uWave: Accelerometer-based Personalized Gesture Recognition

Technical Report TR0630-08, Rice University and Motorola Labs, June 2008

Jiayang Liu*, Zhen Wang*, and Lin Zhong
Department. Of Electrical Computer Engineering
Rice University, Houston TX 77005
{jiayang, zhen.wang, lzhong}@rice.edu
*with equal contribution

Jehan Wickramasuriya and Venu Vasudevan
Pervasive Platforms & Architectures Lab
Applications & Software Research Center
Motorola Labs
{jehan,venu}@motorola.com

Abstract
The proliferation of low power, low cost accelerometers on consumer electronics has brought an opportunity to personalize gesture-based interaction. We present uWave, an efficient personalized gesture recognizer based on a 3-D accelerometer. The core technical components of uWave include quantization of accelerometer readings, dynamic time warping and template adaptation. Unlike statistical methods, uWave requires a single training sample and allows users to employ personalized gestures. Our evaluation is based on a large gesture library with over 4000 samples collected from eight users. It shows that uWave achieves 98.6% accuracy, competitive with statistical methods which require significantly more training samples.

Keywords: gesture recognition, acceleration, dynamic time warping, personalized gesture

1 Introduction

Hand gesture is natural for human users to express themselves and interact with others. It has recently become attractive for spontaneous interaction with consumer electronics and mobile devices. However, there are multiple technical challenges to gesture-based interaction. Firstly, unlike many pattern recognition problems, e.g. speech and handwriting recognition, gesture recognition does not enjoy a standardized or widely accepted "vocabulary". Therefore, it is often desirable and necessary for users to create their own gestures, thus *personalized gesture recognition*. With personalized gestures, it is difficult to collect a large set of training samples which is necessary for established statistical methods, e.g., Hidden Markov Model (HMM) [14, 7, 6]. Moreover, spontaneous gesture-based interaction requires immediate engagement, i.e., the overhead of setting up the recognition instrumentation should be minimal. More importantly, the targeted application platforms for personalized gesture recognition are usually highly constrained in cost and system resources, including battery, computing power, and alternative interfaces, e.g. buttons. As a result, computer vision or "glove" based solutions are usually unsuitable.

In this work, we present uWave to address these challenges and focus on gestures without regard to finger movement, such as sign languages. Our goal is to support efficient personalized gesture recognition on a wide range of devices, in particular, on resource-constrained systems. Unlike statistical methods [6], uWave only requires a single training sample to start; unlike computer vision-based methods [16], uWave only employs a three-axis accelerometer that has already appeared in numerous consumer electronics, e.g. Nintendo Wii remote, and mobile device, e.g. Apple iPhone. uWave matches the accelerometer readings for an unknown gesture with those for a vocabulary of known gestures, or *templates*, based on dynamic time warping (DTW) [10]. uWave is efficient and thus amenable to implementation on resource-constrained platforms. We have implemented a prototype of uWave using the Nintendo Wii remote hardware [11]. Our measurement shows that uWave recognizes a gesture from an eight-gesture vocabulary in 2ms on a modern laptop, 4ms on a Pocket PC, and 300ms on a 16-bit microcontroller, without any complicated optimization.

We evaluate uWave with a pre-defined vocabulary of simple gestures reported in [6] and with a library of 4480 gestures collected from eight participants over multiple weeks. The study shows that uWave achieves accuracy of 98.6% with template adaptation and 93.5% without template adaptation for user-dependent gesture recognition. It also shows uWave is much less successful for user-independent recognition (75% accuracy).

In summary, we make the following original technical contributions.

- We present uWave, an efficient gesture recognition method based on a single accelerometer, quantization, and dynamic time warping (DTW). uWave requires a single training sample per vocabulary gesture. We also present two simple adaptation methods to accommodate gesture variations over the time.
- We show that there are considerable variations in gestures collected over long time and in gestures collected from multiple users; we highlight the importance of adaptive and user dependent recognition.
- We report an extensive evaluation of uWave with over 4000 gesture samples collected from eight users over multiple weeks for a predefined vocabulary of simple gestures. The evaluation shows that uWave is very effective and efficient for user-dependent recognition.

The strength of uWave in user-dependent gesture recognition makes it ideal for personalized gesture-based interaction. With uWave, users can create simple personal gestures for frequent interaction. Its simplicity, efficiency, and minimal hardware requirement (a single accelerometer) has the potential to enable personalized gesture-based interaction with a broad range of devices.

The rest of the paper is organized as follows. We discuss related work in Section 2 and then present the technical details of uWave in Section 3. We next describe a prototype implementation of uWave using the Wii remote in Section 4. We report an evaluation of uWave through a large database for a predefined gesture vocabulary of eight simple gestures in Section 5. We address the limitations of uWave and acceleration-based gesture recognition in general in Section 6 and conclude in Section 7.

