A Smart Watch-based Gesture Recognition System for Assisting People with Visual Impairments

Lorenzo Porzi
Fondazione Bruno Kessler
Trento, Italy
porzi@fbk.eu

University of Perugia
Perugia, Italy

Stefano Messelodi
Fondazione Bruno Kessler
Trento, Italy
messelod@fbk.eu

Carla Maria Modena
Fondazione Bruno Kessler
Trento, Italy
modena@fbk.eu

Elisa Ricci
University of Perugia
Perugia, Italy
elisa.ricci@diei.unipg.it

ABSTRACT

Modern mobile devices provide several functionalities and new ones are being added at a breakneck pace. Unfortunately browsing the menu and accessing the functions of a mobile phone is not a trivial task for visual impaired users. Low vision people typically rely on screen readers and voice commands. However, depending on the situations, screen readers are not ideal because blind people may need their hearing for safety, and automatic recognition of voice commands is challenging in noisy environments. Novel smart watches technologies provides an interesting opportunity to design new forms of user interaction with mobile phones. We present our first works towards the realization of a system, based on the combination of a mobile phone and a smart watch for gesture control, for assisting low vision people during daily life activities. More specifically we propose a novel approach for gesture recognition which is based on global alignment kernels and is shown to be effective in the challenging scenario of user independent recognition. This method is used to build a gesture-based user interaction module and is embedded into a system targeted to visually impaired which will also integrate several other modules. We present two of them: one for identifying wet floor signs, the other for automatic recognition of predefined logos.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Input devices and strategies; I.5.2 [**Design Methodology**]: Classifier design and evaluation

General Terms

User interfaces; mobile computing; algorithms

Keywords

Smart watch; accelerometer-based gesture recognition; dynamic time warping; visual impairments

IMMPD'13, October 22, 2013, Barcelona, Spain.

ACM 978-1-4503-2399-4/13/10 ...$15.00.
http://dx.doi.org/10.1145/2505483.2505487.

1. INTRODUCTION

Recent technological advances can provide interesting opportunities for improving the quality of life of disabled people. In particular, the widespread use of mobile devices has lead to the development of novel solutions for supporting disabled users when they move and carry out daily life activities (such as shopping or walking in the road). Typically, the input to a mobile device is provided with a finger or a stylus which is not convenient for visually impaired people. In order to make mobile solutions more accessible and usable, innovative techniques must be devised. In particular, in the specific situation of low vision people, interaction technologies based on voice have been developed. However, interaction through speech commands may not be useful in some situations (*e.g.* during meetings) or may not work properly in crowded scenarios or noisy environments.

The very recent advent of smart watches technologies provides an interesting alternative to audio-based user interaction systems in mobile phones for blind and low vision users. Modern smart watches are cheap and nonintrusive devices. Previous works [5, 17] have shown that, equipped with accelerometers and a vibration feedback, they can be successfully used for building a simple user interaction system based on gesture recognition. However, to our knowledge no previous works have considered the adoption of smart watches and accelerometers-based gesture recognition for devising novel interaction solutions for visually impaired.

In this paper we present the results of our first works towards the realization of a novel low cost system targeted to visually impaired users, which is based on the combination of a mobile phone and a smart watch. The signals of the smartwatch's integrated accelerometers are used as input to a robust user-independent gesture recognition algorithm. This algorithm runs on the mobile phone and relies on a novel approach based on fast Global Alignment Kernel [6] and on a SVM classifier. We also present two proof-of-concept modules which will be integrated in the final system. One module aims to automatically recognize wet floor signs, providing an alert to the user in the form of vibration feedback. The other module has been built to automatically identify a specific logo or symbol and is ultimately aimed to be used in various scenarios, *e.g.* to assist the user in recognizing products during shopping. Importantly, both the gesture recognition and the visual analysis modules are designed to

