# Activity recognition from on-body sensors by classifier fusion: sensor scalability and robustness

Piero Zappi*, Thomas Stiefmeier, Elisabetta Farella*, Daniel Roggen, Luca Benini*, Gerhard Tröster
Wearable Computing Lab, ETH Zürich, Switzerland
*Department of Electronic Informatic and System, Bologna, Italy
Email: {stiefmeier, roggen, troester}@ife.ee.ethz.ch
*{pzappi, efarella, lbenini}@deis.unibo.it

## Abstract

*Activity recognition from on-body sensors is affected by sensor degradation, interconnections failures, and jitter in sensor placement and orientation. We investigate how this may be balanced by exploiting redundant sensors distributed on the body. We recognize activities by a meta-classifier that fuses the information of simple classifiers operating on individual sensors. We investigate the* robustness *to faults and* sensor scalability *which follows from classifier fusion. We compare a reference majority voting and a naive Bayesian fusion scheme. We validate this approach by recognizing a set of 10 activities carried out by workers in the quality assurance checkpoint of a car assembly line. Results show that classification accuracy greatly increases with additional sensors (50% with 1 sensor, 80% and 98% with 3 and 57 sensors), and that sensor fusion implicitly allows to compensate for typical faults up to high fault rates. These results highlight the benefit of large on-body sensor network rather than a minimum set of sensors for activity recognition and prompts further investigation.*

## 1. INTRODUCTION

Wearable computing aims to provide proactive support by exploiting the knowledge of the user's context, determined from on-body sensors [1], [2]. The activities and manipulative gestures of a user are an important aspect of the context. The recognition of activities and gestures from body-worn sensors may be applied e.g. in human-computer interfaces [3], [4], sports [5], and also for context-aware support in automotive production [6], [7], [8].

The recognition of gestures or activities from body-worn sensors is generally carried out classifying features extracted from a number of sensors that capture the limb motion, such as Inertial Measurement Units (IMUs) or accelerometers. Most activity recognition methods tend to assume the following:

- Sensors placed at an "optimal" body locations for the activities to detect. Variation in sensor placement over time are proscribed as they affect classification.
- The number of sensor is minimized in order to reduce obtrusiveness (e.g. manipulative gestures may be detected from few IMUs placed on limbs and back).
- The sensors characteristics remain constant (i.e. no sensor degradation) and they do not fail.
- Sensor interconnections are reliable.

These assumptions are challenged when seeking extreme miniaturization and integration. Although it allows a dense on-body sensor placement, this may be at the expense of sensor accuracy or robustness, or interconnection reliability. Textile sensing elements are subject to high mechanical stress (e.g. during washing or when worn) which may lead to sensor degradation and faults. Networks of miniature body-worn wireless sensors (such as those used in [9]) may suffer from radio interferences, as well as occlusions caused by body-parts, thereby causing data rate reduction or data loss. In order to avoid relative motion, sensors attached on the body require the use of tight-fitting clothes [10] or relatively high attachment pressure, which limits comfort.

In a general setting the sensor network characteristics may thus change in unpredictable ways due to sensor degradation, interconnection failures, and jitter in the sensor placement. The use of a dense mesh of sensors distributed on the body may allow to overcome these challenges through sensor fusion techniques [11], [12], [13]. As part of an ongoing project, we wish to investigate activity recognition in this challenging context of large and dynamically changing sensor networks.

We investigate the use of sensor fusion techniques for gesture recognition. A meta-classifier fuses the information of simple classifiers operating on individual sensors. We investigate the outcomes of classifier fusion in function of the number of sensors on the recognition performance (*sensor scalability*), and on the robustness to faults (*robustness*).

The paper is organized as follows. We introduce our approach to activity recognition in section 2 and reference related works. In section 3 we describe the experimental setup. The results are presented in section 4. We discuss results in section 5 and conclude in section 6.

