Recognizing Workshop Activity Using Body Worn Microphones and Accelerometers

P. Lukowicz, J. Ward, H. Junker, M. Stäger, G. Tröster
ETH - Swiss Federal Institute of Technology
Wearable Computing Laboratory
8092 Zürich, Switzerland
www.wearable.ethz.ch

A. Atrash and T. Starner
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332-0280
{amin,thad}@cc.gatech.edu

Abstract

Most gesture recognition systems analyze gestures intended for communication (e.g. sign language) or for command (e.g. navigation in a virtual world). We attempt instead to recognize gestures made in the course of performing everyday work activities. Specifically, we examine activities in a wood shop, both in isolation as well as in the context of a simulated assembly task. We apply linear discriminant analysis (LDA) and hidden Markov model (HMM) techniques to features derived from body-worn accelerometers and microphones. The resulting system can successfully segment and identify most shop activities with zero false positives and 83.5% accuracy.

1 Introduction

Advances in technology are allowing computer support for mobile applications. Delivery, maintenance, and manufacturing personnel are adopting mobile computing devices to support their work. Similarly, consumers now have access to mobile electronic tourist guides, communication devices, and health and wellness monitoring devices.

A key issue in most such mobile applications is the effort required to devote to operating the devices. Whereas in a desktop setting the computer is the focus of the user's attention, the user is forced to focus his attention on the environment for many mobile applications. Accessing the computer should require minimal cognitive and physical effort to prevent distracting the user from his primary task.

1.1 Context Sensitivity in Wearable Systems

In addressing the above issues wearable computers have recently emerged as a promising new paradigm. To reduce the physical effort required to operate the device they are designed to be a permanently accessible part of the user's outfit, have mostly hands free input devices, and head-up displays.

With respect to the cognitive load, many wearable systems focus on *context sensitivity* and proactiveness (e.g [1]). The system should be aware of the user's action and the activities occuring in his environment. Based on this awareness, the system can adapt its configuration, deliver information to the user, or record interesting events *without any* explicit user input [17]. For example, a maintenance support system could recognize what particular task is being performed by the user and automatically display the relevant manual pages on the system's head-up display. The wearable could also record the sequence of operations that are being performed for later analysis or could warn the user if an important step has been forgotten.

1.2 Recognition Approach

Past approaches by the authors have used head-mounted cameras and computer vision techniques to identify user context [17]. Although the visual signal contains much relevant information about any given situation, vision–based recognition has several disadvantages. For one, reliable localization and recognition of the relevant objects (hands, machine parts, tools) in complex scenes is an open research problem. In addition, computer vision techniques have difficulty with the unstructured, moving backgrounds and varying lighting condition as is common to many wearable scenarios, and relevant parts of the scene might be out of view or obstructed. Finally, video recognition is computationally intensive, often requiring resources not available on a wearable system.

A recognition approach gaining popularity in the wearable community is simple sensors integrated in the user's outfit and in the user's artifacts (e.g. tools, appliances, or parts of the machinery) [10]. One of the key aspects of this approach is the recognition and tracking of postures and gestures using motion sensors attached to appropriate locations on the user's limbs. Initial experiments have shown that many activities can be well identified through such analysis[8]. Another important source of information about

environmental activity is sound. It has been shown that in many situations ambient sound analysis can be used to distinguish between different settings, activities, and situations [4].

1.3 Paper Aims and Contributions

This paper is part of our work aiming to develop a reliable context recognition methodology based on the above approach. It presents a novel way of combining motion sensor-based gesture recognition with sound data from distributed microphones. In particular we exploit intensity differences between microphones on the wrist of the dominant hand and on the chest to identify relevant actions performed by the user's hand.

In the paper we focus on tracking user activity during assembly or maintenance tasks. Such tasks are among the most important applications of wearable computing (e.g. [2, 7]) and could significantly benefit from context sensitivity. At the same time these tasks are well structured and limited to a reasonable number of often repetitive actions. In addition, machines and tools typical to a workshop environment generate distinct sounds. Therefore, these activities are well suited for a combination of gesture and sound–based recognition.

This paper describes our approach and the results produced in an experiment performed on an assembly task in a wood workshop. We demonstrate that simple sensors placed on the user's body can reliably select and recognize user actions during a workshop procedure.

