ACCELEROMETER-BASED GESTURE RECOGNITION VIA DYNAMIC–TIME WARPING, AFFINITY PROPAGATION, & COMPRESSIVE SENSING

Ahmad Akl, Shahrokh Valaee

Department of Electrical and Computer Engineering, University of Toronto
Email: {ahmadakl, valaee}@comm.utoronto.ca

ABSTRACT

We propose a gesture recognition system based primarily on a single 3-axis accelerometer. The system employs dynamic time warping and affinity propagation algorithms for training and utilizes the sparse nature of the gesture sequence by implementing compressive sensing for gesture recognition. A dictionary of 18 gestures is defined and a database of over 3,700 repetitions is created from 7 users. Our dictionary of gestures is the largest in published studies related to acceleration-based gesture recognition, to the best of our knowledge. The proposed system achieves almost perfect user-dependent recognition and a user-independent recognition accuracy that is competitive with the statistical methods that require significantly a large number of training samples and with the other accelerometer-based gesture recognition systems available in literature.

Index Terms— *gesture recognition, dynamic time warping, affinity propagation, compressive sensing*

1. INTRODUCTION

Gesture recognition has become one of the hottest fields of research since it serves as an intelligent and a natural interface between the human and the computer. The proliferation in technology and in microelectronics more specifically, has inspired research in the field of accelerometer-based gesture recognition. 3-axis accelerometers are increasingly being incorporated and embedded into many personal electronic devices like the Apple iphone, Apple iPod touch, Wiimote, and Lenovo laptops, to name a few [1][2].

The majority of the available literature on gesture or action recognition combines data from a 3-axis accelerometer with data from another sensing device like a biaxial gyroscope [3] or EMG sensors [4] in order to improve the system's performance and to increase the recognition accuracy. Accelerometer-based gesture recognition system using continuous Hidden Markov Models (HMMs) [5] has been developed. However, the computational complexity of statistical or generative models like HMMs is directly proportional to the number as well as the dimension of the feature vectors [5]. Therefore, one of the major challenges with HMMs is estimating the optimum number of states and thus determining the probability functions associated with the HMM. Besides, variations in gestures are not necessarily Gaussian and perhaps, other formulations may turn out to be of better fit.

The most recent gesture recognition system that is accelerometer-based is the uWave [6]. uWave is a user-dependent system that supports personalized gesture recognition. uWave functions by utilizing only one training sample, stored in a template, for each gesture pattern. The core of the uWave is dynamic time warping (DTW) and the system's database undergoes two types of adaptation: positive and negative adaptation. However, uWave's database adaptation resembles continuous training and in some cases, if thorough examination of templates is ignored, removing an older template every other day might lead to replacing a very good representative of a gesture sequence, which is best avoided. Although uWave proves to perform incredibly efficient in terms of computational cost as well as recognition accuracy when compared to other systems and other statistical methods, being user-dependent limits the applications of uWave. Besides, researchers on accelerometer-based gesture recognition are envisaging a universal system that, given a dictionary of gestures, can recognize the different gestures with a competitive accuracy with minimal dependence on the user.

In this work, we propose an accelerometer-based gesture recognition system that uses a single 3-axis accelerometer to recognize gestures, where gestures here are hand movements. A dictionary of 18 gestures is created for which a database of 3,780 repetitions is built by collecting data from 7 participants. Some of the gestures defined in the dictionary are taken from the gesture vocabulary identified by Nokia [7]. Two tests are run: user-dependent and user-independent recognition. The core of recognizer's training phase is a consolidation of Dynamic Time Warping (DTW) and Affinity Propagation (AP) for both types of tests. For user-dependent, recognition involves comparing the unknown gesture repetition by DTW to the set of exemplars obtained during the training phase. On the other hand, for user-independent recognition, simple comparison by DTW doesn't suffice and here is where Compressive Sampling (CS) comes into picture. The system achieves an accuracy of 99.79% for user-dependent recognition for a dictionary size of 18 gestures. As for user-independent recognition, the system achieves an accuracy of 96.89% for a dictionary size of 8 gestures which is very competitive with other statistical models or other available approaches.

