Proceedings of the 2007 IEEE
International Conference on Mechatronics and Automation
August 5 - 8, 2007, Harbin, China

Real-time Recognition of Multi-category Human Motion Using μIMU Data*

Weiwei Wan, Hong Liu
State Key Laboratory of Machine Perception
Peking University
Shenzhen Graduate School, China
wanweiwei@cis.pku.edu.cn,hongliu@pku.edu.cn

Guangyi Shi, Wen J. Li
Centre for Micro and Nano Systems
The Chinese University of Hong Kong
Hong Kong, China
{gyshi,wen}@mae.cuhk.edu.hk

***Abstract*— This paper describes a novel approach for human motion recognition via motion feature vectors collected from a micro Inertial Measurement Unit (μIMU), which measures angular rates and accelerations of the three different directions in the workspace based on MEMS sensors. The recognizer is composed of three parts. The first part is a preprocessor, in which Vector Quantization is used to reduce dimensions of vectors. Recognition is implemented by the second part, which is a classifier composed of Hidden Markov Model and an efficient second layer criterion. The third part uses a sliding window algorithm for precise recognition. There were 200 sequences (about 100,000 vectors) for 10 different kinds of motions tested in our work, including falling-down motion and other typical human motions. Experimental results show that for the given 10 different categories, correct recognition rates range from 95%-100%, of which the falling-down motion can be classified from others with a 100% recognition rate.**

***Index Terms*— μIMU Data, Human Motion Recognition, Vector Quantization, Hidden Markov Model**

I. INTRODUCTION

Hip fracture can result in significant psychological trauma and lead to self-imposed restrictions of activity that can compromise the life quality of the individual. A lot of hip protectors are made to reduce the incidence of hip fractures. However, applications are limited, due to discomfort or wearing difficulties. A novel and more comfortable hip-air-bag is under development by Wen J.Li's group at the Chinese University of Hong Kong[1][2]. It is made of a motion sensing unit (μIMU) which is used to detect body imbalance and trigger the inflation of compact airbags worn by the elderly. There are 6 elements in each vector from the μIMU, which are presented as below. Elements 1,2 and 3 are angular rates of X, Y and Z directions in the workspace (*AX*, *AY* and *AZ*). Axis of X is from left hip to right hip, axis Y is from back to front and axis Z is vertical to the ground. Elements 4, 5 and 6 are acceleration of X, Y and Z (*GX*, *GY* and *GZ*).

Many complicated pattern recognition problems involve identification of stationary and time-varying sequences. Handwriting recognition, speech recognition and sign language recognition are classsical applications of time-varying sequence recognition. The experiments of references[1][2] take motion vectors as stationary signals. Principle Component Analysis (PCA) and Support Vector Machine (SVM) are adopted in the experiment, and results show that 200 experimental data sets can be separated into falling-down and other motions accurately. SVM is a realization of the Vapnik-Chervonenkis (VC) dimension theory, which performs great when training samples are few. Their work runs well in only 350 *ms* with a 100% recognition rate for binary classification (*fall* and *other*).

Although the *fall* motion can be recognized correctly, we are exploring new methods for the following reasons:

(1) SVM is not fit for time-varying sequence recognition. Sequences collected from μIMU sensors vary along time, and each motion happens in a few hundred microseconds.

(2) With SVM, if no motion is detected in a sequence, it will be improperly categorized.

(3) SVM is feasible for binary classification; However, for multi-class classification, classifiers are complicated.

(4) During experiments, sufficient training samples can be collected, so limitation of samples can be ignored.

Considering these factors, we explored a new method.

Traditional approaches to solve time-varying problems are Dynamic Time Warping (DTW) and Hidden Markov Model (HMM). Signals scale along time or in amplitude (DDTW[3]) can be classified easily by DTW or its derivations, but rigorous templates are required. HMM also enjoys obvious advantages in recognition of time-varying sequence[4], such as speech recognition[5] and sign language recognition. A multi-layer classifier based on HMM and SVM to recognize sign language is introduced in [6]. The performance of DTW and HMM is compared in [7], with the application of a gesture recognition system, which shows that performances of DTW are worse compared to HMM. So Discrete Hidden Markov Model (DHMM)[5] is introduced in this paper for time efficiency. Although proved a plain model, with the help of the second layer criterion, a satisfying recognition rate is gained .