## 2. ACTIVITY RECOGNITION WITH ON-BODY SENSOR BY FUSION OF SIMPLE CLASSIFIERS

This work is part of an ongoing effort to explore activity recognition in large networks of simple sensors distributed on the body. Energy constraints favors low-power and miniature sensors, such as MEMS accelerometers or textile stretch sensors in contrast to inertial measurement units (which combine accelerometers with gyroscopes and magnetometers) which are more power hungry. These sensors may be unreliable or subject to faults. Their placement may be subject to jitter

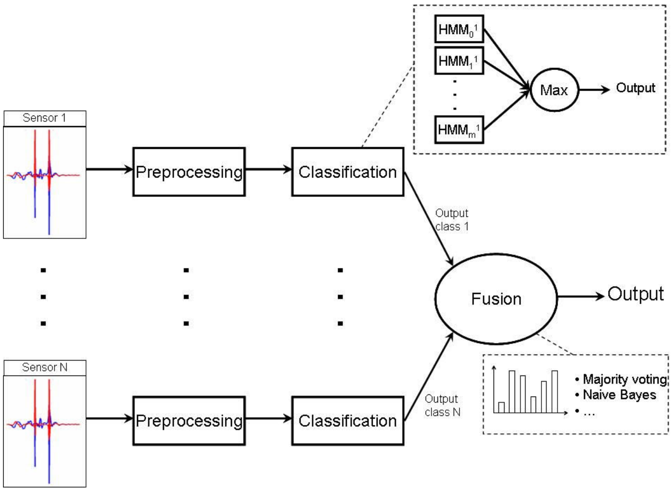

**Fig. 1:** Activity recognition architecture. Features extracted from the sensor data are classified by competing hidden Markov models, each one trained to model one activity class. The most likely model yields the class label. These labels are fused to obtain an overall classification result. A naive Bayesian scheme and a majority voting scheme are used.

during operation due to repeated body motion or the use of loose fitting clothes. We envision activity recognition in a *dynamic* sensor network where the number and type of nodes may even change over time, with sensors added or removed at run-time. Here we consider an homogeneous network as a first step. We investigate how the information of sensors distributed in the body may be fused to recognize activities, and what are the outcomes of sensor fusion in terms of recognition performance and robustness to faults.

Several work address the use of meta-classifier to increase the performances of classification techniques in presence of noise. In [14] the output of a classifier, seen as a ranking of the classes, is fused to obtain better overall classification. Another work presented in [15] fuse the output of $n$ sensors whose input are affected by noise using minimum distance correcting code. In these work sensor fusion does not cope with network of variable size (such as when a faulty sensor is removed from the system). Also faults are limited to “stuck-at” models.

Activity recognition from body-worn sensors consists typically of: sensor signal acquisition, extraction of features from the signal, and feature classification. Sensor fusion may be performed at each level [16]. Here we fuse data at the classifier level with a meta-classifier. This allows the fusion of heterogeneous sensors, with each sensor domain processed with specific algorithms.

The outline of our architecture is illustrated in figure 1. Sensor data is first acquired and preprocessed. Preprocessing consists of feature extraction. Features are classified individually for each sensor, leading to class labels. Finally these class labels are fused, which yields the likely activity class corresponding to sensor data.

### A. Classification of activities from accelerometers

We use three-axis accelerometers for activity recognition because they are small and inexpensive. The signal features are the acceleration direction (positive, negative, or null) obtained by *ternarizing* the acceleration amplitude with two thresholds.

The common approaches to handle the temporal dynamics of gestures and classify them include hidden Markov models (HMMs), dynamic time warping [17] and neural networks [18], [19]. We use hidden Markov model as they showed to be a good approach in previous work [20], [8].

As many HMMs as activity classes are defined and trained to model the activity classes. These HMMs compete and the one modeling best the features indicates the class label (see figure 1). We use ergodic (fully connected) discrete HMMs with 4 hidden states. The possible observations are the 3 acceleration features. Training is performed by optimizing the HMM parameters with the Baum-Welch algorithm starting from HMMs with randomly initialized parameters. For each model, optimization was repeated 15 times with a different random initialization and the HMM modeling best the target gesture was selected. The model likelihood is estimated using the Forward algorithm. We used the Kevin Murphy HMM Toolbox for this purpose.