The rest of the paper is organized as follows: Section II presents the technical details of the proposed gesture recognition system. Section III describes an implementation of the gesture recognition system using a Wiimote. Section IV explains the simulations and discusses the results. Finally, section V concludes the paper.

2. GESTURE RECOGNITION SYSTEM

This section discusses the technical components of the proposed gesture recognition system: Temporal Compression, DTW, AP, and CS.

978-1-4244-4296-6/10/$25.00 ©2010 IEEE

Hand gestures are well-known to suffer from inherent temporal variations. They differ from person to person and even the same person cannot perfectly replicate the same gesture. This entails that gesture sequences can be either compressed or stretched depending on the user and the speed of the hand movement. In other words, the recorded gesture sequences are most of the time, if not always, of different lengths.

2.1. Temporal Compression:

The hand gestures defined are believed to have smooth acceleration waveforms since the hand follows a smooth trajectory while performing the gestures. However, acceleration data acquired suffers from abrupt changes due to hand shaking or accelerometer noise which needs to be eliminated.

The acceleration time series are temporally compressed by an averaging window of 70 ms and moves at a 30 ms step. Temporal compression, basically, filters out any variations not intrinsic to the gesture itself and further reduces the size of the acceleration signals which results in a grand reduction in computational cost especially in DTW.

2.2. Dynamic Time Warping:

Dynamic time warping (DTW) is an algorithm that measures the similarity between two time sequences of different durations. In other words, DTW matches two time signals by computing a temporal transformation causing the signals to be aligned. The alignment is optimal in the sense that a cumulative distance measure between the aligned samples is minimized [8].

Assume we have two time sequences, X and Y, of length *n* and *m*, respectively, where $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_m\}$ and we want to compute the matching cost: DTW(X, Y). The matching cost is computed based on dynamic programming using the following formulation:

$$D(i,j) = d(x_i, y_j) + min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} \qquad (1)$$

where the distance function d(·,·) varies with the application. In our gesture recognition system, $d(x_i, y_j)$ is defined as,

$$d(x_i, y_j) = (x_i - y_j)^2 \qquad (2)$$

and consequently,

$$DTW(X,Y) = D(n,m) \qquad (3)$$

For further details and thorough explanation of DTW, the reader is referred to [8].

2.3. Affinity Propagation:

Affinity propagation (AP) [9] is an algorithm which, unlike all other clustering techniques, simultaneously considers all data points as potential exemplars and recursively transmits real-valued messages until a good set of exemplars and clusters emerge.

AP operates primarily on a matrix **S** whose entities are real-valued similarities between data points, in which *S*(*i,j*) indicates the similarity between data points x_i and x_j. Clustering is based on the exchange of two type of messages: *responsibility* where *r(i,j)* indicates how well-suited point *j* is to serve as an exemplar for point *i*, and *availability* where *a(i,j)* indicates how appropriate it would be for point *i* to choose point *j* as its exemplar. In addition to **S**, AP takes as an input a real number *p* referred to as *preference* or *self-similarity* so that data points with larger values of *p* are more likely to be chosen as exemplars. Generally, *p* takes on the value of the median of the input similarities in case all data points are equally likely to be chosen as exemplars. As far as our gesture recognition system is concerned, the value of *p* is chosen to be *p* = 0.01**p'*, where *p'* stands for the median of the input similarities in both cases: user-dependent and user-independent recognition. As for the similarity function, it is defined as the negative of the cost computed by DTW,

$$S(i,j) = -1 * \left(\left(DTW(i_x, j_x)\right)^2 + \left(DTW(i_y, j_y)\right)^2 + \left(DTW(i_z, j_z)\right)^2 \right) \ \forall\, i,j \in \{1, \dots, N\} \qquad (4)$$

where *N* is the total number of repetitions in the training set.

2.4. Compressive Sensing:

The premise here is that hand gestures are believed to be sparse since the hand follows a smooth trajectory while performing a gesture and therefore, CS can be implemented to recognize a repetition of a gesture.