*For questions regarding recognition algorithms, please contact: hongliu@pku.edu.cn. For questions regarding uIMU, please contact: wen@mae.cuhk.edu.hk

Information from all the six dimensions are necessary for multi-class recognition, though fewer dimensions may be sufficient for binary classification. Dimensional reduction is necessary for time efficiency. Although traditional PCA-based technology is widely used, for the given sequences of 10 different motions affected by different dimensions, features are hard to extract and dimensions are hard to reduce. In order to distinguish more motions, Vector Quantization (VQ)[8][9] is introduced. When sorted, indices of quantized vectors are used to train HMMs, and different motions can be categorized with little error by a sliding window method.

This paper is organized as follows. In Section II , concepts of VQ and HMM are introduced. Section III will focus on the improvement of this paper. Section IV shows the experimental results and conclusions are drawn in Section V.

II. Vector Quantization and Hidden Markov Model

Both VQ and DHMM are introduced in this paper. Basic concepts of VQ and HMM are as follows.

A. Vector Quantization

VQ is a lossy data compression method based on the principle of block coding. It is a fixed-to-fixed length algorithm. The VQ design problem can be stated as follows. Given a vector source with its statistical properties known, a distortion measure, and the number of codevectors, find a codebook and a partition which will result in the least average distortion.

In 1980, Linde, Buzo, and Gray (LBG) proposed a VQ design algorithm based on a training sequence. The use of a training sequence bypasses the need for multi-dimensional integration. VQ designed using this algorithm is referred to in literatures as an LBG-VQ. LBG is used in this paper for VQ.

B. Hidden Markov Model

A Hidden Markov Model derives from Markov Chains, which is a doubly embedded stochastic process with an underlying stochastic process that is not observable, and can only be observed through another set of stochastic processes that produce the sequence of observations. An N-state ($(s_1, s_2, ..., s_N)$) HMM can be described as $\lambda = (\pi, A, B)$. The parameters are as follows.

The initial state distribution,

$$\pi = (\pi_1, \pi_2, ..., \pi_N), \tag{1}$$

which is used to describe the distribution of the state at time $t = 0$, and $\sum_{i=1}^{N} \pi_i = 1$.

The state transition probability distribution can be given as,

$$A = \{a_{ij}\}_{N\times N}, i, j = 1, 2, ..., N, \tag{2}$$

where $a_{ij} = P(q_j, t+1|q_i, t), 1 \le i, j \le N$, and $\sum_{j=1}^{N} a_{ij} = 1$.

In the special case where any state can reach any other in a single step, $a_{ij} > 0$ for any i, j. For other types of HMMs, $a_{ij} = 0$ for one or more (i, j) pairs.

The observation symbol probability distribution in state j can be described as $B = b_j(k)$, where,

$$B = \{b_{jk}\}_{N\times M}, k = 1, 2, ..., M, \tag{3}$$

N denotes the hidden states, and M denotes the length of the codebook (discrete observation density). For continuous observation densities, it is $B = b_i(x)$.

Associating with the distribution of B, HMMs are categorized into Discrete Hidden Markov Model (DHMM, with discrete probability density) and Continuous Hidden Markov Model (CHMM, with continuous probability density).

Using these parameters, testing vectors can be interpreted properly. A testing vector is classified into the Model with the highest Maximum A Posteriori. To fix the idea, refer to Section IV.

III. Recognizer for Multiple Categories

A. Two-layer classifier based on VQ and HMM

1) Splitting method based VQ: Vectors can be projected to a low dimensional space with PCA, which can be recognized easily. However, only data with single dimension can be used to train DHMM. PCA and CHMM are better partners[13]. Although PCA+CHMM technology has higher recognition rate, DHMM is in place of CHMM and PCA is replaced by VQ to finish classification in less than 350 *ms*. In order to be an accurate classifier, a second layer criterion is defined, which has little effect on time efficiency.