So far we do not consider the segmentation of input data: we focus on the recognition of isolated activities.

Classification is done on a desktop computer. It could also be performed independently by each network node. Future work may assess classification algorithms in relation to their computational complexity, and investigate their implementation on low cost, low power devices (see [9] for contributions in this direction). Other classifiers do not affect the qualitative outcomes of sensor fusion investigated here.

### B. Classifier fusion

Among classifier fusion methods [21], we consider a majority voting scheme as a reference and a naive Bayesian fusion method. These methods are tractable for wearable systems and cope with a change in the number of sensors, such as when a sensor fails, without needing any retraining.

Majority voting is a simple way to fuse the classifiers that is used as a baseline. It may suffer of higher degradation in noisy environment because all the sensors are weighted identically for all the classes, without using previous statistic. The label resulting from the classifier fusion is simply the class label that occurs the most after individual sensor classification.

The naive Bayesian classifier is a simple probabilistic classifier based on Bayes’ theorem. The classifier combines the Bayes probabilistic model with a decision rule. A common rule is to classify an input instance as belonging to the class that maximize the *a posteriori* probability. Previous work [22] showed that naive Bayesian classifiers perform well even if the independence assumption is not met.

Given the conditional model $P(C|A_0, A_1, ..., A_n)$, where C denotes the class and $A_i$ $n$ input attributes, and using Bayes theorem we can define:

$$P(C|A_1, A_2, ..., A_n) = \frac{P(A_1, A_2, ..., A_n|C)\, P(C)}{P(A_1, A_2, ..., A_n)}$$

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Marginal}} \tag{1}$$

Where the *Posterior* is the probability of a certain class given the input sequence, *Likelihood* is the conditional probability of a certain sequence given a certain class, *Prior* is the prior probability of the selected class, and *Marginal* is the probability of the input sequence.

Applying the assumption that the input attributes are independent we can write:

$$P(C|A_1, A_2, ..., A_n) = \frac{P(C) \prod_{i=1}^{n} P(a_i|C)}{P(A_1, A_2, ..., A_n)} \quad (2)$$

According to the proposed decision rule we can finally state:

$$C_{out}(a_1, a_2, ..., a_n) = \\ argmax_c \frac{P(C = c) \prod_{i=1}^{n} P(A_i = a_i|C = c)}{P(A_1 = a_1, A_2 = a_2, ..., A_n = a_n)} \quad (3)$$

Since the denominator in equation 3 is constant for every class we only need to compute the numerator. In our setup all gestures are occur with the same probability so the *Prior* probability (i.e. the frequency of each class) is $\frac{1}{\sharp\, classes}$ and only the *Likelihood* is computed. This parameter is obtained during the training phase, by building the confusion matrix for the HMM classifiers of each node and defining $P(A_i = a_i|C = c) = \frac{t_c}{t}$, where $t_c$ is the number of training instances for which the class $C = c$ and the attribute $A_i = a_i$ and $t$ is the number of training instances for class $c$. When for a class $c$ we do not have a sample for which $A_i = a_i$, $\prod_{i=1}^{n} P(A_i = a_i|C = c)$ for that class is always zero, despite the value of the other input attribute. For this reason we use the M-estimate of the likelihood:

$$P(A_i = a_i|C = c) = \frac{t_c + m\,p}{t + m} \quad (4)$$

where $p$ is an *a priori* estimation of $P(A_i = a_i|C = c)$ and $m$ is a user specific value. Typical choice for p is $\frac{1}{\sharp\, A\, values}$ (0.1 in our work), while we fixed $m = 1$.

In our system the number of input attributes changes when nodes fail. The posterior probability is then calculated including only the contributions of the available HMM outputs.

## 3. EXPERIMENTAL SETUP: ACTIVITIES IN CAR MANUFACTURING

In order to assess our approach, we consider the recognition of the activities of automotive workers. The knowledge of workers' activities enables context-aware support [6], [7], [8]. In the last stage of car production, workers check the car quality and functionality. We consider the recognition of 10 activity classes (table 1) in this quality assurance checkpoint. They are a subset of 46 activities identified in previous work, that form the complete checking procedure [23].