2 Related Work

Gesture recognition has been extensively investigated [2]. The majority of the past work has focused on detecting the contour of hand movement. Computer vision techniques in different forms have been extensively explored in this direction [16]. For a recent example, VisionWand [3] employs computer vision to recognize the movement of a passive wand with a predefined color pattern. While the most common form requires one or more cameras to capture hand movement, the Wii remote has the “camera” (IR sensor) inside the remote and detects motion by tracking the relative movement of IR transmitters mounted on the display. Therefore, it basically maps the three-dimensional remote movement onto a planar surface. This translates a “gesture” into “handwriting”, lending itself to a rich set of handwriting recognition techniques. Vision-based methods, however, are fundamentally limited by their hardware requirements (i.e. cameras or transmitters) and relatively high computation load.

“Smart glove” based solutions [13] have been investigated to recognize very fine gestures, for example the finger movement and conformation, instead of hand movement. These solutions require the user to wear a glove tagged with multiple sensors to capture the motion of fingers and hand in fine granularity. While they often yield impressive accuracy, these solutions are inadequate for spontaneous interaction with consumer electronics and mobile devices, because of the high cost of the glove and the high overhead of engagement.

As ultra low power, low cost accelerometers, gyroscopes, and compasses start to appear on consumer electronics and mobile devices, many have recently investigated gesture recognition based on the time series of acceleration, often with additional information from a gyroscope or compass. Signal processing and ad hoc recognition methods were explored in [8]. LiveMove Pro [21] from Ailive provides a gesture recognition library based the accelerometer in the Wii remote. Unlike uWave, LiveMove Pro targets at user-independent gesture recognition with predefined gesture classifiers and requires 5 to 10 training samples. No systematic evaluation of the accuracy of LiveMove Pro exists. HMM, investigated in [5, 7, 6, 18], is the mainstream method for speech recognition. However, HMM-based methods require extensive training data to be effective. The authors of [7] realized this and attempted to address it by converting two samples into a large set of training data by adding Gaussian noise. While the authors showed improved accuracy, the effectiveness of this method is likely to be highly limited because it essentially assumes that variations in human gestures are Gaussian. In contrast, uWave requires as few as a single training sample for each gesture and delivers competitive accuracy. Another limitation of HMM-based methods is that they often require knowledge of the vocabulary in order to configure the models properly, e.g. the number of states in the model. Therefore, HMM-based methods may suffer when users are allowed to choose gestures freely. Moreover, as we will see in the evaluation section, the evaluation dataset and the test procedure used in [6, 7, 18] did not consider gesture variations over the time. Thus their results can be overly optimistic.

Dynamic time warping (DTW) is the core of uWave. It was extensively investigated for speech recognition in the 1970s and early 1980s [10], in particular speaker-dependent speech recognition with a limited vocabulary. Later, HMM-based methods became the mainstream because they are more scalable toward a large vocabulary and can better benefit from a large set of training data. However, DTW is still very effective in coping with limited training data and a small vocabulary, which matches up well with personalized gesture-based interaction with consumer electronics and mobile devices.

Wilson and Wilson applied DTW and HMM with XWand [18] to user-independent gesture recognition. The low accuracies, 72% for DTW and 90% for HMM with seven training samples, render them almost impractical. In contrast, uWave focuses on personalized and user-dependent gesture recognition, thus achieving much higher recognition accuracies. It is also important to note that the evaluation data set employed in this work is considerably more extensive than previously reported work, including [6, 7, 18]

Similar to uWave, the “$1 recognizer” presented in [19] was also based on template matching. It is important to note that “gestures” in that work refer to handwritings on touch screen, instead of three-dimensional free-hand movement. The $1 recognizer is related to uWave in multiple aspects. First, although it is also based on template matching, the $1 recognizer may not apply to time series of accelerometer readings, which are subject to temporal dynamics (how fast and forceful the hand moves), three-dimensional acceleration data due to movement of six degrees of freedom, and the confusion introduced by gravity. Second, the authors concluded that DTW is slower but achieves similar accuracy as the $1 recognizer. We show that with proper quantization, DTW-based uWave can be