Information from all the dimensions of motion feature vectors are necessary for multi-class recognition, though fewer dimensions will be sufficient to binary classification. To convert a vector into a single dimensional parameter using VQ can both hold all the information and save a lot of time.

64 vectors are generated after Vector Quantization. There are six elements in each vector, namely x_1, x_2, x_3, x_4, x_5 and x_6. Changes of indices to these vectors can be observed easily after sorting by the sum of all the six dimensions.

The key problem of VQ is to choose the initial codebook. Splitting method helps to sample and distribute features into the quantized vectors. By using splitting method, the system will be extensible and more kinds of motions can be introduced easily. For $i = 1, 2, ..., N$, set

$$C_i^{(0)} = (1 + \epsilon)C_i^* \tag{4a}$$

$$C_i^{(0)} = (1 - \epsilon)C_i^* \tag{4b}$$

C_1 denotes the initial centroid, it splits into two with a small ϵ each time. With a longer codebook, there will be more noises, while with a shorter codebook, there will be too much loss. 64 vectors are generated after 6 splits for better performance.

2) HMMs based on indices: Indices to sorted codevectors are used to train HMMs. Similar to speech recognition with paragraphs, initial point and end point must be found, namely endpoint detection. In speech recognition, this problem can be solved using breaks between words, energetic or frequent differences. However, the same methods are not fit to classification of motions, a man who is climbing the stairs may fall. A sliding window method is introduced to avoid these confusion, and the result shows high accuracy rate.

Baum-Welch algorithm[5] is used to train HMMs for different classes, while forward and backward algorithm is used to evaluate each window of the testing sequences.

3) Second layer criteria: DHMM and VQ were introduced together since the 1980s, which perform much worse compared with CHMM+PCA[5] except for time efficiency. In pattern recognition, none of the approaches is perfect, so multi-layer classifiers are explored[4]. Obviously, complicated second-layer criteria will occupy plenty of time.

References[6][7][10]-[12] are complicated multi-layer classifiers, however they are not fit to this recognition. A simple criterion is generated by experience, with the help of which ambiguous results from DHMM are finally separated.

Criteria for the second layer is, the difference between the log-likelihood of two succeeding 50-frames belonging to the same motion is larger than -300, and 2000 higher than any other motions.

B. Recognition algorithm

1) Training algorithm: Given training samples, Hidden Markov chains are generated as below.

Step 1-1. Samples are used to generate 64-codebook C with LBG-algorithm.

Step 1-2. Each sample is quantized according to C, and the results are sequences with index numbers pointing to items in C.

Step 1-3. $S_0, S_1, S_2, ...$ are introduced to denote these sequences. 50-frames are cut from each S_i with a tagged starting point.

Step 1-4. Train the corresponding Hidden Makov chains with these 50-frames until all the 10 models are obtained.

Step 1-5. End.

2) Recognition algorithm: Given testing vectors, recognition is done in the following steps.

Step 2-1. A sequence which is a series of vectors(m-length) with 6 dimensions is preprocessed. For a given sliding window ω, a ω-length sequence will be cut from the original n-length one.

Step 2-2. The ω-length sequence will be input into the first part of the recognizer for Quantization. Each 6-dimensional vector will be converted into a single dimensional index to a vector in Codebook C , and a ω-length single dimensional vector will be generated.

Step 2-3. Output of Step 2-2 is judged by the two-layer classifier to find out if any motion appears during this period.

Step 2-4. Slide the window with one frame and start again from Step 2-1 until all the vectors are tested and motions will be recognized.

Step 2-5. Try other sequences and repeat Step 2-1 to 2-4, until all the sequences are tested.

Step 2-6. End.

IV. Experiments

Experiments are carried out following the previous algorithms. They are introduced in this section, and results are compared with [5] and [7]. MatLab is used to simulate the device, which is performed on a Pentium M 1.5GHz PC with 512MB memory.