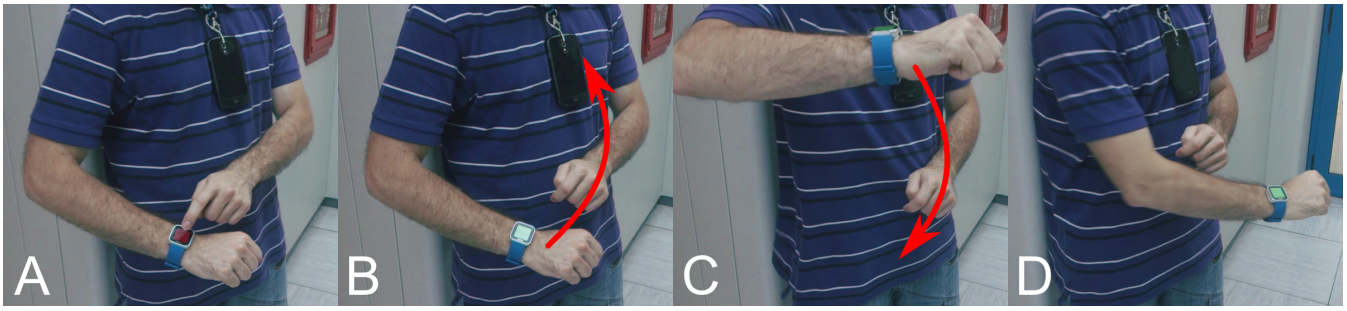

Figure 1: A user wearing a smart watch at her wrist and a smartphone at her neck. A: the user presses the smart watch's screen to start the gesture input. B,C: the user performs a gesture. D: the system recognizes the gesture and activates the corresponding function.

keep the computational cost limited in order to be able to run directly on the smartphone.

2. RELATED WORKS

Blind and in general low vision people have limited access to the world of mobile devices, mainly due to the reliance on visual displays and to the lack of audio or tactile feedback. To overcome these limitations, in the past several studies have been conducted to create more usable and accessible portable devices for the visually impaired [4, 11, 13]. However, at the present, we are not aware of a system which makes use of a smart watch. A notable exception is represented by the recent Freevox [2]. This system has been designed with accessibility for the visually impaired as a primary requirement, *i.e.* such that a blind or partially sighted user can activate the required functions on a mobile device very easily and intuitively. In a nutshell, it is a "talking" watch with a simplified touch screen interface (black and white, only four buttons in the corners of the screen). However it does not use gesture control, which we believe can be a natural way for the user to access the functionalities of the mobile phone, in alternative or together with current solutions based on the touch screen or voice commands.

Automatic gesture recognition has been studied for many years. The vast majority of current systems are based on computer vision techniques [14]. Unfortunately, the performance of vision-based approaches with traditional cameras are strongly influenced by lighting conditions and point of view. More recently, the advent of low cost depth cameras has led to the new generation of accurate visual-based gesture recognition solutions. However, when looking at user interaction systems for people freely moving around in arbitrary environments, a better choice seems to be offered by gesture recognition methods from accelerometer data. In the last few years several works have been proposed to address this problem [3, 8, 10, 12, 16, 19, 20].

Current gesture recognition methods based on accelerometers consider two possible scenarios: user-dependent and user-independent recognition. Most of previous works focuses mainly on user-dependent gesture recognition. This implies that a user is required to perform few gestures before using the system. These gesture will be used as template samples for subsequent recognition. An example of a user-dependent system is the popular uWave [12]. However, these solutions have limited applicability. On the other hand, the user-independent gesture recognition problem is more difficult as many variations of the same gesture can occur for different users. User-independent interaction methods do not require any preliminary phase of recording users' gesture before using the system and are typically based on a classifier which has been trained in advance. Some of these methods are based on Hidden Markov Models (HMMs) [8, 16]. However, the applicability of generative models like HMMs is limited by their computational complexity which is directly proportional to the number and the dimension of the feature vectors. Other approaches are based on the use of classifiers such as Bayesian Networks [19] or SVM [10, 20]. In particular the SVM-based methods have proven to be quite successful both for user-dependent and user-independent gesture recognition. In [10] the Haar transform is adopted in the feature extraction phase and is shown to produce effective descriptors for modeling accelerometers data. In [20] spectral and temporal features are combined and provided as input to the SVM classifier.

In this paper we adopt a different descriptor which has been shown to be particularly effective for other applications involving time series data [7, 18]: the fast GAK proposed by Cuturi [6]. The GAK has been introduced in order to adopt descriptors based on DTW within the framework of kernel methods. Approaches using DTW have shown to be very effective for gesture recognition from accelerometer data [3, 12]. However, the traditional DTW distance cannot be used rigorously with SVMs or other kernel approaches, as it does not satisfy the triangle inequality and does not define a positive definite kernel. Differently GAK is positive definite, and thus a valid kernel. Furthermore, the version of GAK presented in [6] provides some advantages in terms of computational complexity since its quadratic computational cost can be reduced, at the expenses of an approximate calculation, by tuning a single parameter. Our experimental results demonstrates the effectiveness of GAK for gesture recognition.

