 94%. In addition, we combined some of the most specific classes to present more generic activities, which led to 9 activities. The data set of combined activities gave a recognition rate of 96%. Our results indicate that the proposed algorithm is able take advantage of the sequential nature of activity data, showing superior performance compared with typical non-sequential (standard SVM) and sequential (standard HMM) classifiers. The accurate recognition of human activities can provide useful knowledge for different applications in the ubiquitous computing field.

The method proposed in this paper is general. It is not restricted to SVM-based classifiers but applies to any method that is able to produce probabilistic outputs. For example, Gaussian process classification leads naturally to probabilistic class confidence values compared with the more ad-hoc method used with SVM. In addition, the cross-validation can be replaced by Bayesian model selection strategies. This is one possible direction for future research. Furthermore, in this study we used a data set that was collected in a semi-naturalistic manner, i.e., the subjects performed and labeled the activities in a predefined order to minimize possible disturbance. Using a data set where the activities are performed in a more naturalistic order, e.g., sitting-standing-walking, the sequential method is more advantageous than conventional methods due to its ability to place importance on more probable transitions between activities in real-life sequences.

Acknowledgments

The authors would like to thank K. Fujinami for the sensors and data collection, as well as the voluntary subjects at Waseda University. J.S. would like to thank InfoTech Oulu Graduate school for financial support. Additionally, this work was funded by the Academy of Finland and the Finnish Funding Agency of Technology.

References

1. Bao, L., Intille, S.: Activity recognition from user-annotated acceleration data. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 1–17. Springer, Heidelberg (2004)
2. Lester, J., Choudhury, T., Kern, N., Borriello, G., Hannaford, B.: A hybrid discriminative/generative approach for modeling human activities. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, pp. 766–722 (2005)
3. Ward, J.A., Lukowicz, P., Troster, G., Starner, T.E.: Activity recognition of assembly tasks using body-worn microphones and accelerometers. IEEE Trans. Pattern Anal. Mach. Intell. 28(10), 1553–1567 (2006)
4. Herbrich, R.: Learning Kernel Classifiers. The MIT Press, USA (2002)
5. Schölkopf, B., Smola, A.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press, USA (2001)
6. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, UK (2004)
7. Jebara, T.: Machine Learning: Discriminative and Generative. Kluwer Academic Publishers, Dordrecht (2004)
8. Ng, A.Y., Jordan, M.I.: Classification with hybrid generative/discriminative models. In: NIPS 2001. Advances in Neural Information Processing Systems, vol. 14 (2001)
9. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2), 257–286 (1989)