Ten different classes of motions are defined in this paper with number of experimental samples listed in the bracket, fall (*fall*, 100), squat (*squat*, 10), sitting down to a chair (*scsd*, 10), climbing upstairs (*upstairs*, 10), stand up from a chair (*scsu*, 10), going downstairs (*downstairs*, 10), walk (*walk*, 20), jump (*jump*, 10), run (*run*, 10) and stand up from squatting (*susq*, 10). There are 200 vectors total.

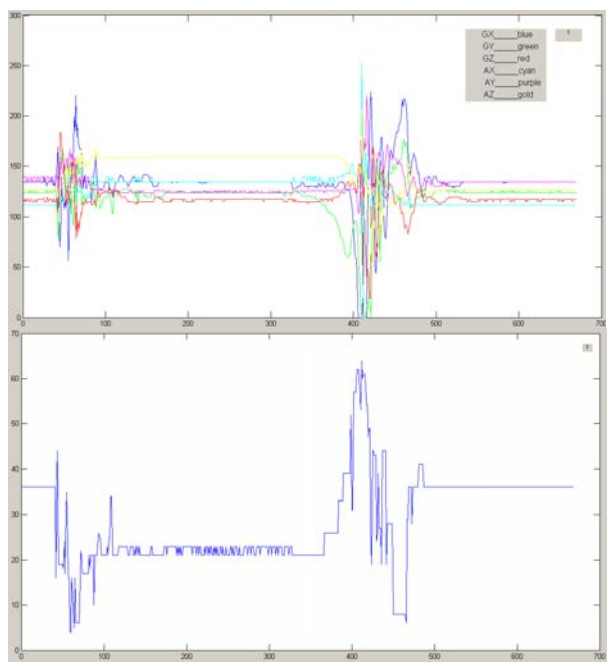
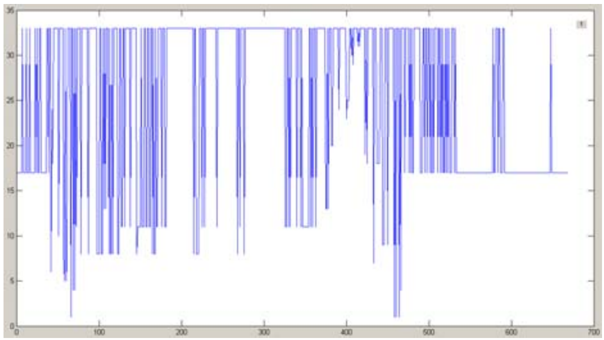


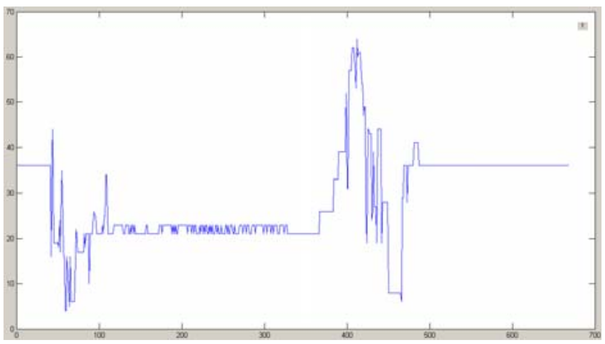
Fig. 1. Original data and 64 quantized indices

A. Codebook construction

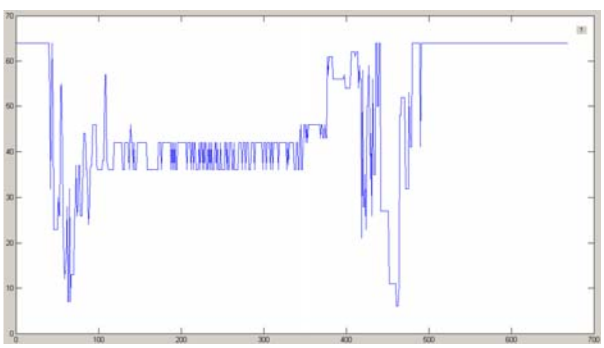
As introduced in Section II, with a longer codebook, there will be more noises, while with a shorter codebook, there will be too much loss. Changes in original 6-dimensional sequence (*AX, AY, AZ, GX, GY, GZ*) and the corresponding quantized single dimensional sequences of the first falling-down motion are shown in Fig.1. In Fig.2 performances of 32, 64 and 128 codebook quantization are compared and 64 does the best.