CS provides a novel framework for recovering sparse or compressible signals, with much fewer noisy measurements than that needed by the nyquist rate [10][11]. If the unknown gesture repetition is denoted by $\mathcal{Y}$ and if we let **R** denote the matrix whose columns represent template repetitions pertaining to the gestures in our dictionary. Assuming that $\mathcal{Y}$ is a replica or a close resemblance of one of the repetitions in **R**, then $\mathcal{Y}$ can be related to **R** by,

$$\mathcal{Y} = \boldsymbol{R}\theta \qquad (5)$$

and assuming that the number of repetitions in **R** is *P*, then θ is a 1-sparse *P* x 1 vector whose elements are all zeros except $\theta(n) = 1$, where *n* is the index of the repetition which $\mathcal{Y}$ best resembles, namely,

$$\theta = [0, \dots, 0, 1, 0, \dots, 0]^T \qquad (6)$$

where the superscript *T* denotes transposition. Recall that the gesture repetitions are of different lengths and in order to solve (5), $\mathcal{Y}$, **R**, and θ have to be of compatible sizes. One solution to tackle this problem is to find the maximum length, *l*, among repetitions in **R** and $\mathcal{Y}$. The shorter repetitions are then padded with zeros until they have an *l*-length. This process is repeated for every $\mathcal{Y}$.

Recall also that gestures suffer from inherent temporal variations and therefore the above ideal example of having an unknown repetition replicating a template repetition doesn't exist in a real scenario. Yet, it should be of close resemblance. Therefore, the problem can be formulated as follows:

$$\mathcal{Y} = \boldsymbol{R}\theta + \varepsilon \qquad (7)$$

where ε is the measurement noise.

Using the same formulation as in [12], we introduce the preprocessor W (*i.e.* $Y = W\mathcal{Y}$), which is defined as,

$$W = QR^{\dagger} \qquad (8)$$

where $Q = orth(R^T)^T$, and *orth*(**R**) is an orthogonal basis for the range of **R** and $R^{\dagger}$is the pseudo-inverse of the matrix **R**. θ can be well recovered from Y with a high probability through the following ℓ_1 minimization problem,

$$\hat{\theta} = \arg\min ||\theta||_1, \ s.t.\ Y = Q\,\theta + \varepsilon' \qquad (9)$$

where $\varepsilon' = W\varepsilon$.

For a 3-axis accelerometer, the acceleration waveforms constitute three signals: in the x-, y-, and z- directions. Therefore, when applying the above formulation to the gesture recognition problem, $\mathcal{Y}$ will in fact be three vectors, **R** will be three matrices and θ will be three vectors. Since θ is of three vectors, then

$$\theta_{eq} = \theta_x + \theta_y + \theta_z \qquad (10)$$

For user-independent recognition, entities of θ_{eq} that belong to the same user for each gesture are added up as well resulting in θ_{eq}', and then the gesture with the highest θ_{eq}' is recognized as the correct gesture.

3. PROTOTYPE IMPLEMENTATION

The acceleration data corresponding to the different gestures is collected using a Wiimote, which has a built-in 3-axis accelerometer. A gesture repetition starts by pressing and holding the "trigger" button or "B" button on the bottom of the remote and it ends by releasing the button and hence the problem of gesture spotting is solved.

3.1. Gesture Vocabulary:

A dictionary of 18 gestures is created as shown in figure 1. Our dictionary of gestures is the largest in published studies for accelerometer-based gesture recognition systems. The defined gestures are not limited only to one plane as in other studies [6][7], but span the two planes: XZ and YZ planes.

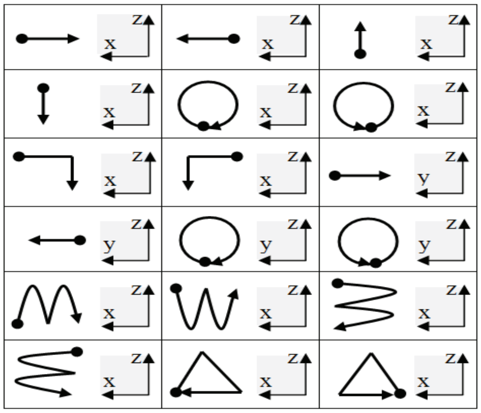

Figure 1. The dictionary of 18 gestures.

3.2. Gesture Database Collection:

The database consists of 3,780 repetitions and is built by acquiring gestures from 7 participants (2 females and 5 males) using the Wiimote. Each participant is asked to repeat each gesture 30 times resulting in a total of 540 repetitions for all gestures per participant or a total of 210 repetitions from all participants per gesture.