10. Forney, G.D.: The viterbi algorithm. Proceedings of the IEEE 61, 268–277 (1973)
11. Pirttikangas, S., Fujinami, K., Nakajima, T.: Feature selection and activity recognition from wearable sensors. In: Youn, H.Y., Kim, M., Morikawa, H. (eds.) UCS 2006. LNCS, vol. 4239, pp. 516–527. Springer, Heidelberg (2006)
12. Pentland, A.: Smart rooms. Scientific American 274, 68–76 (1996)
13. Essa, I.A.: Ubiquitous sensing for smart and aware environments: Technologies towards the building of an aware home. IEEE Personal Communications, Special issue on networking the physical world, pp. 47–49 (October 2000)
14. Rhodes, B.J., Minar, N., Weaver, J.: Wearable computing meets ubiquitous computing: reaping the best of both worlds. In: Proc. Third International Symposium on Wearable Computers. Digest of Papers, San Francisco, California, USA, pp. 141–149 (October 18-19, 1999)
15. Huynh, T., Schiele, B.: Analyzing features for activity recognition. In: sOc-EUSAI 2005: Proceedings of the 2005 joint conference on Smart objects and ambient intelligence, pp. 159–163. ACM Press, New York (2005)
16. Maurer, U., Smailagic, A., Siewiorek, D.P., Deisher, M.: Activity recognition and monitoring using multiple sensors on different body positions. In: BSN 2006. Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks, pp. 113–116. IEEE Computer Society, Washington, DC (2006)
17. Kern, N., Schiele, B.: Multi-sensor activity context detection for wearable computing. In: European Symposium on Ambient Intelligence, Eindhoven, The Netherlands (November 2003)
18. Mäntylä, V.-M., Mäntyjärvi, J., Seppänen, T., Tuulari, E.: Hand gesture recognition of a mobile device user. In: ICME 2000. IEEE International Conference on Multimedia and Expo., New York, USA, vol. 1, pp. 281–284 (2000)
19. Brashear, H., Starner, T., Lukowicz, P., Junker, H.: Using multiple sensors for mobile sign language recognition. In: ISWC 2003. Proceedings of the 7th IEEE International Symposium on Wearable Computers, pp. 45–52. IEEE Computer Society, Washington, DC (2003)
20. Van Laerhoven, K., Cakmakci, O.: What shall we teach our pants? In: ISWC 2000: Proceedings of the 4th IEEE International Symposium on Wearable Computers, p. 77. IEEE Computer Society, Washington, DC (2000)
21. Chieu, H.L., Lee, W.S., Kaelbling, L.P.: Activity recognition from physiological data using conditional random fields. In: Singapore-MIT Alliance Symposium (2006)
22. Naphade, M.R., Wang, R., Huang, T.S.: Classifying motion picture soundtrack for video indexing. In: Proceedings of IEEE International Conference on Multimedia and Expo., pp. 1160–1163. IEEE Computer Society Press, Los Alamitos (2001)
23. Dietterich, T.: Machine learning for sequential data: A review. In: Caelli, T.M., Amin, A., Duin, R.P.W., Kamel, M.S., de Ridder, D. (eds.) SPR 2002 and SSPR 2002. LNCS, vol. 2396, pp. 15–30. Springer, Heidelberg (2002)
24. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS 1998. Advances in Neural Information Processing Systems, vol. 11 (1998)
25. Altun, Y., Tsochantaridis, I., Hofmann, T.: Hidden markov support vector machines. In: ICML 2003. Proceedings of the International Conference of Machine Learning (2003)
26. Altun, Y., Hofmann, T., Smola, A.: Gaussian process classification for segmenting and annotating sequences. In: ICML 2004. Proceedings of the International Conference of Machine Learning (2004)
27. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, UK (2000)
28. Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Smola, A., Bartlett, P., Schölkopf, B., Schuurmans, D. (eds.) Advances in Kernel Methods - Support Vector Learning, pp. 185–208. MIT Press, Cambridge (1999)

29. Platt, J.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Smola, A., Bartlett, P., Schölkopf, B., Schuurmans, D. (eds.) Advances in Kernel Methods - Support Vector Learning, pp. 61–74. MIT Press, Cambridge (1999)
30. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. In: Jordan, M.I., Kearns, M.J. (eds.) NIPS 1998. Advances in Neural Information Processing Systems, vol. 10, MIT Press, Cambridge (1998)
31. Wu, T.-F., Lin, C.-J., Weng, R.C.: Probability estimates for multi-class classification by pairwise coupling. Journal of Machine Learning Research 5, 975–1005 (2004)
32. Duan, K.-B., Keerthi, S.: Which is the best multiclass SVM method: An empirical study. In: Oza, N.C., Polikar, R., Kittler, J., Roli, F. (eds.) MCS 2005. LNCS, vol. 3541, pp. 278–285. Springer, Heidelberg (2005)
33. The Spider: Machine learning toolbox for matlab. Software available: `http://www.kyb.tuebingen.mpg.de/bs/people/spider/`
34. Chang, C.-C., Lin, C.-J.: LIBSVM: A Library for Support Vector Machines, Software available (2001), `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`
35. Hidden Markov Model (HMM) toolbox for matlab. Software available: `http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html`