(a) 32 vector quantized indices



(b) 64 vector quantized indices



(c) 64 vector quantized indices

Fig. 2. Performance of quantization by codebooks of different length

B. Endpoint detection and training of samples

A lot of samples are needed to train HMMs. However, the sequences collected will not be exactly what happens in the first 350 *ms* of a falling motion, therefore, preprocessing is necessary. The first 50 frames from the falling point are used. Although 50 frames take much longer than 350 *ms*, falling-down motion can be detected long before the subject hits the ground. Since a 64-codebook is chosen, falling points can be tagged easily, as shown in Fig 3. All these 50 falling frames will generate the HMMs. Same operations are carried out on the other motions.

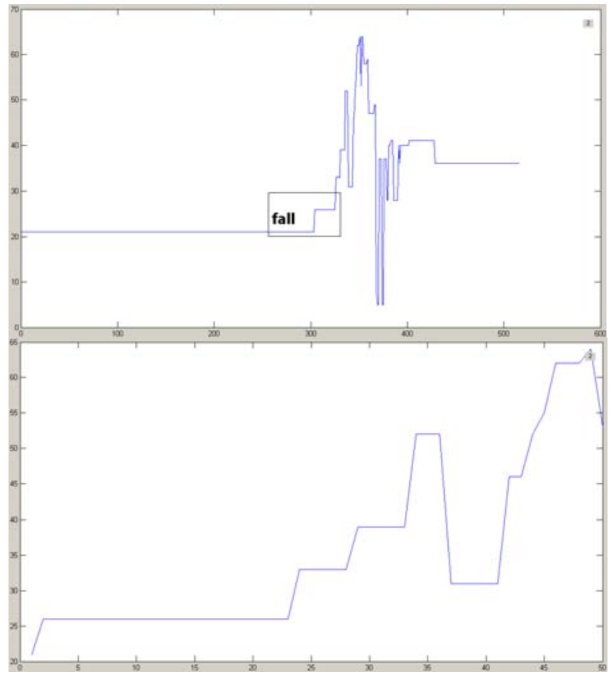


Fig. 3. Tag the 50 frames from falling point

C. Generation of Hidden Markov chains

When calculating probability may be so small that current precise of computers will not do, so logarithms of probabilities are used. Iteration stops when difference between the succeeding two iterations is less than 0.01. Observation states is set to 64, the same as codevectors. Log-likelihood is shown in table I with 10, 20, 30, 40, 50 and 60 hidden states.

TABLE I

LOG-LIKELIHOOD OF DIFFERENT HIDDEN STATE

HS	fall	squat	scsd	upstairs	scsu
10	-5798.979	-435.971	-380.899	-528.393	-249.736
20	-4460.617	-367.930	-285.251	-423.143	-189.953
30	-4448.483	-345.784	-308.808	-385.415	-170.034
40	-3996.614	-284.574	-236.182	-329.617	-188.125
50	-3667.044	-264.364	-216.028	-298.115	-145.665
60	-3651.717	-244.564	-218.904	-273.772	-137.147
HS	downstairs	walk	jump	run	susq
10	-599.117	-1486.262	-550.554	-725.790	-486.901
20	-479.395	-957.501	-436.151	-557.489	-399.576
30	-449.137	-757.011	-408.469	-456.284	-359.535
40	-391.583	-770.074	-377.348	-431.772	-351.780
50	-364.322	-678.077	-311.128	-401.077	-303.748
60	-319.431	-624.072	-285.618	-345.632	-286.788

As shown in Table I, there is little improvement between 50 and 60. Also, it takes a longer time as the hidden states increase. 50 is chosen as the hidden states, and ten different hidden markov chains are generated, λ_{fall}, λ_{squat}, λ_{scsd}, $\lambda_{upstairs}$, λ_{scsu}, $\lambda_{downstairs}$, λ_{walk}, λ_{jump}, λ_{run} and λ_{susq}.