Data from 19 body-worn acceleration sensors nodes were recorded while a user performed these activities. The sensor nodes were placed at regular intervals along the left and right arm, on the inner and outer sides (see fig. 2). Regular placement ensures the generality of the results and avoid designer bias. A sensor node contains a 3 axis accelerometer (Analog Device ADXL330). Nodes are connected over USB hubs to a laptop PC that performs data acquisition[1]. Each accelerometer axis on the node is processed as an independent sensor, and will be called "sensor" hereafter. Thus the total number of classifier outputs to be fused in the last stage (figure 1) may be up to 57 (N=57).

**TABLE 1:** LIST OF ACTIVITY CLASSES TO RECOGNIZE FROM BODY-WORN SENSORS.

| Class | Description |
|---|---|
| 0 | write on notepad |
| 1 | open hood |
| 2 | close hood |
| 3 | check gaps on the front door |
| 4 | open left front door |
| 5 | close left front door |
| 6 | close both left door |
| 7 | check trunk gaps |
| 8 | open and close trunk |
| 9 | check steering wheel |

**Fig. 2:** Picture during test of Class 1. Highlighted the Node used.

For each activity class a single subject performed 19 repetitions of each activity, yielding 190 gesture instances. Data was synchronously labeled during recording by an experimenter who marked the beginning and the end of every instance by pressing laptop keys. This dataset is used for training and validation. We divided it in 4 folds: for each fold 14 nodes are used to train HMM and generate the confusion matrix for the Bayesian fusion and 5 nodes are used for testing.

## 4. TEST AND RESULTS

### A. *Single sensor classification*

Acceleration is ternarized with a threshold to provide the features for the HMM. We test the classification accuracy obtained from a single sensor in function of this threshold. To ensure the generality of the results, we consider each sensor individually and derive the average accuracy (fig. 3). The accuracy is maximized with thresholds of ±400mg.

[1] In future work we plan to use wireless sensor nodes, e.g. based on Bluetooth or Zigbee protocols. However for this experiment we wanted to ensure the acquisition of data of good quality for the initial investigations. The actual number of sensor on the body was 20, however one of the node did not collect data for some classes and we discarded it.

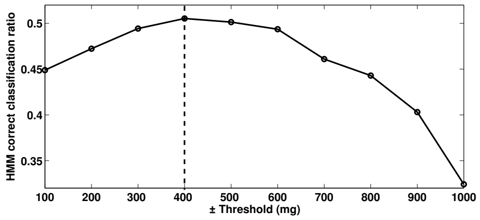

**Fig. 3:** Average correct classification ratio of single HMM as a function of the selected threshold.

### *B. Sensor scalability: classification with fusion*

We evaluate the classification accuracy as a function of the number of sensor nodes which are fused (cluster size). For each cluster size, from 1 to 19, we select a random subset from the available nodes. Clusters include all the accelerometer axis from this set (i.e. the number of sensors is a multiple of 3). For every cluster size we repeat the experiment 50 times, each time with a different random set of nodes, and average the results.

Figure 4 shows the classification accuracy for majority voting and Bayesian fusion. Also the a visual indication of the confusion matrix for clusters of 1, 7, 13 and 19 nodes using Bayesian fusion are presented in figure 5.