A gesture's acceleration waveforms from the same person can differ drastically if the tilting angle of the accelerometer is big. Therefore, all participants are asked to try their best to perform the gestures without any, or with minimal, tilting of the remote.

4. SIMULATION & RESULTS

4.1. Training Phase:

Training is the first main phase in our proposed gesture recognition system and is depicted in figure 2.

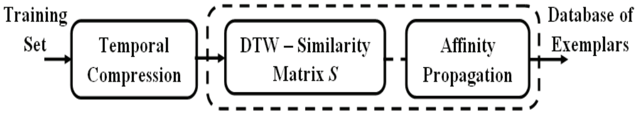

Figure 2. Training Phase.

User Dependent Training: Data pertaining to each user is treated separately as follows: The training set is created by randomly choosing *M* repetitions for each gesture. All the chosen repetitions are then temporally compressed in order to remove any noise that might have been added by inevitable hand shaking or minor remote tilting. DTW is then implemented to find the similarity between any two repetitions and thus forming the similarity matrix **S**. Next, AP works on **S** and partitions the training set into different clusters each represented by an exemplar. AP in the case of user-dependent recognition is forced to partition the data into *N* clusters where *N* is the number of gestures in the dictionary. In other words, all repetitions pertaining to one gesture form a cluster and thus, the output of the training phase is a set of *N* exemplars, one for each gesture.

User Independent Training: The training set is created by randomly choosing *M* repetitions from each gesture for *K < 7* users only, resulting in a total of *K*M* repetitions for each gesture. Again, all repetitions are temporally compressed for noise removal and DTW is implemented to generate the similarity matrix. Next, AP is run to partition the repetitions into different clusters. However, in this case, although AP is forced to create a cluster for each gesture, it doesn't always succeed and as a result, repetitions for one gesture may form different clusters, but all repetitions from one user end up in the same cluster. As a result, unlike user-dependent recognition, the output of the training phase is an arbitrary number of exemplars.

4.2. Testing Phase:

Testing is the second main phase in our proposed gesture recognition system and is depicted in figure 3.

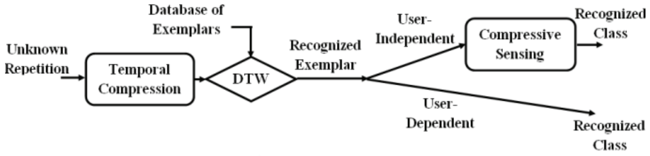

Figure 3. Testing Phase

User-Dependent Recognition: The testing set is formed by putting together all the repetitions that were not used in the training stage. Testing is done on each user separately and the average accuracy among all users is computed. So, an unknown repetition undergoes the same preprocessing of temporal compression and then compared by DTW to the *N* exemplars that were found in the training phase. The unknown repetition gets classified to the gesture whose exemplar yields the lowest cost. For the seven participants, *M* is varied in order to examine the dependence of the system's performance on the number of training repetitions per gesture. Figure 4 shows a graph of the average recognition rate against the number of training repetitions per gesture. The graph shows that with as minimum as three repetitions, the recognizer is capable of achieving an accuracy of 99.79%.

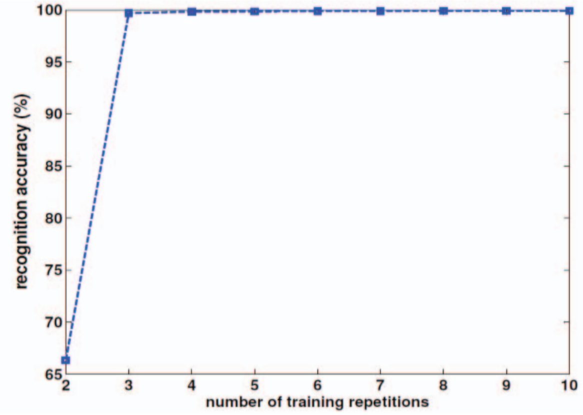

Figure 4. Average accuracy against the number of training repetitions for user-dependent recognition.