D. Sliding window-based real-time recognition

In order to perform real-time detection and diminish misjudgement, a sliding window method is taken. Starting from the 50^{th} frame, the previous 50 frames are detected, that is frame 1-50, 2-51, ..., n-50+n, ...

As shown in Fig.4, although 50 frames take a much longer time than 350 *ms*, falling-down motion can be detected at the 342^{th} frame (Far from the trough of AZ). For detection of each 50 frame, only 120 *ms* is taken by using MatLab. With a proper criterion, all the 200 sequence are tested, and the results are shown in Table II.

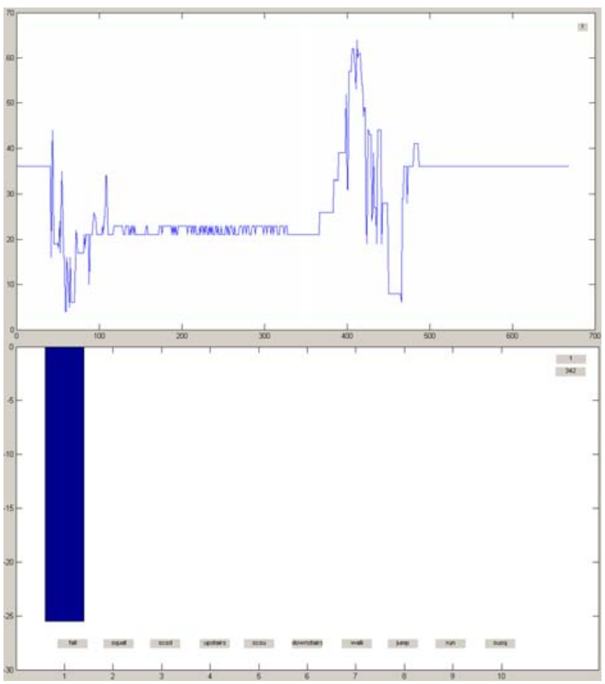


Fig. 4. Falling motion detected

TABLE II

THE RESULT OF OUR EXPERIMENT

class	correct	error	undetected	performance
fall	200	0	0	100%
squat	198	0	2	99%
scsd	193	0	7	96.5%
upstairs	196	2	2	98%
scsu	196	1	3	98%
downstairs	197	1	2	98.5%
walk	199	1	0	99.5%
jump	190	1	9	95%
run	199	0	1	99.5%
susq	192	1	7	96%

Performances are observed independently. All models $\lambda_{fall}, \lambda_{squat}, ..., \lambda_{susq}$ are introduced to compose the first layer. Only one category is tried each time to recognize all the 200 sequences. Category *fall* only happens in 100 of the 200 ones . For category λ_{squat}, 190 of them do not include the motion. Here is the equation to calculate the performance.

$$P_{performance} = \frac{200 - W_{C_{motion}}}{200}, \tag{5}$$

in which $W_{C_{motion}}$ is the error and undetected number of sequences judged by the corresponding Category C_{motion}.

The ”correct” column in Table II shows that corresponding motions are detected correctly, the ”error” column denotes misjudgement when a motion happens, and the ”undetected” column shows that motions are not detected although it occured. The result is satisfying, although only 10 training samples were used, such as the *squat*, *upstairs*, *downstairs* and *run* models. Table III shows the performance of the XWand system[7]. A two-layer classifier is far better than basic HMM, with a falling-down recognition rate 100% which is the same as SVM-based methods in[1][2].