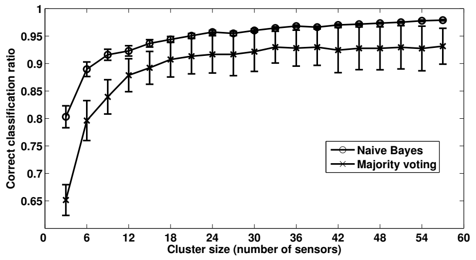

**Fig. 4:** Correct classification ratio as a function of the number of nodes used

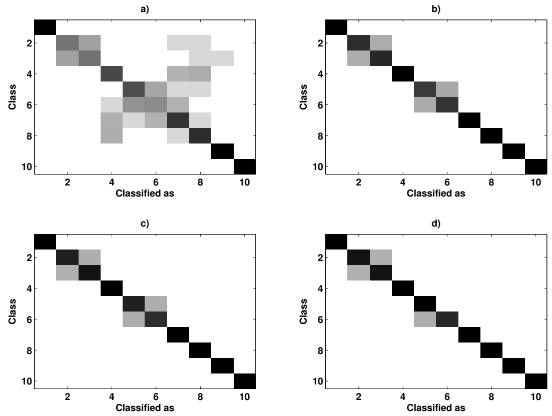

**Fig. 5:** Visual graph of the confusion matrix for different cluster size. a) 1 node; b) 7 nodes; c) 13 nodes; d) 19nodes.

The overall classification ratio increases as the number of nodes increases. With 19 sensors we achieve an average of 98% correct classification with Bayes fusion algorithm and 93% with majority voting. Bayesian fusion shows better results with respect to majority voting especially in small clusters. This is understandable because sensors with low classification accuracy are given more weight by majority voting than Bayesian fusion. In a small cluster their relative importance is thus higher than in a larger cluster, which limits performance.

### *C. Robustness to noise and faults*

We evaluate the performance of the fusion algorithm in the presence of noise and faults. To our knowledge no previous effort tried to model the noise that may affect body worn accelerometers. We define the 2 sources of noise below.

**Rotational noise.** The nodes, due to imperfect adherence to the body, may change their orientation around their attachment point. We model this by a rotation of the coordinate system of the accelerometer. We define the new coordinate system by successive application of a rotation along the X, Y, and Z axis. The raw acceleration is converted in this new coordinate system and presented to the HMM and the following fusion step. The rotation along each axis is random within the range $[0°; \alpha_{max}]$. We test the classification accuracy with $\alpha_{max}$ from 5° to 60° in 5° step. As for the sensor scalability, we repeat tests with random cluster of nodes of increasing size.

**Random noise.** In order to model a noise that seriously impact on HMM classifier (such as sensor fault, connection failure or misclassification) we change the label of a randomly selected number of "faulty" sensors with a random class. We test the fusion algorithm with 19 nodes and increase the number of affected sensors from 1 up to 57.

In figure 6a we show the correct classification ratio versus the maximum level of rotational noise, using a fixed number of sensors. In figure 6b we compare the results of the majority voting and naive Bayes fusion algorithm for 2 different cluster. Figure 7 shows the classification ratio as a function of size of cluster and noise level.

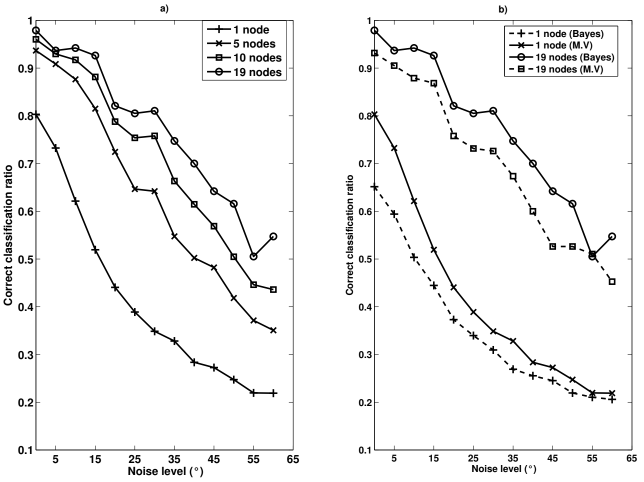

**Fig. 6:** a) Effect of rotational noise level on the fusion classification ratio for different cluster sizes.b) Comparison between naive Bayesian fusion and majority voting fusion for cluster size 1 and 19

Although larger noise levels decrease classification accuracy, sensor fusion however allows to reduce the impact of noise in larger sensor set and the performance remains relatively constant up to high fault rates. In figure 6a illustrates this. For example, if we use one node, on average we achieve 80% correct classification ratio without noise. If we add some noise, but still want to keep that level of accuracy, we must augment the number of nodes in cluster. For example, using 5 nodes we can tolerate up to 15° of noise, using 10 nodes