3. SYSTEM OVERVIEW

The proposed system is made up of two commercial subsystems, a wrist-worn smart watch, and a smartphone which the user holds in front of her breast using a necklace. Figure 1 shows a user whith the proposed system. The smart watch is used as an input device: first the user awakens the system by tapping the watch's screen. Then the user selects one of the available functions by performing a gesture with the arm, which is captured by the smart watch's accelerometer. The smartphone, on the other side, acts as the brain and eye of the system: it recognizes the gestures recorded

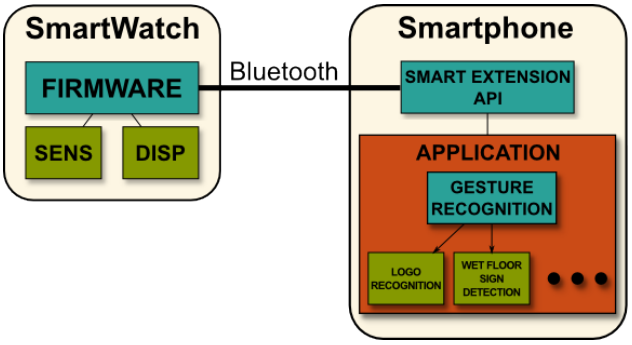

Figure 2: Smart watch to smartphone communication and software architecture.

by the smart watch and performs the relative vision-based task using its camera as input. When the task is completed the outcome is communicated to the user through vibration feedback. The devices used are a Sony Xperia Z smartphone and a Sony SmartWatch.

A simplified description of the hardware and software architecture of the system can be seen in Figure 2. The smart watch can not be directly programmed. Instead, its embedded firmware manages the device's hardware and communicates with the smartphone through Bluetooth radio. On the smartphone, a software abstraction layer (Sony Smart Extension APIs) acts as a middleware, providing a set of APIs for drawing on the smart watch's screen, reading its sensors etc. Due to this peculiar hardware-software configuration all the processing must be performed on the smartphone.

The smart watch is equipped with a 3-axis MEMS accelerometer, which provides linear acceleration measurements along three orthogonal axes. While the hardware sensor is theoretically capable of sampling rates of up to 5kHz [1], the current APIs limit the available data rate to around 10 samples per second. This sampling rate is quite low if compared to that available on more expensive smart watches, however our results show that it does not prevent the possibility to effectively use accelerometer data to recognize arm gestures.

4. GESTURE RECOGNITION SYSTEM

Using linear acceleration data as input, the problem of gesture recognition can be formulated as a sequence classification problem: a gesture is described as a time series of acceleration measurements $\mathbf{a} = (a_1, ..., a_{n_i})$, where $a_j = [a_{x,j}, a_{y,j}, a_{z,j}]^T$ is expressed as a vector of its x, y and z components. Notice that n_i, the number of samples, is not fixed and can vary greatly depending on gesture type and specific user.

4.1 Global Alignment Kernel

The DTW distance between two sequences $\mathbf{x} = (x_1, ..., x_n)$ and $\mathbf{y} = (y_1, ..., y_m)$ can be written as

$$\mathrm{DTW}(\mathbf{x}, \mathbf{y}) = \min_{\pi \in \mathcal{A}(n,m)} D_{\mathbf{x},\mathbf{y}}(\pi), \qquad (1)$$

where $D_{\mathbf{x},\mathbf{y}}(\pi)$ is a cost defined as:

$$D_{\mathbf{x},\mathbf{y}}(\pi) = \sum_{i=1}^{|\pi|} \varphi\left(x_{\pi_1(i)}, y_{\pi_2(i)}\right),$$

In Equation 1, $\mathcal{A}(n, m)$ stands for the set of all possible alignments π between the two sequences. The function $\varphi(x, y)$ measures the discrepancy between two points in the sequences $\mathbf{x}$ and $\mathbf{y}$, and in our case can be defined as the euclidean distance between two acceleration vectors.