TABLE III

PERFORMANCE OF DIFFERENT TECHNOLOGY IN [7]

Algorithm	Accuracy	Correct/Total
Linear Time Warping	40.42%	17/42
Dynamic Time Warping	71.64%	30/42
Hidden Markov Model	90.43%	38/42

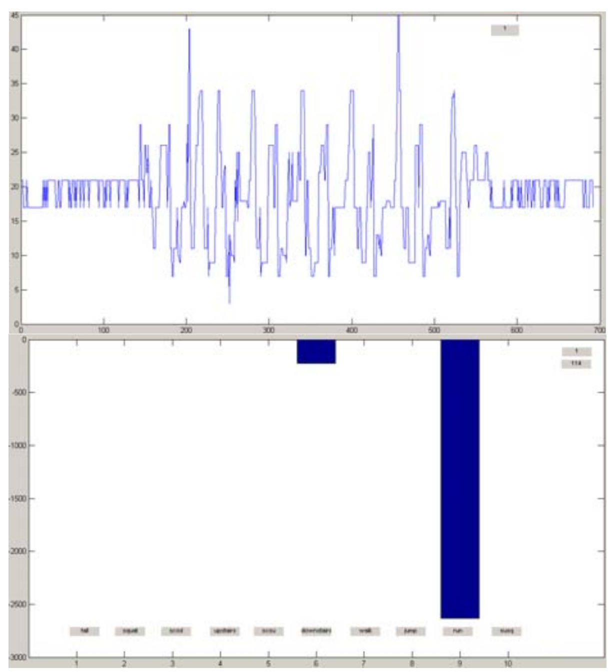


Fig. 5. Detection of downstairs

Only 4 of the 200 samples, are misjudged. The 10^{th} sample of *upstairs* is classified as *downstairs*, the 4^{th} sample of *jump* is classified as *susq*, the 5^{th} sample of *susq* is classified as *scsu*, and the 11^{th} of *walk* is recognized as *upstairs*. When trying to recognize the *jump* motion, the worst performance is reached. Although none of the other 190 sequences are misjudged, unfortunately, 10 sequences are not categorized correctly. Such bad performance also happened to *susq*. 192 sequences of which are not misjudged, whereas the other 8 error or undetected.

The models of *scsd*, *jump* and *susq* did not perform well, hence, another experiment is carried out to find the reason. We have fount that it is because of the second-layer criterion, which is defined as the difference between the log-likelihood of two succeeding 50 frames with the class of a motion is larger than -300, and 2000 higher than any other motions. The more rigorous the criterion is, the less misjudgement there will be. With a less rigorous criterion, undetectable samples can be diminished. Both the two aspects are considered when designing the criterion. Classification based on Maximum A Posteriori, including HMM, needs a lot of training samples, as shown in the experiment. With more training samples, precise classification of all the motions can be made. Fig.5 shows the detection of downstairs, and Fig.6 shows the robustness of the algorithm, though a lot of noises, falling point is detected at frame 235.