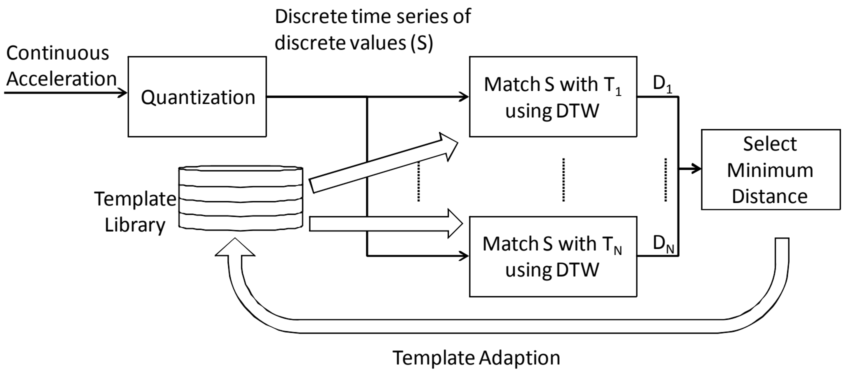

Figure 1: uWave is based on acceleration quantization, template matching with DTW, and template adaptation

extremely efficient. Moreover, uWave allows DTW to start as the very first point of the time series comes in and to proceed as more points are available. As a result, the delay can be masked by the much slower hand movement. Third, three-dimensional gestures may be projected onto a surface as handwritings. Therefore, the $1 recognizer can potentially be applied to recognize certain gestures. However, this will suffer from the limitation of vision-based gesture recognition. In contrast, uWave is completely camera free and recognizes three-dimensional free hand movement. Fourth and most importantly, uWave and $1 recognizer are related in their focus on personalization. While [19] is mostly focused on recognition accuracy and speed, we are interested in the interaction between uWave and the personalizing process as our user studies are designed for. Moreover, we also investigate broader issues that concern accelerometer-based gesture recognition, such as user dependence, tilt, and vocabulary selection.

3 uWave Algorithm Design

In this section, we present the key technical components of uWave: acceleration quantization, dynamic time warping (DTW), and template adaptation. The premise of uWave is that *human gestures can be characterized by the time series of forces applied to the handheld device*. Therefore, uWave bases the recognition on the matching of two time series of forces, measured by a single three-axis accelerometer.

For recognition, uWave leverages a *template library* that stores one or more time series of known identities for every vocabulary gesture, often input by the user. Figure 1 illustrates the recognition process. The input to uWave is a time series of acceleration provided by a three-axis accelerometer. Each time sample is a vector of three elements, corresponding to the acceleration along the three axes. uWave first quantizes acceleration data into a time series of discrete values. The same quantization applies to the templates too. It then employs DTW to match the input time series against the templates of the gesture vocabulary. It recognizes the gesture as the template that provides the best matching. The recognition results, confirmed by the user as correct or incorrect, can be used to adapt the existing templates to accommodate gesture variations over the time.

Acceleration Data (a)	Converted Value
a > 2g	16
g < a < 2g	11~15 (five levels linearly)
0 < a < g	1~10 (ten levels linearly)
a = 0	0
-g < a < 0	-1~-10 (ten levels linearly)
-2g < a < -g	-11~-15 (five levels linearly)
a < -2g	-16

Table 1: uWave quantizes acceleration data in a non-linear fashion before template matching

3.1. Quantization of Acceleration Data

uWave quantizes the acceleration data before template matching. Quantization reduces the length of input time series for DTW in order to improve computation efficiency. It also converts the accelerometer reading into a discrete value thus reduces floating point computation. Both are desirable for implementation in resource-constrained embedded systems. Quantization improves recognition accuracy by removing variations not intrinsic to the gesture, e.g. accelerometer noise and minor hand tilt.

uWave quantization consists of two steps. In the first step, the time series of acceleration is temporally compressed by an averaging window of 50ms that moves at a 30ms step. This significantly reduces the length of the time series for DTW. The rationale behind it is that intrinsic acceleration produced by hand movement does not change erratically; and rapid changes in acceleration are often caused by noise and minor hand shake/tilt. In the second step, the acceleration data is converted into one of 33 levels. Non-linear

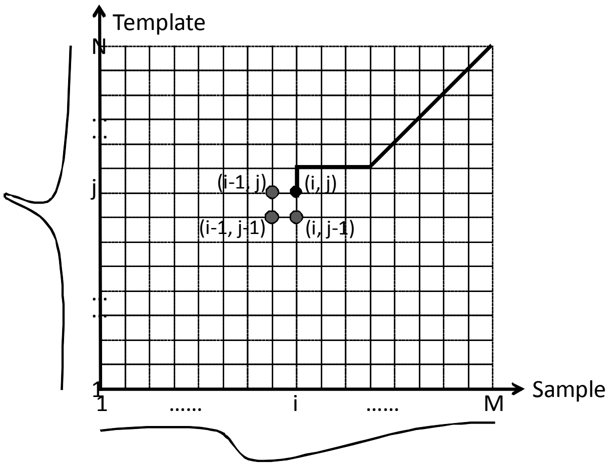

Figure 2: Dynamic time warping of time series for best matching

quantization is employed because we find that most samples are between –g and +g and very few go beyond +2g or below –2g. The non-linear conversion table is presented in Table 1.