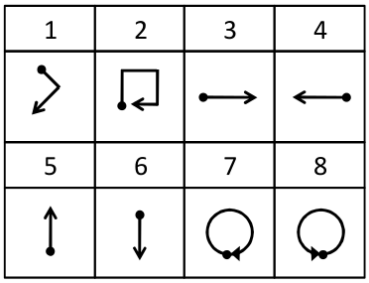

Figure 3: The set of gestures used for performance evaluation.

From this definition the global alignment kernel $k_{\mathrm{GA}}(\mathbf{x}, \mathbf{y})$ of [6] can be derived as an exponentiated soft minimum

$$k_{\mathrm{GA}}(\mathbf{x}, \mathbf{y}) = \sum_{\pi \in \mathcal{A}(n,m)} \prod_{i=1}^{|\pi|} \kappa(x_{\pi_1(i)}, y_{\pi_2(i)}), \qquad (2)$$

where $\kappa(x, y)$ is a local similarity function defined as $\kappa = e^{-\varphi(\cdot)}$. This kernel function, similar to the DTW distance, is quite expensive to compute, its complexity scaling with $O(nm)$. For this reason, in [6] a faster alternative to GAK is proposed, $k_{\mathrm{fGA}}(\mathbf{x}, \mathbf{y})$, obtained by replacing the local similarity $\kappa(\cdot)$ in Equation 2 with the function:

$$\tau(i, x; j, y) = \frac{\omega(i, j)\kappa_\sigma(x, y)}{2 - \omega(i, j)\kappa_\sigma(x, y)},$$

where:

$$\omega(i, j) = \left(1 - \frac{|i - j|}{T}\right)_+.$$

and $\kappa_\sigma(x, y)$ is the well-known Gaussian kernel. Given that $\omega(i, j)$ is zero when $i - j \geq T$, the calculation of $k_{\mathrm{fGA}}(\mathbf{x}, \mathbf{y})$ can be simplified, leading to a computational complexity that is $O(T \min(n, m))$.

An interesting property of this fast GAK is that the parameter T can be used to regulate a trade-off between accuracy and computational complexity. In fact, if $T = \max(n, m)$ the calculation of k_{fGA} requires $O(nm)$ operations as for k_{GA}, but the accuracy in general increases, as all possible alignments are taken into account when comparing sequences.

4.2 Experimental Results

To evaluate the performance of our recognition algorithm we use the set of gestures originally proposed in [9], shown in Figure 3, which is quite popular in the accelerometer-based gesture recognition literature. Currently our system only uses gestures three and four to activate the two proposed vision-based modules. The other gestures will be used as more applications become available. We acquired data from 15 users, each one repeating the eight gestures 15 times, for a total of 1800 sequences. For comparison we also implemented the Haar-SVM classifier in [10]: this classifier applies an Haar wavelet transform to the input sequences and uses the first 8 octaves as features to train a SVM with a radial basis function kernel. As a baseline we also provide

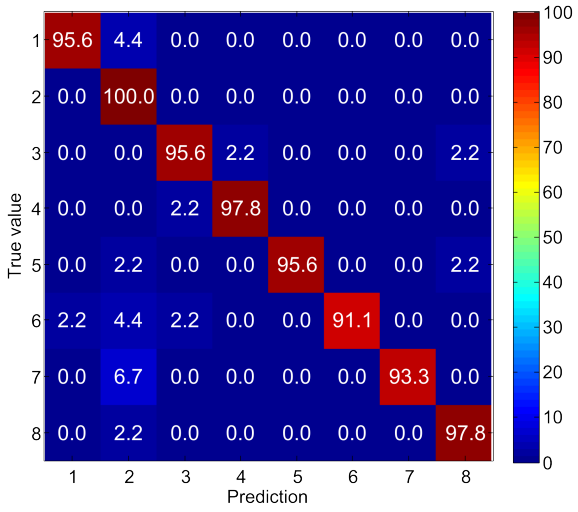

Figure 4: Confusion matrix for the `fGAK-SVM` classifier. Average accuracy: 95.8%.