1.4 Related Work

Acceleration–based activity recognition has been studied by different research groups [11, 14, 19]. However all of the above work focused on recognizing comparatively simple activities (walking, running, and sitting). Sound based situation analysis has been investigated by Pelton *et al.* and in the wearables domain by Clarkson and Pentland [12, 5]. Intelligent hearing aids have also exploited sound analysis to improve their performance [3].

2 Experimental Setup

Performing initial experiments on live assembly or maintenance tasks is unadvisable due to the cost and safety concerns and the ability to obtain repeatable measurements under experimental conditions. As a consequence we have decided to focus on an "artificial" task performed at the workbench of wood workshop of our lab (see Figure 1). The task consisted of assembling a simple object made of two pieces of wood and a piece of metal. The task required 8 processing steps using different tools and including walking and
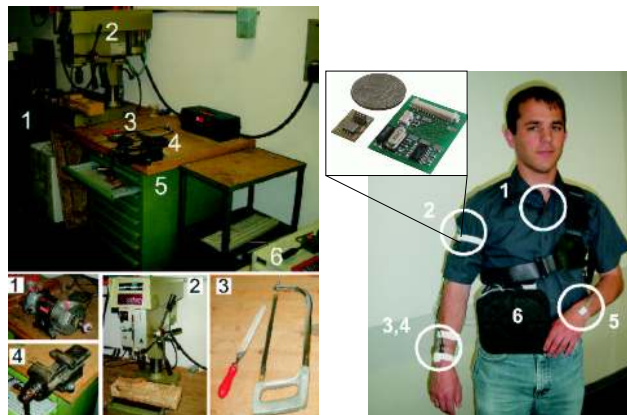

Figure 1: Left: the wood workshop with 1) grinder, 2) drill, 3)file and saw, 4) vice, and 5) cabinet with drawers. Right: The sensor type and placement is identical with the one used in our experiment: 1,4: microphone, 2,3 and 5: 3-axis acceleration sensors.

other gestures similar to an assembly task in a real world setting.

2.1 Procedure

No	action
1	take the wood out of the drawer
2	put the wood into the vice
3	take out the saw
4	saw
5	put the saw into the drawer
6	take the wood out of the vice
7	drill
8	get the nail and the hammer
9	hammer
10	put away the hammer, get the driver and the screw
11	drive the screw in
12	put away the driver
13	pick up the metal
14	grind
15	put away the metal, pick up the wood
16	put the wood into the vice
17	take the file out of the drawer
18	file
19	put away the file, take the sandpaper
20	sand
21	take the wood out of the vice

Table 1: Steps of workshop assembly task.

The assembly sequence consists of sawing a piece of wood, drilling a hole in it, grinding a piece of metal, attaching it to the piece of wood with a screw, hammering in a nail to connect the two pieces of wood, and then finishing the product by smoothing away rough edges with a file

and a piece of sandpaper. The wood was fixed in the vice for sawing, filing, and smoothing (and removed whenever necessary). The test subject moved between areas in the workshop between steps. Also, whenever a tool or an object (nail screw, wood) was required, it was retrieved from its drawer in the cabinet and returned after use.

The exact sequence of actions is listed in Table 1. The task was to recognize all tool-based activities. Tool-based activities excludes drawer manipulation, user locomotion, and clapping (a calibration gesture). The experiment was repeated 10 times in the same sequence to collect data for training and testing. For practical reasons, the individual processing steps were only executed long enough to obtain an adequate sample of the activity. This policy did not require the complete execution of any one task (e.g. the wood was not completely sawn), allowing us to complete the experiment in a reasonable amount of time. However this protocol influenced only the duration of each activity and not the manner in which it was performed.

2.2 Data Collection System

The data was collected using the ETH PadNET sensor network [8] equipped with 3 axis accelerometer nodes and two Sony mono microphones connected to a body worn computer. The position of the sensors on the body is shown in Figure 1: an accelerometer node on both wrist and on the upper arm of the right hand and a microphone on the chest and on the right wrist (the test subject was right handed).