3.2. Dynamic Time Warping

Dynamic time warping is a classical algorithm based on dynamic programming to match two time series with temporal dynamics [10], given the function for calculating the distance between two time samples. uWave employs the Euclidean distance for matching quantized time series of acceleration. Let S[1…M] and T[1…N] denote the two time series. As shown in Figure 2, any matching between S and T with time warping can be represented as a monotonic path from the starting point (1, 1) to the end point (M, N) on the M by N grid. A point along the path, say (i, j), indicates that S[i] is matched with T[j]. The matching cost at this point is calculated as the distance between S[i] and T[j]. The path must be monotonic because the matching can only move forward. The similarity between S and T is evaluated by the minimum accumulative distance of all possible paths, or *matching cost*.

DTW employs dynamic programming to calculate the matching cost and find the corresponding optimal path. As illustrated in Figure 2, the optimal path from (1, 1) to point (i, j) can be obtained from the optimal paths from (1, 1) to the three predecessor candidates, i.e. (i-1, j), (i, j-1), (i-1, j-1). The matching cost from (1, 1) to (i,j) is therefore the distance at (i, j) plus the smallest matching cost of the predecessor candidates. The time complexity and space complexity of DTW are both O(M·N).

3.3. Template Adaptation

As we will show in the evaluation section, there are considerable variations between gesture samples by the same user collected from different days. Ideally, uWave should adapt its templates to accommodate such time variations. Template adaption of DTW for speech recognition has been extensively studied, e.g. [20,9], and proved to be effective. In this work, however, we only devise two simple schemes to adapt the templates. *Our objective is not to explore the most effective adaptation methods but to demonstrate the template adaptation can be easily implemented and effective in improving recognition accuracy over multiple days.*

Our template adaptation works as follows. uWave keeps two templates generated in two different days for each vocabulary gesture. It matches a gesture input with both templates of each vocabulary gesture and take the smaller matching cost of the two as the matching cost between the input and vocabulary gesture.

Each template has a timestamp of when it is created. On the first day, there is only one training sample, or template, for each gesture. As the user input more gesture samples, uWave updates the templates based on how old the current templates are and how well they match with new inputs. We develop two simple updating schemes. In the first scheme, if both templates for a vocabulary gesture in the library are at least one day old and the input gesture is correctly recognized, the older one will be replaced by the newly correctly recognized input gesture. We refer to this scheme as *Positive Update*. The second scheme differs from the first one only in that we replace the older template with the input gesture when it is incorrectly recognized. We call this scheme *Negative Update*. Positive Update only requires the user to notify uWave when recognition result is incorrect. Negative Update requires the user to point out the correct gesture when a recognition error happens, e.g. by pressing a button corresponding to the identity of the input sample.

4 Prototype Implementation

We have implemented a prototype of uWave using the Wii remote as the interaction device. The Wii remote has a built-in three-axis accelerometer from Analog Devices, ADXL330 [17]. The accelerometer has a range of -3g to 3g and noise below 3.5mg when operating at 100Hz [1]. The Wii remote can send the acceleration data and button actions through Bluetooth to a PC in real time. We implement uWave and its variations on a Windows PC using Visual C#. The implementation is about 300 lines of code. The prototype detects the start of a gesture when the 'A' button on the Wii remote is pressed; and detects the end when the button is released. While our prototype is based on the Wii remote hardware, uWave can be implemented with any device with a three-axis accelerometer of proper sensitivity and range as are those found in most consumer electronics and mobile devices.

4.1. Recognition Speed

The structure of uWave allows the quantization and matching to start with the very first sample in the unknown time series and proceed as more samples come in. Nevertheless,

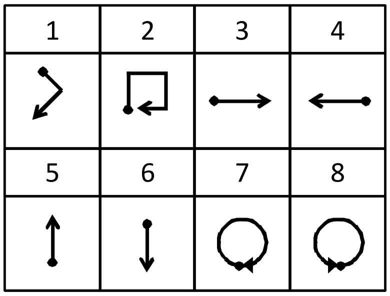

Figure 3: Gesture vocabulary adopted from [KKM+06]. The dot denotes the start and the arrow the end

our prototype does not leverage this: it starts after a complete time series is collected. Even so, it gives out recognition result without perceptible delay in our experiments based on PCs. We measured the speed of uWave implemented in C on multiple platforms. On a Lenovo T60 with 1.6GHz Core 2 Duo, it takes less than 2ms for a template library of eight gestures on a T-Mobile MDA Pocket PC with Windows Mobile 5.0 and 195MHz TI OMAP processor, it takes about 4ms for the same vocabulary. Such latencies are too short to be perceptible to human users. We also tested uWave on an extremely simple 16-bit microcontroller in the Rice Orbit sensor [12], TI MSP430LF1611. The delay is about 300ms. While this may be perceptible to the user, it can be easily masked if uWave is implemented to proceed at the same time as samples come in.