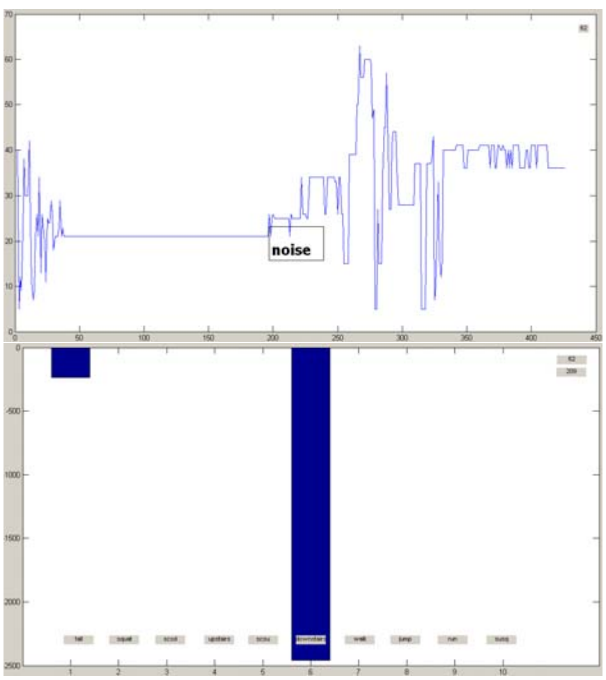


Fig. 6. Robustness of the algorithm

V. Conclusions

This paper presents a novel method to recognize different human motion sequences collected by μIMU. Dimensions are reduced by Vector Quantization, and Hidden Markov chains are trained by indices of quantized vectors. To reach a higher recognition rate, a second layer criterion is presented. Although the criterion in the second-layer is empirical, both high time efficiency and accuracy rate are ensured. Experimental results show that the method proposed is precise as well as efficient. For the given 200 sequences of 10 different kinds of motions, fallinng-down can be recognized from all the other motions without any error, i.e., a 100% correct recognition rate is reached. For the other 9 motions with fewer samples, correct recognition rates are from 95%-99%.

Acknowledgements

This project is funded by the Hong Kong Innovation and Technology Commission (Project ITF-GHP-029-06), supported by National Science Foundation of China (NSFC No.60675025) and National Natural High Technology Research and Development Program of China (863 Program, No.2006AA04Z247).

References

[1] Yilun Luo, Guangyi Shi, Josh Lam, Guanglie Zhang, Wen J.Li, Philip H.W. Leuong, Pauline P.Y.Lui and Kwok-Sui Leung, *Towards a Human Airbag System Using μIMU with SVM Training for Falling-Motion Recognition*, in Proceedings of International Conference on Robotics and Biomimetics IEEE, pp 634-639, 2005

[2] Guangyi Shi, Cheung-Shing Chan, Yilun Luo, Guanglie Zhang, Wen J.Li, Philip H.W.Leong and Kwok-Sui Leung, *Development of a Human Airbag System for Falling Protection Using MEMS Motion Sensing Technology*, in Proceedings of Intelligent Robot and System 2006 IEEE ($IROS'$06), pp 4405-4410, 2006

[3] Eamonn J. Keogh, Michael J.Pazzeni, *Derivative Dynamic Time Warping*, in First SIAM International Conference on Data Mining (SDM'01), 2001

[4] Chard. O. Duda, *Pattern Classification, Second Edition*, John Wiley, 2003

[5] Lawrence R. Rabiner, *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, in Proceedings of the IEEE, pp 257-286, Vol.77, No.2, 1989

[6] Jianjun Ye, Hongxun Yao, Feng Jiang, *Based on HMM and SVM Multilayer Architecture Classifier for Chinese Sign Language Recognition with Large Vocabulary*, in Proceedings of the Third International Conference on Image and Graphics ($ICIG'$04), pp 377-380, 2004

[7] Daniel Wilson, Andy Wilson, *Gesture Recognition Using XWand*, report CMU-RI-TR-04-57, Robotics Institute, Carnegie Mellon University, 2004

[8] R. M. Gray, *Vector Quantization*, IEEE ASSP Magazine, pp 4-29, 1984

[9] Y. Linde, A. Buzo, and R. M. Gray, *An Algorithm for Vector Quantizer Design*, IEEE Transactions on Communications, pp 84-95, Vol.COM-28, 1980

[10] Aravind Ganapathiraju, Joseph Picone, *Hybrid SVM/HMM architecture for speech recognition*, in Proceedings of the International Conference on Spoken Language Process, pp 504-507, 2000

[11] Yasemin Altun, Ioannis Tsochantaridis, Thomas Hofmann, *Hidden Markov Support Vector Machine*, in Proceedings of the Twentieth International Conference on Machine Learning ($ICML'$03), pp 3-10, 2003

[12] Hyekyung Lee, Seungjin Choi, *PCA+HMM+SVM for egg pattern recognition*, in Proceedings of the Seventh International Symposium on Signal Processing and Its Applications, pp 541-544, Vol.1, 2003

[13] Shaoyuan Zhou, Shuqing Wang, Jianming Zhang, *Research on integrated PCA-CHMM based framework for fault diagnosis in chemical process*, Journal of Zhejiang University (Engineering Science), pp 1475-1480, Vol.39, No.10, 2005