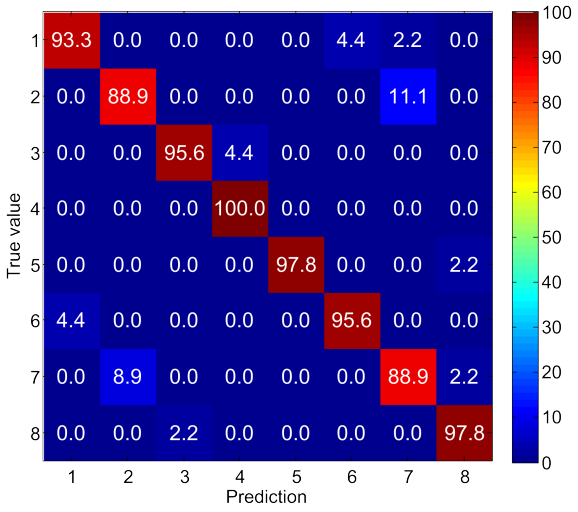

Figure 5: Confusion matrix for the `HAAR-SVM` classifier. Average accuracy: 94.7%.

the results obtained by a simple proximity search scheme in which the test sequences are classified by comparing them with DTW to every training sequence and choosing the gesture class with the lower average distance. In the following we will refer to our algorithm as `fGAK-SVM`, to the second algorithm as `HAAR-SVM` and to the last one as `DTW`.

In a preliminary step, we find experimentally optimal parameters for our algorithm through grid search. A five-fold cross validation is performed on the whole training set. As suggested in [6], instead of directly assigning a value to T and σ, we find two factors $\tilde{T}$ and $\tilde{\sigma}$, which are then scaled using the median of the lengths of the sequences and the median of the distances between randomly sampled acceleration vectors, to obtain the actual T and σ. The resulting values are: $\tilde{T} = 0.5, \tilde{\sigma} = 0.6, C = 2$. For `HAAR-SVM` the parameters obtained with cross validation are: $\gamma = 0.5, C = 8$.

Figure 4 and Fig. 5 show the confusion matrices obtained by running `fGAK-SVM` and `HAAR-SVM` on the whole dataset. The dataset is randomly split so that 70% of each user's sequences fall into the training and 30% in the test set. The overall performances of the two algorithms in this case are similar, with a slight advantage in favour of `fGAK-SVM`. It is worth noting that both classifiers show a tendency to confuse between class 2 and class 7, which can be intuitively justified by the fact that the two gestures are essentially the same clockwise motion with different shapes. The same experimental setup is used to evaluate the performance varying the parameter $\tilde{T}$. As shown in Fig. 7, increasing the value of $\tilde{T}$ the classification accuracy also increases, reaching a plateau at $\tilde{T} = 0.5$ (the chosen value).

As a last comparison, we show the accuracies obtained by testing the three algorithms in a leave-one-user-out scheme: the sequences recorded by an user are classified using the remaining as training data. The results are shown in Fig. 6. As can be seen, `fGAK-SVM` reaches higher accuracy than `HAAR-SVM`, while `DTW` performs much worse than the other two. The average accuracies are respectively 92.33%, 91.08% and 54.80%. It is evident that `DTW` is not suitable for user-independent learning. In this graph a comparison between the standard Global Alignment Kernel, denoted as `GAK-SVM`,

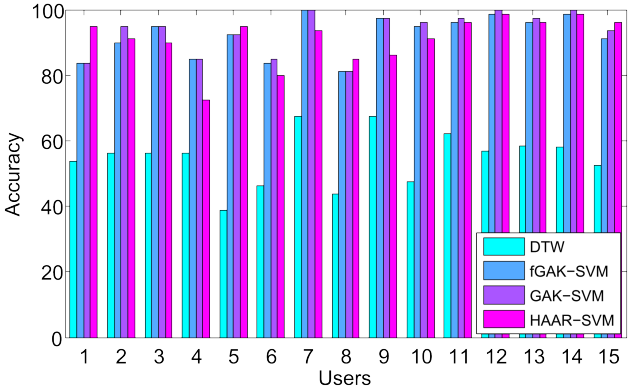

Figure 6: Accuracy of `fGAK-SVM`, `GAK-SVM`, `HAAR-SVM` and `DTW` in a leave-one-user-out evaluation scheme.

and `fGAK-SVM` is also provided. As expected slightly higher accuracy (93.33%) is obtained at the expenses of a higher computational cost.

To conclude, we report that our implementation of `fGAK-SVM`, running on a Sony Xperia Z smartphone, achieves an average processing time of $125ms$ to recognize a single gesture, making it suitable for the use in the proposed application.

5. VISION-BASED MODULES

The range of applications for visually impaired people based on the analysis of the images captured by a smartphone's camera is potentially unlimited.