5 Evaluation

We next present our data evaluation of uWave for a vocabulary of predefined gestures based on the prototype described above.

5.1. Gesture Vocabulary from Nokia

We employ a set of eight simple gestures identified by a Nokia research study [6] as preferred by users for interaction with home appliances. The work also provided comprehensive evaluation of HMM-based methods so that a comparison with uWave is possible. Figure 3 shows these gestures as the paths of hand movement.

5.2. Gesture Database Collection

We collected gestures corresponding to the Nokia vocabulary from eight participants with the Wii remote-based prototype. Two of them are undergraduates and others are graduate students; all but one are males. They are in 20s or early 30s, right handed.

The gesture database was collected via the following procedure. For a participant, gestures are collected from seven days within a period of about three weeks. On each day, the participant holds the Wii remote in hand and repeats each of the eight gestures in the Nokia vocabulary ten times. The database consists of 4480 gestures in total and 560 for each participant. This database provides us a statistically significant benchmark for evaluating the recognition accuracy.

It is important to note that the dataset used in [6] consists of 30 samples for each gesture collected from a single user. All of the 30 samples for the same gesture were collected on the same day (the entire dataset of eight gestures were collected over two days). As we will highlight in this work, users exhibit high variations in the same gesture over the time. Samples for the same gesture from the same day cannot capture this and may lead to overly optimistic recognition results.

5.3. Recognition without Adaptation

We first report recognition results for uWave without template adaptation.

5.3.1. Test Procedure

Because our focus is personalized gesture recognition, we evaluate uWave using the gestures from each subject separately. That is, the samples from a participant are used to provide templates and test samples for the same subject.

We employ Bootstrapping [4] to further improve the statistical significance of our evaluation. The following procedure applies to each participant separately. For clarity, let us label the samples for each gesture by the order they were collected. For the i^{th} test, we use the i^{th} sample for each gesture from the participant to build eight templates and use the rest samples from the same participant to test uWave. As *i* is from 1 to 70 (10 times by 7 days), we have 70 tests for each participant. Each test produces a confusion matrix that shows the percentage of times how a sample is recognized. We average the confusion matrixes for the 70 tests to produce the confusion matrix for each participant.

We average confusion matrixes of all eight participants to produce the final confusion matrixes. Figure 5 (Left) summarizes the recognition results of uWave over the database for the Nokia gesture vocabulary. In the matrixes, columns are recognized gestures and rows are the actual identities of input gestures.

uWave achieves an average accuracy of 93.5%. Figure 5 (Left) also shows that gesture 1, 2, 6 and 7 have lower recognition accuracy in that they involve similar hand movement as each other, e.g. both gesture 1 and gesture 6 are featured by waving down movement. A closer look into the confusion matrixes for each participant reveals large variation (9%) in recognition accuracy among different participants. We observed that the participant with the highest accuracy performed the gestures in larger amplitude and slower speed compared to other participants.

Our evaluation also shows the effectiveness of quantization, i.e., temporal compression and non-linear conversion, of the raw acceleration data. The former speeds up the recognition process by more than nine times without negative impact on accuracy, and the latter improves the average

	↙	↲	→	←	↑	↓	↻	↺
↙	92.1	0.1	2.4	1.9	0.1	2.9	0.6	0.1
↲	1.6	91.6	1.3	1.1	0.7	0.4	2.7	0.6
→	0.5	0	95.9	1.2	0.7	1.7	0	0
←	0.3	0	1.6	96.2	0.7	1.1	0	0.1
↑	0.3	0	1.5	0.6	97.0	0.5	0	0.1
↓	2.4	0	2.4	2.3	1.0	91.7	0.1	0
↻	3.4	1.9	2.6	1.7	0.4	0.7	89.2	0
↺	1.1	0.6	1.7	0.9	0.8	0.7	0	94.2

	↙	↲	→	←	↑	↓	↻	↺
↙	98.4	0	0.3	0.4	0	0.4	0.3	0.2
↲	0.5	98.3	0.2	0	0.3	0.1	0.4	0.1
→	0.2	0	98.3	0.6	0.1	0.6	0.2	0
←	0.2	0	0.3	98.8	0.3	0.2	0.2	0
↑	0.4	0	0.2	0.4	98.7	0.1	0.2	0
↓	0.7	0	0.6	0.5	0.3	97.7	0.2	0
↻	0.5	0.4	0.4	0.1	0.1	0.3	98.1	0.2
↺	0.2	0.1	0.1	0.2	0	0	0.2	99.2