As can be seen in Figure 1 each PadNET sensor node consist of two modules. The main module incorporates a MSP430149 low power 16-Bit mixed signal microprocessor (MPU) from Texas Instruments running at 6 MHz maximum clock speed. The current module version reads out up to three analog sensor signals including amplification and filtering and handles the communication between modules through dedicated I/O pins. The sensors themselves are hosted on an even smaller 'sensor-module' that can be either placed directly on the main module or connected through wires. In the experiment described in this paper sensor modules were based on a 3-axis accelerometer package consisting of two ADXL202E devices from Analog Devices. The analog signals from the sensor were lowpass filtered ($f_{cutoff} = 50Hz$) and digitized with 12Bit resolution using a sampling rate of 100Hz.

3 Recognition

3.1 Acceleration Data Analysis

Figure 2 shows a segment of the acceleration data collected during the experiment. The segment includes sawing, removing the wood from the vice, and drilling. The user ac-

cesses the drawer two times and walks between the vice and the drill. Clear differences can be seen in the acceleration signals. For example, sawing clearly reflects a periodic motion. By contrast, the drawer access (marked as 1a and 1b in the figure) shows a low frequency "bump" in acceleration. This bump corresponds to the 90 degree turns of the wrist as the user releases the drawer handle, retrieves the object, and grasps the handle again to close the drawer.

Given the data, time series recognition techniques such as hidden Markov models (HMMs) [13] should allow the recognition of the relevant gestures. However, a closer analysis reveals two potential problems. First, not all relevant activities are strictly constrained to a particular sequence of motions. While the characteristic motions associated with sawing or hammering are distinct, there is high variation in drawer manipulation and grinding. Secondly, the activities are separated by sequences of user motions unrelated to the task (e.g the user scratching his head). Such motions may be confused with the relevant activities. We define a "noise" class to handle these unrelated gestures.

3.2 Sound Data Analysis

Considering that most gestures relevant for the assembly/maintanance scenario are associated with a distinct sounds, sound analysis should help to address the problems described above. We distinguish between three different types of sounds:

1. *Sounds made by a handtool:* - Such sounds are directly correlated with user hand motion. Examples are sawing, hammering, filing, and sanding. These actions are generally repetitive, quasi–stationary sounds (i.e. relatively constant over time - such that each time slice on a sample would produce an identical spectrum over a reasonable length of time). In addition these sounds are much louder than the background noise (dominant) and are likely to be much louder at the microphone on the user's hand than on his chest. For example, the intensity curve for sanding (see Figure 2 top right) reflects the periodic sanding motion with the minima corresponding to the changes in direction and the maxima coinciding with the maximum sanding speed in the middle of the motion. Since the user's hand is directly on the source of the sound the intensity difference is large. For other activities it is smaller, however in most cases still detectable.

2. *Semi-autonomous sounds:* These sounds are initiated by user's hand, possibly (but not necessarily) remaining close to the source for most of the sound duration. This class includes sound produced by a machine, such as the drill or grinder. Although ideal quasi-stationary sounds, sounds in this class may not necessarily be
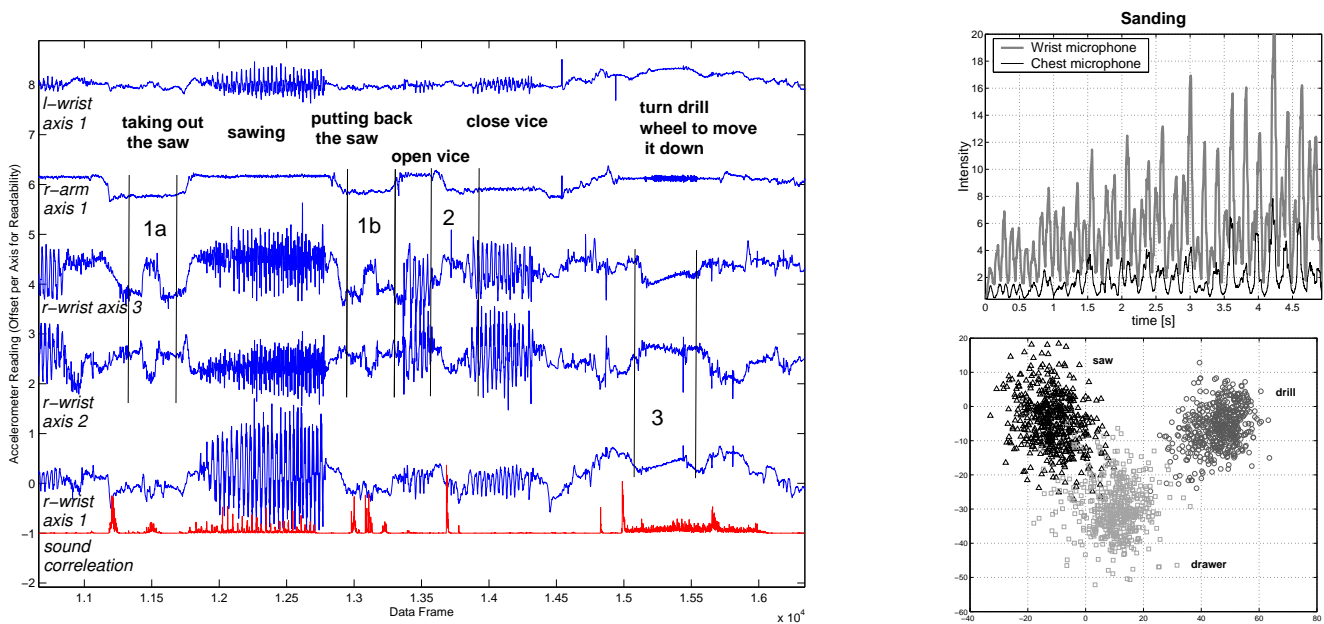
Figure 2: Left: example accelerometer data from sawing and drilling. Right top: audio profile of sanding from wrist and chest microphones. Right bottom: clustering of activities in LDA space