User-Independent Recognition: The testing set is generated by putting together all the repetitions that were not used in the training stage specific to the *K* chosen users and adding to them all the repetitions from the remaining users. So, an unknown repetition is first temporally compressed and then compared by DTW to the exemplars that were found in the training stage. All the repetitions that fall in the cluster whose exemplar gives the lowest cost are recovered and used to construct the matrix **R** in (7) and then the unknown gesture is recognized as explained in section 2.4. Figure 5 shows a graph of the average recognition rate against the number of gestures for user-independent gesture recognition. Simulations were run with $K = 3$. Three plots are shown: top one based on the remaining repetitions belonging to the users whose data is used to train the system only, bottom one based on all the repetitions from the other users only, and the middle one based on repetitions from both users.

5. CONCLUSION

In conclusion, we have proposed a novel gesture recognition system. The system utilizes a single 3-axis accelerometer and thus can be readily implemented on any commercially available consumer device that has a built-in accelerometer. The system employs dynamic time warping and affinity propagation algorithms for efficient training of the system and utilizes the sparse nature of the gesture sequence by implementing compressive sensing for user-independent gesture recognition. The system is tested on a dictionary of 18 gestures whose database contains over 3,700 repetitions collected from 7 users. For some users, the proposed system achieves an accuracy of 100% when carrying out user-dependent recognition. As for user-independent recognition, an accuracy is obtained that is competitive with many systems that are available in literature.

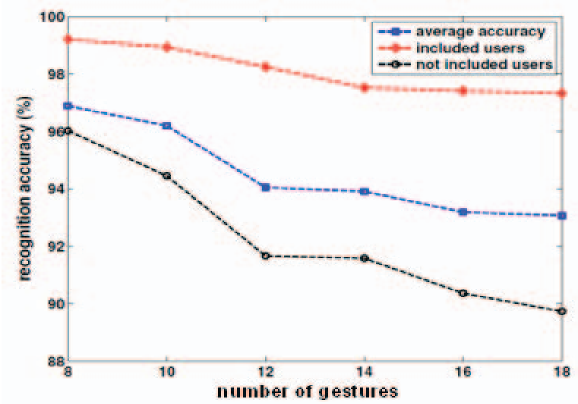

Figure 5. Average accuracy against the number of **gestures** for user-independent recognition.

6. REFERENCES

[1] "Apple – iPhone," 2009. [Online]. Available: http://www.apple.com/iphone/iphone-3gs/high-technology.html

[2] "Lenovo – Active Protection System." [Online]. Available: http://www.pc.ibm.com/ca/think/thinkvantagetech/aps.html

[3] A. Y. Yang, S. Iyengar, S. Sastry, R. Bajcsy, P. Kuryloski, and R. Jafari, "Distributed segmentation and classification of human actions using a wearable motion sensor network," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshop,* pp. 1-8, June 2008.

[4] Z. Xu, C. Xiang, W. Wen-hui, Y. Ji-hai, V. Lantz, W. Kong-qiao, "Hand Gesture Recognition and Virtual Game Control Based on 3D Accelerometer and EMG Sensors," *International Conference on Intelligent User Interfaces***,** pp. 401 - 406, February 2009.

[5] T. Pylyanainen, "Accelerometer Based Gesture Recognition Using Continuous HMMs," International Conference on Pattern Recognition and Image Analysis, pp. 639–646, 2005.

[6] J. Liu, Z. Wang, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uWave: Accelerometer-based personalized gesture recognition and its applications," in *IEEE PerCom*, 2009.

[7] J. Kela, P. Korpipää, J. Mäntyjärvi, S. Kallio, G. Savino, L. Jozzo, and D. Marca, "Accelerometer-based gesture control for a design environment," *Personal Ubiquitous Computing*, vol. 10, pp. 285-299, 2006.

[8] E. Keogh, C. A. Ratanamahatana, "Exact Indexing of Dynamic Time Warping," Knowledge and Information Systems, Hong Kong, China, pp. 406-417, August 2002.

[9] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 1, pp. 972–976, February 2007.

[10] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, pp. 21–30, March 2008.

[11] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Processing Magazine*, pp. 14–20, March 2008.

[12] C. Feng, S.W.A. Au, S. Valaee, and Z.H. Tan, "Orientation-aware Localization Using Affinity Propagation and Compressive Sensing", IEEE CAMSAP, December 2009, Aruba.