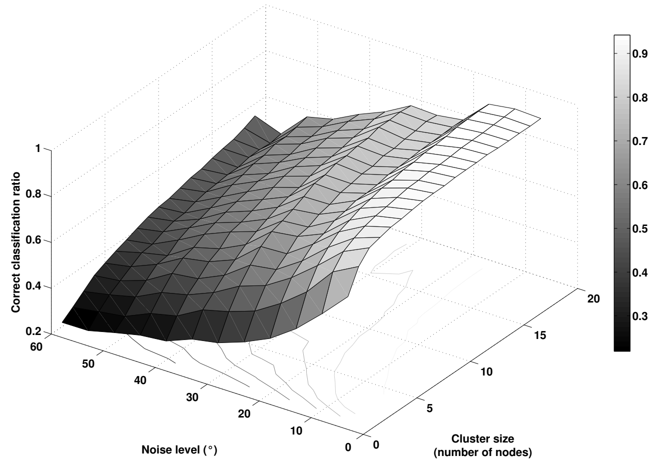

**Fig. 7:** Effect of both rotational noise level and cluster size on correct classification ratio (Bayesian fusion).

up to 20° and with the whole set of nodes up to 35° keeping the initial accuracy.

Although Bayesian fusion achieves better classification ratio, both fusion techniques show similar performance patterns with increased faults. Thus both approaches are equally robust to this kind of noise.

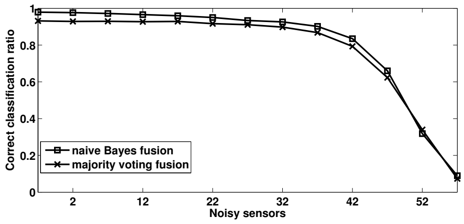

**Fig. 8:** Effect random noise on both majority and naive Bayes fusion method.

Figure 8 shows the correct classification ratio with respect to the number of sensors affected by random noise using clusters of 19 nodes (57 sensors) and both naive Bayes and majority voting fusion algorithms. Both fusion algorithms show good tolerance to this kind of noise and they achieve above 90% correct classification ratio for up to 42 (naive Bayes) and 32 (majority voting) faulty nodes, respectively 73% and 56% of the inputs. Also in this situation naive Bayes shows better classification ratio. When the number of faulty sensors reaches the maximum (57), both fusion methods shown similar classification ratio (slightly below 10%) because the inputs of the fusion algorithm are random.

## 5. DISCUSSION

In previous chapters we showed how it is possible to combine data from a large number of small and cheap accelerometers, easily integrable into garments, in order to achieve high gesture recognition accuracy even in presence of noisy data or unreliable sensors.

We compared two fusion techniques: majority voting and naive Bayesian. In all the tests naive Bayes fusion has achieved a higher correct classification ratio than majority voting. This is due to the fact that naive Bayes technique is able to gather useful information even from incorrect HMM classification output. In fact, if a sensor has an high probability to misclassify a certain class (for example class $m$ is often classified as class $n$) it is possible to exploit this information and interpret his output according to this biasing. This property may also be used in order to reduce complexity of the classification algorithm.

The minimum correct classification ratio required strictly depends on the application that is developed. However, is possible to define some bounds on the number of sensor needed to achieve such accuracy as a function of requirement and noise tolerance. For example, if we are using Bayesian fusion and our application need to achieve 90% correct classification we must use, at least, 3 nodes (9 sensors) randomly placed on the two arms. Also, if we assume that a slight rotational or random noise may affect the nodes, we must augment the number of sensor used. For example, if we want to tolerate a maximum of 10° rotational noise we need at least, 8 nodes (24 sensors) or, if we use all the 19 nodes we can tolerate that 37 sensors may produce a corrupted output or a 15° maximum rotational noise.