dominant and tend to have a less distinct intensity difference between the hand and the chest (for example, when a user moves their hand away from the machine during operation).

3. *Autonomous sounds:* These are sounds generated by activities not driven by the user's hands (e.g loud background noises or the user speaking).

Obviously the vast majority of relevant actions in assembly and maintanance are associated with hand tool sounds and semi–autonomous sounds. In principle, these sounds should be easy to identify using intensity differences between the wrist and the chest microphone. In addition, if extracted appropriately, these sounds may be treated as quasi-stationary and can be reliably classified using simple spectrum pattern matching techniques.

The main problem with this approach is that many irrelevant actions are also likely to fall within the definition of handtool and semi–autonomous sound. Such actions include scratching or putting down an object. Thus, like acceleration analysis, sound–based classification also has problem distinguishing relevant from irrelevant actions and will produce a number of false positives.

3.3 Recognition Methodology

Neither acceleration nor sound provide enough information for perfect extraction and classification of all relevant activities; however, we hypothesize that their sources of error are likely to be statistically distinct. Thus, we develop a technique based on the fusion of both methods. Our procedure consists of three steps:

1. Extraction of the relevant data segments using the intensity difference between the wrist and the chest microphone. We expect that this technique will segment the data stream into individual actions (including many actions we will model as noise).

2. Independent classification of the actions based on sound or acceleration. This step will yield imperfect recognition results by both the sound and acceleration subsystems.

3. Removal of false positives. While the sound and acceleration subsystems are each imperfect, when their classifications of a segment agree, the result may be more reliable (if the sources of error are statistically distinct).

4 Isolated Activity Recognition

As an initial experiment, we segment the activities in the data files by hand and test the accuracy of the sound and acceleration methods separately.

4.1 Sound Recognition

4.1.1 Method

The basic classification scheme operates on individual sound segments of length t_w. The approach follows a three step process: feature extraction, dimensionality reduction, and the actual classification.

The features used are the spectral components of each t_w obtained by Fast Fourier Transformation (FFT). This produces $N = \frac{f_s}{2} \cdot t_w$ dimensional feature vectors.

Rather than attempting to classify such large N-dimensional vectors directly, Linear Discriminant Analysis (LDA)[6] is employed to derive an optimal projection of the data into a smaller, M dimensional feature space (where M is the number of classes). In the "recognition phase", the LDA transformation is applied to the data segment under test to produce the corresponding $M - 1$ dimensional feature vector.

Using a labeled training-set, class means are calculated in the $M - 1$ dimensional space. Classification is performed simply by choosing the class mean which has the minimum Euclidean distance from the test feature vector (see Figure 3 bottom right).