The development of such applications is mainly bounded by two constraints: the limited processing power available and the low quality of the acquired images, which usually suffer from heavy motion blur. In the visually impaired assistance scenario, a further limitation is given by the difficulty or impossibility for the user to exploit visual feedback to improve image quality or aim the camera at the object of interest.

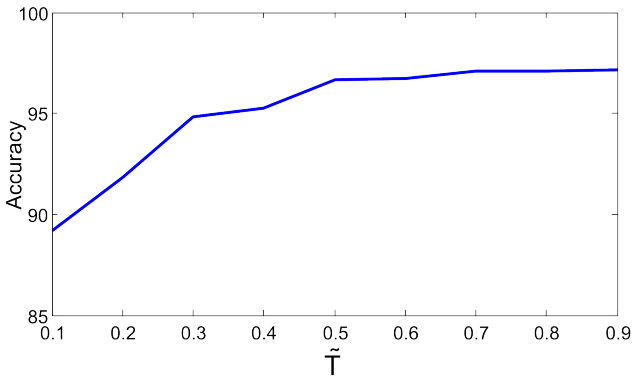

Figure 7: Accuracy of `fGAK-SVM` as the parameter $\tilde{T}$ varies.

In the following sections we describe two proof-of-concept modules that illustrate how, under certain conditions, it is indeed possible to develop potentially useful smartphone-based assisted living applications, even taking into account the previously described issues.

5.1 Wet Floor Sign Recognition

The first module implements an automatic recognizer of the classical yellow "wet-floor" signs which are usually placed in rooms, corridors or halls after cleaning the floor (Fig.8). The module represents an example of an application for the detection and signaling of potentially critical situations. As a matter of fact the warning sign itself, if not detected, constitutes an additional danger for the blind.

We start from the consideration that this kind of signs are characterized by an icon painted inside a black triangle on a yellow board.

In the first step of the algorithm the input image is converted into the HSL color-space and searched for the detection of relevant yellow regions: pixels having a certain saturation are classified as yellow depending on their hue. The convex hull of the yellow components are then analyzed: pixels with contrast with respect to the yellow background are extracted. This method is robust to changing illumination conditions and to image blurring, allowing to detect the black region of the external triangle and the internal icon.

A filtering and aggregation phases are then applied to the contrasted pixels and, finally, the regions that pass a "triangularity" test represent the output of the module.

The module has been tested in indoor environments under a wide range of illumination conditions. The average processing time per frame on a Sony Xperia Z smartphone is about 250 milliseconds, corresponding to about 4 fps.

5.2 Logo Recognition

The second module implements an automatic localizer of a predefined visual pattern, specifically the "LEGO" logo. A logo is a synthetic image designed to be clearly visible and immediately recognizable in order to strongly characterize a brand or a product.

Color, shape, graphical elements are the principal characteristics of the logo design, so that people often recognize a logo without reading the logotype as text.

We use the logo design to guide the search through the image. For example, the "LEGO" logo is characterized by the presence of two main colors, red and white, and in minor part yellow and black. The red region depicts a square box with white text inside it.

Figure 8: Use case: wet floor sign detection. 1) The user walks in a corridor. 2) The user comes across a wet floor sign. 3) The system detects the sign and notifies the user through smart watch vibration.

The first step of the detection algorithm consists of the partitioning of the image into $N+1$ categories, corresponding to the N main colors which describe the logo of interest, and a category including all the other colors. In the LEGO example we segment the images into red, white, and all-the-rest, and compute the red and white connected components.

The search stops if no white or red components are detected, otherwise the image is analysed, column by column, in order to find triples of red-white-red runs with comparable lengths.

The set of the red and white connected components is then clustered: two components are merged if they appear in the same triple red-white-red. Finally a check is applied to each cluster in order to retain only squared shaped ones.

This method is able to detect multiple instances of the searched logo, is scale independent, and robust to small rotations. An example of the application of the logo recognition system is shown in Fig.9. The average processing time per frame on the same Sony device is about 145 milliseconds corresponding to about 7 fps.

6. CONCLUSIONS

We presented a first prototype of a system[1] for the assistance of visually impaired people based on the use of a smartphone and a smart watch. The measurements from the accelerometers integrated in the smart watch are used as input to a gesture recognition module, which allows the user to select one of several vision-based functions implemented in the application. The proposed system provides several advantages over current user interaction technologies for low vision people. First of all, its use is not hampered by noisy environments, in contrast to solutions based on screen readers and voice commands. Moreover, smart watches can be worn without any prejudice, as they look similar to normal watches. Finally, the proposed system is quite cheap.