Figure 5: Confusion matrixes for the Nokia vocabulary without adaptation. Columns are recognized gestures and rows are the actual identities of input gestures. (Left) Tested with samples from all days (average accuracy is 93.5%); (Right) Tested with samples from the same day as the template (average accuracy is 98.4%)

	↙	↲	→	←	↑	↓	↻	↺
↙	96.8	0	1.5	0.3	0	1.1	0	0.2
↲	0.7	96.4	0.5	0.2	0.2	0.4	1.2	0.5
→	0	0	98.9	0.6	0	0.5	0	0
←	0.2	0	0.3	98.9	0.2	0.5	0	0
↑	0.2	0	0.2	0.1	99.3	0.2	0	0
↓	0.6	0	0.6	0.3	1.7	96.8	0	0
↻	0.8	2.0	2.0	0.4	0	0.2	94.6	0
↺	1.0	0.4	1.1	0.4	0	0	0	97.1

	↙	↲	→	←	↑	↓	↻	↺
↙	97.7	0	1.2	0.6	0	0.6	0	0
↲	0.6	98.6	0.2	0.1	0	0.1	0.3	0.1
→	0.1	0	99.1	0.4	0.1	0.4	0	0
←	0.1	0	0.4	99.0	0.1	0.4	0	0
↑	0.2	0	0.3	0.1	99.2	0.2	0	0
↓	0.5	0	0.4	0.2	0.5	98.3	0	0.1
↻	0.4	0.5	07	0.2	0.1	0.2	98.0	0
↺	0.2	0	0.3	0.4	0.1	0.1	0	98.9

Figure 4: Confusion matrixes for the Nokia vocabulary with adaptation, tested with samples from all days. Columns are recognized gestures and rows are the actual identities of input gestures. (Left) Positive Update (average accuracy is 97.4%); (Right) Negative Update (average accuracy is 98.6%)

accuracy for eight participants by 1% and further speed up the recognition.

5.3.2. Evaluation using Samples from the Same Day

To highlight how gesture variations from the same user over multiple days impact the gesture recognition, we modify the test procedure above so that when a sample is chosen as the template, uWave is tested only with other samples collected in the same day.

Figure 5 (Right) summarizes the recognition results averaged cross all eight participants. It shows a significantly higher accuracy (98.4%) than that of using samples from all different days. *The difference between Figure 5 (Left) and Figure 5 (Right) highlights the possible variations for the same gesture from the same user over multiple days and the challenge it poses to recognition.* This also indicates that the results reported by some previous work, e.g. [7, 6], were overly optimistic because the evaluation dataset was collected over a very short time.

The same-day accuracy of 98.4% by uWave with one training sample per gesture is comparable to HMM-based methods with 12 training samples (98.6%) reported in [6]. It is worth noting that the accelerometer in Wii remote provides comparable accuracy but larger acceleration range (-3g to

3g) than that used in [6] (-2g to 2g). In reality, however, the acceleration produced by hand movement rarely exceeds the range from -2g to 2g. Hence, the impact of difference in the accelerometers on the accuracy should be insignificant.

5.4. Recognition with Adaptation

The considerable difference between Figure 5 (Left) and Figure 5 (Right) motivates the use of template adaptation to accommodate variations over the time in order to achieve accuracy close to that in Figure 5 (Right). We report the results next.

Again, we evaluate uWave with adaptation for each participant separately. Because the adaption is time-sensitive, we have to apply Bootstrapping in a more limited fashion. Let us label the days in which a participants' gestures were collected by the time order, from one to seven. For the *i*th test, we assume the evaluation starts on the *i*th day and applies the template adaptation in the following days, from (*i+1*)th to 7th and then from 1st to (*i-1*)th. We have seven tests for each participants and each produces a confusion matrix. We average them to produce the confusion matrix for each participant and average the confusion matrixes of all participants for the final one.

Figure 4 summarizes the recognition results averaged across all eight participants. It shows an accuracy of 97.4% for Positive Update and 98.6% for Negative Update, significantly higher than that without adaptation (Figure 5 Left) and close to that tested with samples from the same day (Figure 5 Right). While template adaptation requires user feedback when a recognition error happens, the high accuracy indicates that it is needed only for 2-3% of all the test samples.

6 Discussion

We next address the limitation of uWave and gesture recognition based on accelerometers in general.