4.1.2 Intensity Analysis

Making use of the fact that signal intensity is inversely proportional to the square of the distance from its source, the ratio of the two intensities I_{wrist}/I_{chest} is used as a measure of absolute distance of source from the user. Assuming the sound source is distance d from the wrist microphone and $d + \delta$ from the chest, the ratio of the intensities will be proportional to

$$\frac{I_{wrist}}{I_{chest}} \simeq \frac{(d+\delta)^2}{d^2} = \frac{d^2 + 2d\delta + \delta^2}{d^2} = 1 + \frac{2\delta}{d} + \frac{\delta^2}{d^2}$$

When both microphones are separated by at least δ, any sound produced a distance d (where $d \gg \delta$) from the user will bring this ratio close to one. Sounds produced near the chest microphone (e.g. the user speaking) will cause the ratio to approach zero whereas any sounds close to the wrist mic will make this ratio large.

Sound extraction is performed by sliding a window w_{ia} over the f_s Hz resampled audio data. On each iteration, the signal energy over w_{ia} for each channel is calculated. For these windows, the difference in ratio I_{wrist}/I_{chest} and its reciprocal are obtained, which are then compared to an empirically obtained threshold th_{ia}.

The difference $I_{wrist}/I_{chest} - I_{chest}/I_{wrist}$ provides a convenient metric for thresholding - zero indicates a far off (or exactly equidistant) sound while above or below zero values indicate a sound closer to the wrist or to the chest mic respectively.

Sound	LDA	IA+LDA	maj(IA+LDA)
Hammer	96.79	98.85	100
Saw	92.71	92.98	100
Filing	69.68	81.43	100
Drilling	99.59	99.35	100
Sanding	93.66	92.87	100
Grinding	97.77	97.75	100
Screwing	91.17	93.29	100
Vice	80.10	81.14	100
Overall	90.18	92.21	100

Table 2: Isolated Recognition Accuracy Per Sound (in %) for LDA alone, LDA with IA preselection and majority decision.

4.2 Results

In order to analysis the performance of the LDA on isolated classes, individual examples of each class were partitioned from each of the 10 experiments, providing 10 examples of each class. Eight examples of each class were used for training while testing on the remaining two examples.

Earlier work[15] cited f_s=5kHz and t_w=0.05 seconds (256 points) as optimal parameters for general purpose sound recognition tasks. In this task, it was found that recognition rates were improved using a larger t_w=0.1; at the same time f_s could be reduced to 2kHz without any notable adverse effects.

With these parameters, a sliding window t_w LDA classification was run directly over all the class partitioned samples. This process returned an overall recognition rate of 90.19%. The individual class results are given in the first column of Table 2. We next used intensity analysis to select only the samples over a given threshold to pass to the LDA procedure. This technique resulted in a slightly higher accuracy of 92.21% as shown in the second column of Table 2. The third column of Table 2 shows a variation of this technique where we slide a window over the data and classify the data at each window segment. A majority decision over the window segments was used to determine the overall label for a given isolated activity. This technique resulted in 100% recognition over the test data.
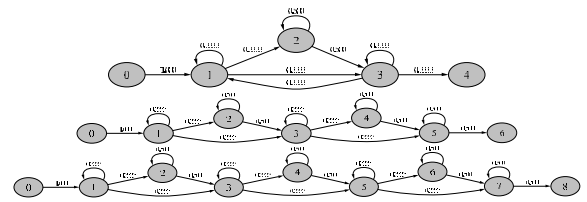

Figure 3: HMMs topologies.

4.3 Accelerometer–Based Activity Recognition

Hidden Markov models (HMMs) are probabilistic models used to represent non-deterministic processes in partially observable domains and are defined over a set of states, transitions, and observations. Details of HMMs and the respective alogrithms are beyond the scope of this paper but may be found in Rabiner's tutorial on the subject [13].

Hidden Markov models have been shown to be robust for representation and recognition of speech [9], handwriting [16], and gestures [18]. HMMs are capable of modelling important properties of gestures such as time variance (the same gesture can be repeated at varying speeds) and repetition (a gesture which contains a motion which can be repeated any number of times). They also handle noise due to sensors and imperfect training data by providing a probabilistic framework.

For gesture recognition, a model is trained for each of the gestures to be recognized. In our experiment, the set of gestures includes saw, drill, screw, hammer, sand, file, drawer, vice, and clap. Once the models are trained, a sequence of features can be passed to a recognizer which calculates the probability of each model given the observation sequence and returns the most likely gesture. For our experiments, the set of features consists of readings from the accelerometers positioned at the wrist and at the elbow. This provides 6 total feature values which are then normalized to sum to one and collected at approximatedly 93 Hz.