On the other hand it may be useful to use such information to vary the number of working sensor according to dynamic application constraint. For this objective we can augment the number of sensor used, and thus the correct classification ratio, only in critical situations and keep unused sensor in a low-power, idle state in order to increase network life.

Another thing that must be highlighted is the comparison between the correct classification rate when random noise is added and the correct classification ratio as the number of nodes decreases. In figure 9 such curves are plotted. As can be see, the classification ratio of the fusion algorithm decrease quicker when the output of the HMM is affected by noise instead than absent. Thus it will be important to develop fault detection algorithms able to recognize sensors with corrupted output and leave them out the fusion algorithm

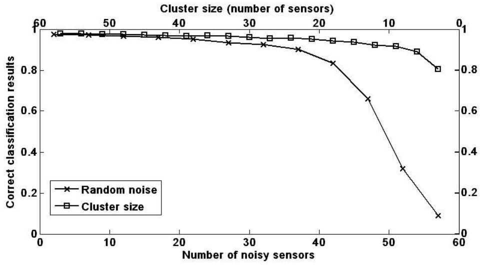

**Fig. 9:** Correct classification ratio as a function of cluster size or number of random noise affected sensors.

## 6. CONCLUSION AND FUTURE WORK

Wearable computers aim to provide to the user valuable information and support at anytime and anywhere. Among the others, gesture can be an important aspect of the users's "context" that can be used in several activity.

Automatic detection of gestures is usually carried out from a set of body worn motion sensors such as accelerometers and IMUs. However they rely on strict assumption about sensor position, orientation, number and characteristics and about interconnection reliability.

In this work, we investigated the implications of using classifier fusion techniques to perform activity recognition from a number of sensors distributed on the body in terms of sensor scalability and robustness to faults. This approach does not depend on the previously listed constraints. It allows to achieve high correct classification ratio using only accelerometers despite the fact that they may reduce in number (due to fault or to application requirements, as pointed in section 5), change orientation or be affected by noise.

Our approach is based on a two steps process: in the first step each accelerometer classify the gesture using a set of HMMs (isolated recognition), then the vector with the classification output from each sensor is fused using a either majority voting mechanism or a discrete naive Bayes classifier.

Within the 10 activity classes considered we have compared the classification accuracy with a variable number of sensors randomly placed on upper limbs. We have shown that although the individual classification accuracy remains low (50% on average) the classification accuracy can be improved with already 3 sensor to 80%, and up to 98% with all the 57 sensors.

Although we did not include any fault detection mechanism, we show that we obtain an implicit robustness to faults thanks to sensor fusion. The implicit robustness to faults, together with the increase in classification accuracy allowed by sensor fusion, motivate further research.

In future work, we want to investigate energetic aspects related to activity recognition in such sensor network, as well as algorithmic complexity and performance tradeoffs. In particular, we wish to investigate whether simplifying activity classification and sensor fusion methods while increasing the number of sensors makes sense from an energetic and performance viewpoint.

Finally we wish to investigate the technical implementation of such activity recognition in wireless sensor networks using a framework allowing activity recognition algorithms to be distributed among wireless sensor nodes [24].

## 7. ACKNOWLEDGEMENTS

Part of this work is funded by EU project n°004216 WearIT@Work.

## REFERENCES

[1] A. K. Dey and G. D. Abowd, “Towards a better understanding of context and context awareness,” Georgia Tech, Tech. Rep. GITGVU-99-22, 1999.

[2] P. Lukowicz, H. Junker, M. Staeger, T. von Bueren, and G. Troester, “WearNET: A distributed multi-sensor system for context aware wearables,” in *UbiComp 2002: Proceedings of the 4th International Conference on Ubiquitous Computing*, Sept. 2002, pp. 361–370.

[3] H. Kang, C. W. Lee, and K. Jung, “Recognition-based gesture spotting in video games,” *Pattern Recognition Letters*, vol. 25, no. 15, pp. 1701–1714, 2004.

[4] S. Kallio, J. Kela, P. Korpipää, and J. Mäntyjärvi, “User independent gesture interaction for small handheld devices.”