6.1. Gestures and Time Series of Forces

Resulted from lack of a standardized gesture vocabulary, human users may have diverse opinions on what constitute of a unique gesture. As noted early, the premise of uWave is that human gestures can be characterized as time series of forces applied to handheld device. With this view, the temporal dynamic of gestures is more similar to speech in nature than to handwritings, which are usually recognized by human users as the final contours without regard to the time sequence of the contours. However, it is important to note that while one may produce the three-dimensional contour of the hand movement given a time series of forces, the same contour may be produced by very different time series of forces. In particular, the contour can be produced with various velocities. Nevertheless, our evaluation gesture samples were collected without enforcing any definition of gestures to our participants. The high accuracy of uWave indicates that its premise is close to how users perceive gestures and how users perform gestures.

6.2. Challenge of Tilt

On the other hand, uWave relies on a single three-axis accelerometer to infer the force applied. However, *the reading of the accelerometer does not directly reflect the external force*, because the accelerometer can be tilted around three axes. The same external force may produce different accelerations along the three axes of the accelerometer if it is tilted differently; likewise, the different forces may also produce the same accelerometer readings. Only if the tilt is known, the force can be inferred from the accelerometer readings.

The opportunity for detecting the tilt during hand movement is very limited with a single accelerometer. We attempted to address it by allowing each pair of matching points on the DTW grid (See Figure 2) to calculate the distance based on tilts of small angles. While it helped with matching samples of the same gesture collected with different tilts, it also increased the confusion between certain gestures, largely due to the confusion between gravity and the external force. To fully address tilt variation, it requires more sensors to provide more information, e.g. compass and gyroscope.

6.3. User-Dependent vs. User Independent Recognition

This work and numerous others are targeted at user-dependent gesture recognition only. The reasons are multiple. First, user-independent gesture recognition is difficult. Our database shows great variations among participants even for the same predefined gesture. For example, if we treat all the samples in the database as from the same participant and repeat our bootstrapping test procedure, the accuracy will decrease to 75.4% compared with 98.4% for user-dependent recognition. To improve the accuracy of user-independent recognition, a large set of training samples and a statistical method are necessary. More importantly, research is required to identify the common "features" from the acceleration data for the same gesture. In speech recognition, MFCC and LPCC have been found to capture the identity of speech very effectively. Unfortunately, we do not know their counterparts for acceleration-based gesture recognition. Second, user-independent gesture recognition may not be as attractive as speaker-independent speech recognition because there is no standard or commonly accepted gestures for interaction. Commonly recognized gestures by humans are often simple, such as those in the Nokia vocabulary. As they are short and simple, however, they can be easily confused with each other, in particular with the presence of tilt and user variations. On the other hand, for personalized gestures composed by users, it is almost impossible to collect a large dataset for statistical methods to be effective.

6.4. Gesture Vocabulary Selection

The confusion matrixes presented in Figure 5 and Figure 4 highlight the importance of selecting the right gesture vocabulary for higher accuracy. As from Figure 5, we can see that uWave often confuses Gesture 1 with Gesture 7. The reason is that tilt of the handheld device can transform different forces into similar accelerometer readings. Unlike speech recognition where the selection of words is constrained by the language, gesture recognition has more flexible inputs, because the user can compose gestures without the constraint of a "language". More complicated gestures may lead to higher accuracy because they are likely to have more features that distinguish them from each other, in particular, offsetting the effect of tilt and gravity. Nevertheless, complicated gestures pose a burden to human users: the user has to remember how to perform complicated gestures in a consistent manner and associate them with some unrelated functionality. Eventually, the number of complicated gestures a user can comfortably command may be quite small. This may limit gesture-based interaction to a relatively small vocabulary, for which uWave indeed excels.

6.5. Further Algorithmic Improvement

It is important to note that the objective of this work is to demonstrate the effectiveness of uWave as a combination of quantization, DTW, and template adaptation in personalized gesture recognition. While uWave achieves very competitive accuracy without user perceptible latency, we stop short of applying more advanced DTW and template adaptation algorithms to uWave. However, we are aware of numerous existing solutions that may be exploited. For recognition accuracy, extensive researches have explored more complicated template adaptation methods [20, 9]. For computation efficiency, advanced techniques have already been proposed to compute DTW in linear time and space [15].

7 Conclusions

We present uWave for personalized gesture-based interaction. uWave employs a single accelerometer so that can be readily implemented on many commercially available consumer electronics and mobile devices. The core of uWave includes 1) dynamic time warping (DTW) to measure similarities between two time series of accelerometer readings; 2) quantization for reducing computation load and suppressing noise and non-intrinsic variations in gesture performance; and 3) template adaptation for coping with gesture variation over the time. Its simplicity and efficiency allow implementation on a wide range of devices, including simple 16-bit microcontrollers, as long as an accelerometer is available.