We found that most of the workshop activities typically require only simple HMMs for modelling. For file, sand, saw, and screw, a 5 state model with 1 skip transition and 1 loopback transtion suffice because they consist of simple repetitive motions. Drill is better represented using a 7 state model. Clapping, drawer, and grinding are slightly more complex and required 9 state models. The vice is unique in that it has two seperate motions, opening and closing. Thus a 9 state model is used with two appropriate loopbacks to correctly represent the gesture. These models were selected through inspection of the data, an understanding of nature of the activities, and experience with HMMs.

4.4 HMM Isolation Results

For this project, a prototype of the Georgia Tech Gesture Recognition Toolkit was used to train the HMMs and for recognition. The Toolkit is an interface to the HTK toolkit [20] designed for training HMMs for speech recognition. HTK handles the algorithms for training and recognizing the Hidden Markov Models allowing us to focus primarily on properly modelling the data.

To test the performance of the HMMs in isolation, the shop accelerometer data was partitioned by hand into 255 individual examples of gestures then used as a training set

for the HMMs. Accuracy of the system was calculated by performing leave-one-out validation by iteratively reserving one sample for testing and training on the remaining samples for each sample. The HMMs were able to correctly classify 93.33% of the gestures over data collected from the shop experiments. While this method has slightly more errors than the LDA method in isolation, we continue to pursue it for use in the continuous recognition case.

5 Continuous Recognition

Recognition of gestures from a continuous stream of features is difficult. However, we can simplify the problem by partitioning the continuous stream into segments and attacking the problem as isolated recognition. This approach requires a method of determining a proper partitioning of the continuous stream. We take advantage of the intensity analysis described in the previous section as a technique for identifying appropriate segments for recognition.

Since neither LDA nor the HMM are perfect at recognition, and each is able to recognize a different set of gestures well due to working in different feature space, it is advantagous to compare their independant classifications of a segment. If the classification of the segment by the HMMs matches the classification of the segment by the LDA, the classification can be believed. Otherwise, the noise class can be assumed, or perhaps a decision appropriate to the task can be taken (such as requesting additional information from the user).

Thus, the recognition is performed in three main stages: 1) Extracting potentially interesting partitions from the continuous saquence, 2) Classifying these individually using the LDA and HMMs, and 3) Combining the results from these approaches.

5.1 LDA for partitioning

For classification, partitioned data needs to be arranged in continuous sections corresponding to a single user activity. Such partitioning of the data is obtained in two steps: First, LDA classification is run on segments of data chosen by the IA. Those segments not chosen by intensity analysis are returned with classification zero. (In this experiment, classifications are returned at the same rate as accelerometer features); Secondly, these small window classifications are further processed by a larger (several seconds) majority decision window, which returns a single result for the entire window duration.

This partitioning mechanism helps reduce the complexity of continuous recognition. It will not give accurate bounds on the beginning and end of a gesture. Instead, the goal is to provide enough information to generate context at a general level; ie., "The user is hammering" as opposed to

Gesture	HMM					LDA					HMM + LDA					
	C	I	D	S	Acc	C	I	D	S	Acc	C	I	D	S	Acc	P(G|Class)
Hammer	9	3	0	0	66.7	9	1	0	0	88.9	9	0	0	0	100	1.00
Saw	9	0	0	0	100	9	1	0	0	88.9	9	0	0	0	100	1.00
Filing	9	0	0	1	100	9	7	0	1	23.2	8	0	2	0	80	1.00
Drilling	9	4	0	0	55.6	9	1	0	0	88.9	9	0	0	0	100	1.00
Sanding	8	0	0	1	88.9	9	8	0	0	11.1	8	0	1	0	88.9	1.00
Grinding	11	11	0	0	0	9	0	0	2	81.8	9	0	2	0	81.8	1.00
Screwing	4	0	0	5	44.4	9	75	0	0	-733.3	4	0	5	0	44.4	1.00
Vice	41	0	0	2	95.3	34	1	2	7	76.6	35	0	8	0	81.3	1.00
Overall	100	18	0	9	72.5	97	94	2	10	2.8	91	0	18	0	83.5	1.00

Table 3: Continuous recognition accuracy per gesture (Correct | Insertions | Deletions | Substitutions | Accuracy) and probability of gesture given classification P(G|Class)

"A hammering gesture occured between sample 1500 and 2300." The system is tolerant of, and does not require, perfect alignment between the partitions and the actual gesture. The example alignment shown in Figure 5 is acceptable for our purposes.