Future works will involve extending the set of possible gestures in order to increase the number of user commands, performing a user study to find a set of more meaningful gestures and investigating approaches to approximate the

[1]Video demonstrating the functionalities of the proposed system can be found at: `www.lorenzoporzi.com/gesture-recognition/`

Figure 9: Use case: logo recognition. The user moves the arm and activate the logo recognition functionality. As the user get closer to the LEGO boxes, the multiple logos are detected.

decision function of the classifier to further speed up the recognition process [15]. Moreover we will focus on the integration of the proposed solution with a voice-based interaction technology, the development of novel vision-based modules (*e.g.* an automatic currency detector, a module which helps the user in finding buttons in the elevator) and ultimately a validation phase of the overall system recruiting visually impaired people.

7. ACKNOWLEDGMENTS

This research has been partly funded by the European 7th Framework Program, under grant VENTURI (FP7-288238).

8. REFERENCES

[1] *http://developer.sonymobile.com/services/open-smartwatch-project/smartwatch-hacker-guide/.*

[2] *http://myfreevox.com/en/.*

[3] A. Akl and S. Valaee. Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing. In *ICASSP*, pages 2270–2273, 2010.

[4] R. Amar, S. Dow, R. Gordon, M. R. Hamid, and C. Sellers. Mobile advice: an accessible device for visually impaired capability enhancement. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '03, pages 918–919, New York, NY, USA, 2003. ACM.

[5] G. Bieber, T. Kirste, and B. Urban. Ambient interaction by smart watches. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '12, pages 39:1–39:6, New York, NY, USA, 2012. ACM.

[6] M. Cuturi. Fast global alignment kernels. In *ICML*, pages 929–936, 2011.

[7] C. Joder, S. Essid, and G. Richard. Temporal integration for audio classification with application to musical instrument classification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(1):174–186, 2009.

[8] H. Junker, O. Amft, P. Lukowicz, and G. Tröster. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recognition*, 41(6):2010–2024, 2008.

[9] J. Kela, P. Korpipää, J. Mäntyjärvi, S. Kallio, G. Savino, L. Jozzo, and D. Marca. Accelerometer-based gesture control for a design environment. *Personal and Ubiquitous Computing*, 10(5):285–299, 2006.

[10] M. Khan, S. Ahamed, M. Rahman, and J.-J. Yang. Gesthaar: An accelerometer-based gesture recognition method and its application in nui driven pervasive healthcare. In *Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on*, pages 163–166, 2012.

[11] F. C. Y. Li, D. Dearman, and K. N. Truong. Leveraging proprioception to make mobile phones more accessible to users with visual impairments. In *ASSETS*, pages 187–194, 2010.

[12] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan. uwave: Accelerometer-based personalized gesture recognition and its applications. *Pervasive and Mobile Computing*, 5(6):657–675, 2009.

[13] R. Manduchi and J. M. Coughlan. (computer) vision without sight. *Commun. ACM*, 55(1):96–104, 2012.

[14] S. Mitra and T. Acharya. Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(3):311–324, 2007.

[15] R. Perfetti, E. Ricci. Reduced complexity RBF classifiers with support vector centres and dynamic decay adjustment. *Neurocomputing*, 69(16-18):2446-2450, 2006.

[16] T. Pylvänäinen. Accelerometer based gesture recognition using continuous hmms. In *IbPRIA (1)*, pages 639–646, 2005.

[17] G. Raffa, J. Lee, L. Nachman, and J. Song. Don't slow me down: Bringing energy efficiency to continuous gesture recognition. In *ISWC*, pages 1–8, 2010.

[18] E. Ricci, F. Tobia, and G. Zen. Learning pedestrian trajectories with kernels. In *ICPR*, pages 149–152, 2010.

[19] Sung-Jung. Two-stage recognition of raw acceleration signals for 3-D Gesture-Understanding cell phones. In *Tenth International Workshop on Frontiers in Handwriting Recognition*, 2006.

[20] J. Wu, G. Pan, D. Zhang, G. Qi, and S. Li. Gesture recognition with a 3-d accelerometer. In *UIC*, pages 25–38, 2009.