We evaluate the application of uWave to user-dependent recognition of predefined gestures with over 4000 samples collected from eight users over multiple weeks. Our experiments demonstrate that uWave achieves 98.6% accuracy starting with only one training sample. This is comparable to the reported accuracy by HMM-based methods [6] with 12 training samples (98.9%). We show that the quantization improves recognition accuracy and reduces the computation load. Our evaluation also highlights the challenge of variations over the time to user-dependent gesture recognition and the challenge of variations across users to user-independent gesture recognition.

We believe uWave is a first major step toward building technology that facilitates personalized gesture recognition. Its accurate recognition with one training sample is critical to the adoption of personalized gesture recognition in a range of devices and platforms. It has the potential to enable novel gesture-based navigation and operation of next generation user interfaces.

Acknowledgments The work is supported in part by NSF awards CNS/CSR-EHS 0720825 and IIS/HCC 0713249 and by a gift from Motorola Labs. The authors would like to thank the partic ipants in our user studies who remain anonymous.

References

[1] Analog Device, Small, Low Power, 3-Axis ±3g i MEMS® Accelerometer, ADXL330 datasheet, 2006.

[2] Baudel, T. and Beaudouin-Lafon, M. Charade: remote control of objects using free-hand gestures. *ACM Communication,* 36, 7, 28-35, Jul. 1993

[3] Cao, X. and Balakrishnan, R. VisionWand: interaction techniques for large displays using a passive wand tracked in 3D. In *Proc. 16th Annual ACM Symp. User Interface Software and Technology,* November 2003.

[4] Chernick, Michael R. Bootstrap Methods, A practitioner's guide. *Wiley Series in Probability and Statistics,* 1999.

[5] Hofmann, F. G., Heyer, P., and Hommel, G. Velocity Profile Based Recognition of Dynamic Gestures with Discrete Hidden Marko[v Models. In *Proc. Int. Gesture Wrkshp. Gesture and Sign Language in Human-Computer Interaction,* September 1997.

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

[7] Mäntyjärvi, J., Kela, J., Korpipää, P., and Kallio, S. Enabling fast and effortless customisation in accelerometer based gesture interaction. In *Proc. 3rd Int. Conf. Mobile and Ubiquitous Multimedia,* October 2004.

[8] Jang, I. J. and W. B. Park. Signal processing of the accelerometer for gesture awareness on handheld devices.

Proc. 12th IEEE Int. Wrkshp Robot and Human Interactive Communication, 2003.

[9] McInnes, F.R., Jack, M.A., and Laver, J. Template adaptation in an isolated word-recognition system. In *IEE Proceedings*, Vol. 136, Pt. I, No.2, April 1989.

[10] Myers, C. S. and Rabiner, L. R.. A comparative study of several dynamic time-warping algorithms for connected word recognition. *The Bell System Technical Journal*, 60(7):1389-1409, September 1981.

[11] Nintendo Wii, http://www.nintendo.com/wii/.

[12] Rice Orbit Sensor Platform,http://www.recg.org/orbit/

[13] Perng, J.K., Fisher, B., Hollar, S., and Pister, K.S.J. Acceleration sensing glove (ASG). In *Proc. Int. Symp. Wearable Computers*, 178 – 180, 18-19 October 1999.

[14] Rabiner, L. R. and Juang, B. H., An Introduction to Hidden Markov Models. In *IEEE ASSP Magazine*, pp. 4-15, January 1986.

[15] Salvador, S. and Chan, P. FastDTW: Toward accurate dynamic time warping in linear time and space. In *Proc. ACM Wkshp. Mining Temporal and Sequential Data*, August 2004.

[16] Wu, Y. and Huang, T. S. Vision-Based Gesture Recognition: A Review. In *Proc. Int. Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction*, March 1999.

[17] Wisniowski, H. Analog Devices and Nintendo collaboration drives video game innovation with iMEMS motion signal processing technology. Analog Devices, Inc. Retrieved on 2006-05-10.

[18] Wilson, D. and Wilson, A. Gesture Recognition Using XWand, *report CMU-RI-04-57, Robotics Institute, Carnegie Mellon University*, 2004.

[19] Wobbrock, J. O., Wilson, A. D., and Li, Y. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proc. 20th Annual ACM Symp. User Interface Software and Technology*, October 2007.

[20] Zelinski, R. and Class, F. A learning procedure for speaker-dependent word recognition systems based on sequential processing of input tokens. In *Proc. IEEE ICASSP*, 1983.

[21] AiLive LiveMove Pro:

http://www.ailive.net/liveMovePro.html