5.2 LDA Results

Analysis of the data was performed to test the system's ability to reconstruct the sequence of gestures in the shop experiments based on the partitioning and recognition techniques described to this point. Figure 4 shows an example of the automated partitioning versus the actual events. For this analysis of the system, the non-tool gestures, drawer and clapping, were considered as part of the noise class. After applying the parition scheme, a typical shop experiment resulted in 25-30 different partitions.

5.3 HMM Classification

Once the partitions are created by the LDA method, they are passed to set of HMMs for further classification. For this experiment, the HMMs are trained on individual gestures from the shop experiments using 6 accelerometer features from the wrist and elbow. Ideally, the HMMs will return a single gesture classification for each segment. However, the segment sometimes includes the beginning or end of the next or previous gesture respectively, causing the HMMs to return a sequence of gestures. In such cases, the gesture which makes up the majority of the segment is used as the classification. For example the segment labelled "B" in Figure 5 may return the sequence "hammer vice" and would then be assigned as the single gesture "vice."

5.4 Combining LDA and HMM classification

For each partitioned segment, the classification of the LDA and HMM methods were compared. If the classifications matched, that classification was assigned the segment. Otherwise, the noise class was returned.
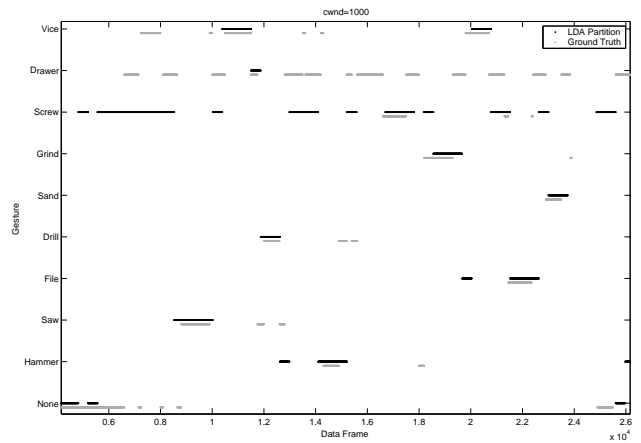

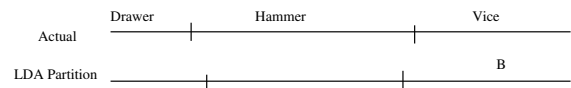Figure 4: LDA partitions versus ground truth

Figure 5: Detailed example of LDA partitioning

Table 3 shows the number of correct classifications (C), insertions (I), deletions (D), and substitutions(S) for the HMMs, the LDA, and the combination. Insertions are defined as noise gestures identified as a tool gesture. Deletions are tool gestures recognized as noise gestures. A substitution for a gesture occurs when that gesture is incorrectly identified as a different gesture. In addition, the accuracy of the system is calculated based on the following metric:

$$\% Accuracy = \frac{Correct - Insertions}{Total Samples}$$

The final column reports the probability of a gesture having occurred given that the system reported that gesture.

The accuracies for the HMMs and the LDA are calculated as well as for the combined method. Clearly, the HMMs and LDA each perform better than the other on various gestures and tended to err in favor of a particular gesture. When incorrect, LDA tended to report the "screw" gesture. Similarly, the HMMs tended to report "grinding" or "drilling." Comparing the classification of the LDA and the HMMs help eliminate the false positives and improve the performance of the system. The data shows that the comparison method performed better than the HMMs and the LDA in many cases and improved the accuracy of the system.

Although the accuracy of the system in general is not perfect, it is important to note that the combined HMM + LDA method results in no insertions or substitutions. This result implies that when the system returns a gesture, that gesture, in fact, did occur. While the system still misses some gestures, a user interface designer can know that the system did not return a false positive and be confident in his use of context. This attribute is especially important in the ubiquitous computing and wearable fields which are sensitive to such errors.