[5] G. Chambers, S. Venkatesh, G. West, and H. Bui, “Hierarchical recognition of intentional human gestures for sports video annotation,” in *Proc. 16th IEEE Conference on Pattern Recognition*, 2002, pp. 1082–1085.

[6] I. Maurtua, P. T. Kirisci, T. Stiefmeier, M. L. Sbodio, and H. Witt, “A wearable computing prototype for supporting training activities in automative production,” in *4th International Forum on Applied Wearable Computing (IFAWC)*, 2007.

[7] T. Stiefmeier, C. Lombriser, D. Roggen, and G. Tröster, “Event-based activity tracking in work environments,” in *Third International Forum on Applied Wearable Computing*, 2006.

[8] T. Stiefmeier, G. Ogris, H. Junker, P. Lukowicz, and G. Tröster, “Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario,” in *10th IEEE International Symposium on Wearable Computers*, 2006.

[9] D. Roggen, N. B. Bharatula, M. Stäger, P. Lukowicz, and G. Tröster, “From sensors to miniature networked sensorbuttons,” in *Proc. of the 3rd Int. Conf. on Networked Sensing Systems (INSS06).*

[10] C. Mattmann, O. Amft, H. Harms, G. Tröster, and F. Clemens, “Recognizing upper body postures using textile strain sensors,” in Accepted for publication in *Proc. of the 11th IEEE International Symposium on Wearable Computers (ISWC)*, 2007.

[11] K. Van Laerhoven, K. Aidoo, and S. Lowette, “Real-time analysis of data from many sensors with neural networks,” in *Proceedings of Fifth International Symposium on Wearable Computers, 2001*, 2001, pp. 115–122.

[12] K. Van Laerhoven, A. Schmidt, and H.-W. Gellersen, “Multi-sensor context aware clothing,” in *Proc. of the 6th International Symposium on Wearable Computers*, 2002, pp. 49–56.

[13] K. Van Laerhoven and H.-W. Gellersen, “Spine versus porcupine: a study in distributed wearable activity recognition,” in *Proceedings of Eighth International Symposium on Wearable Computers (ISWC).*, 2004, pp. 142–149.

[14] T. Ho, J. Hull, and S. Srihari, “Decision combination in multiple classifier systems,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 1, pp. 66–75, 1994.

[15] T. Wang, Y. Han, P. Varshney, and P. Chen, “Distributed fault-tolerant classification in wireless sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 3, no. 4, pp. 724– 734, 2005.

[16] R. Sharma, “Toward multimodal human-computer interface,” *Proceedings of the IEEE*, vol. 86, no. 5, pp. 853–869, 1998.

[17] K. Ming Hsiao, G. West, S. Vedatesh, and K. M., “Online context recognition in multisensor system using dynamic time warping,” 5-8 Dec. 2005, pp. 283–288.

[18] R. Watson, “A survey of gesture recognition techniques,” Department of Computer Science, Trinity College, Dublin, Tech. Rep. TCD-CS-93-11, 1993.

[19] S. Mitra and T. Acharya, “Gesture recognition: A survey,” *IEEE Transactions on Systems, Man and Cybernetics - Part C*, vol. 37, no. 3, pp. 311–324, May 2007.

[20] T. Starner and A. Pentland, “Real-time american sign language recognition from video using hidden markov models,” in *International Symposium on Computer Vision*, 1995.

[21] J. Kittler and al., “On combining classifiers,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[22] I. Rish, “An empirical study of the naive bayes classifier,” in *International Joint Conference on Artificial Intelligence*, 2001, pp. 41–46.

[23] T. Stiefmeier, D. Roggen, and G. Tröster, “Fusion of string-matched templates for continuous activity recognition,” in *11th IEEE International Symposium on Wearable Computers*, October 2007.

[24] C. Lombriser, D. Roggen, M. Stäger, and G. Tröster, “Titan: A tiny task network for dynamically reconfigurable heterogeneous sensor networks,” in *15. Fachtagung Kommunikation in Verteilten Systemen (KiVS)*, 2007, pp. 127–138.