6   Conclusion

We have shown a system capable of segmenting and recognizing typical user gestures in a workshop environment. The system uses wrist and chest worn microphones and accelerometers and leverages the feature attributes of each modality to improve the system's performance. The system demonstrated perfect performance in isolated gesture testing and a zero false positive rate in the continuous case. In the future, we hope to apply these promising techniques for recognizing everyday gestures in more general scenarios.

References

[1] D. Abowd, A. K. Dey, R. Orr, and J. Brotherton. Context-awareness in wearable and ubiquitous computing. *Virtual Reality*, 3(3):200–211, 1998.

[2] L. Bass, D. Siewiorek, A. Smailagic, and J. Stivoric. On site wearable computer system. In *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, volume 2 of *Interactive Experience*, pages 83–84, 1995.

[3] M. C. Büchler. *Algorithms for Sound Classification in Hearing Instruments*. PhD thesis, ETH Zurich, 2002.

[4] B. Clarkson and A. Pentland. Extracting context from environmental audio. In *2nd Int'l Symposium on Wearable Computers*, pages 154–155, Oct. 1998.

[5] B. Clarkson, N. Sawhney, and A. Pentland. Auditory context awareness in wearable computing. In *Workshop on Perceptual User Interfaces*, Nov. 1998.

[6] R. Duda, P. Hart, and D. Stork. *Pattern Classification, Second Edition*. Wiley, 2001.

[7] J. H. Garrett and A. Smailagic. Wearable computers for field inspectors: Delivering data and knowledge-based support in the field. *Lecture Notes in Computer Science*, 1454:146–164, 1998.

[8] N. Kern, B. Schiele, H. Junker, P. Lukowicz, and G. Tröster. Wearable sensing to annotate meeting recordings. In *6th Int'l Symposium on Wearable Computers*, pages 186–193, Oct. 2002.

[9] F. Kubala, A. Anastasakos, J. Makhoul, L. Nguyen, R. Schwartz, and G. Zavaliagkos. Comparative experiments on large vocabulary speech recognition. In *ICASSP*, Adelaide, Australia, 1994.

[10] P. Lukowicz, H. Junker, M. Staeger, T. von Bueren, and G. Troester. WearNET: A distributed multi-sensor system for context aware wearables. In G. Borriello and L. Holmquist, editors, *UbiComp 2002: Proceedings of the 4th International Conference on Ubiquitous Computing*, pages 361–370. Springer: Lecture Notes in Computer Science, Sept. 2002.

[11] J. Mantyjarvi, J. Himberg, and T. Seppanen. Recognizing human motion with multiple acceleration sensors. In *2001 IEEE International Conference on Systems, Man and Cybernetics*, volume 3494, pages 747–752, 2001.

[12] V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi, and T. Sorsa. Computational auditory scene recognition. In *IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, volume 2, pages 1941–1944, May 2002.

[13] L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, pages 4–16, January 1986.

[14] C. Randell and H. Muller. Context awareness by analysing accelerometer data. In *Digest of Papers. Fourth International Symposium on Wearable Computers.*, pages 175–176, 2000.

[15] M. Stäger, P. Lukowicz, N. Perera, T. von Büren, G. Tröster, and T. Starner. Soundbutton: Design of a low power wearable audio classification system. Submitted to 7th Int'l Symposium on Wearable Computers, 2003.

[16] T. Starner, J. Makhoul, R. Schwartz, and G. Chou. Online cursive handwriting recognition using speech recognition methods. In *ICASSP*, pages 125–128, 1994.

[17] T. Starner, B. Schiele, and A. Pentland. Visual contextual awareness in wearable computing. In *IEEE Intl. Symp. on Wearable Computers*, pages 50–57, Pittsburgh, PA, 1998.

[18] T. Starner, J. Weaver, and A. Pentland. Real-time American Sign Language recognition using desk and wearable computer-based video. *IEEE Trans. Patt. Analy. and Mach. Intell.*, 20(12), December 1998.

[19] K. Van-Laerhoven and O. Cakmakci. What shall we teach our pants? In *Digest of Papers. Fourth International Symposium on Wearable Computers.*, pages 77–83, 2000.

[20] S. Young. *HTK: Hidden Markov Model Toolkit V1.5*. Cambridge Univ. Eng. Dept. Speech Group and Entropic Research Lab. Inc., Washington DC, 1993.