Proceedings of the 2008 IEEE
International Conference on Robotics and Biomimetics
Bangkok, Thailand, February 21 - 26, 2009

Real-Time Written-Character Recognition Using MEMS Motion Sensors: Calibration and Experimental Results

Zhuxin Dong and Uchechukwu C. Wejinya
Department of Mechanical Engineering
University of Arkansas
Fayetteville, Arkansas 72701, USA
uwejinya@uark.edu

Shengli Zhou, Qing Shan, Wen J. Li
Centre for Micro and Nano Systems
Department of Mechanical & Automation Eng.
The Chinese University of Hong Kong, HKSAR
wen@mae.cuhk.edu.hk

Abstract **– A Micro Inertial Measurement Unit (μIMU) based on Micro Electro Mechanical Systems (MEMS) sensor is applied to sense the motion information produced by human subjects. The μIMU is built with three-dimensional accelerometer. During our experiments, although we write the characters in a plane, all the three-dimensional acceleration information is taken from μIMU in processing data as the third dimension is very helpful to be applied in practical application. In our previous work, the effectiveness of different data processing methods including Fast Fourier Transform (FFT) and Discrete Cosine Transform (DCT) are compared, thus the latter, which is better, is adopted in this paper. Also, Hidden Markov Models (HMM) is introduced and used as a tool to realize hand gesture classification. With this method, new experimental results of hand-written recognition are obtained and stated in this paper. Five Arabic numbers, 0-4, are written forty times by two different persons and we utilize all the data as training samples. Then when another new 29 samples are input to the network for recognition, we obtain a high correct rate at 93%. Ultimately, this technology will provide the feasibility of character recognition and potential for human-gesture recognition.**

Index Terms – MEMS; μIMU; Accelerometer; DCT; HMM.

I. INTRODUCTION

With the advent of computing technologies, specifically in the areas of improved computational speed and reduced system cost with additional functionalities, real-time and virtual interactions between humans and computers have gained significant research and commercial interests recently. Our work discussed in this paper focuses on human interactions with computing devices using characters and gesture recognition. There are two primary character recognition methods based on different inputs: one is Optical Character Recognition (OCR), which gets data information by scanning the printed text; the other is Dynamic Character Recognition (DCR), which recognizes the characters based on their motion information, such as acceleration, angular velocity and so on. Since DCR method can realize interactions between a human and a computer much more effectively than OCR method, and together with the rapid development of advanced micro sensor technology, many researchers have been exploring the possibility of realizing DCR using MEMS motion sensors [1-7].

In [8], acceleration information was obtained from the input device, then integrated twice to obtain the position information. Also, the characters could theoretically be recognized directly base on accelerations and angular velocities. However, due to various noise signals, which may include significant intrinsic drift of the sensor, circuit thermal noise, random noise etc. [9], it becomes difficult to reconstruct the characters either directly or by double integration.

For assessment purposes, we wrote some characters and numbers using a MEMS sensing device as input on a transparent plane. At the same time, the sequential images of the motion of the input device were recorded by a high speed camera [9]. Then, we acquired the motion data information of each character in two forms: one is angular velocity and acceleration of the device measured by the MEMS sensor; the other is the motion trajectory of the input device recorded by a high speed camera. Thus, we obtained the trajectory of the input device by combining the sequential pictures and compared it to position trajectory obtained using the MEMS sensor by double-integration of acceleration signal. However, from experiments, the trajectories of written characters obtained by integrating acceleration are unrecognizable owing to sensor's drift and intrinsic errors. This phenomenon told us the MEMS acceleration sensor could not provide enough sensitivity and accuracy to give position trajectories as simple as a single letter or number.

Therefore, another method is explored to recognize characters. We utilize Discrete Cosine Transform (DCT), which has been proved that it is a better method for us than Fast Fourier Transform, to carry on data processing [10]. After data from writing Arabic numbers *0*, *1*, *2*, *3* and *4* are recorded and processed by DCT. Hidden Markov Models (HMM) is used as a tool to realize hand-written recognition. With the training samples, another new 29 testing samples are input into the network and a high correct rate of numbers' recognition at 93% is obtained.

II. RECALL OF PREVIOUS WORK

A. Experimental Setup

Fig.1 illustrates the experimental setup for the previous experiments. The sensor mode is mainly integrated by three-dimensional accelerometer, three-dimensional gyroscope, Bluetooth module and micro processor. Then the motion data can be captured and transmitted to PC wirelessly while a hand-writing motion is activated on the table plane. The captured data, the acceleration value, is used to compute the velocity and position information by integral. Only two-dimensional acceleration information is adopted for integral. Meanwhile, since the table plane is transparent and a high speed camera is set underneath to record the movement of the sensor, another group of position information can be obtained through sequence of frames.

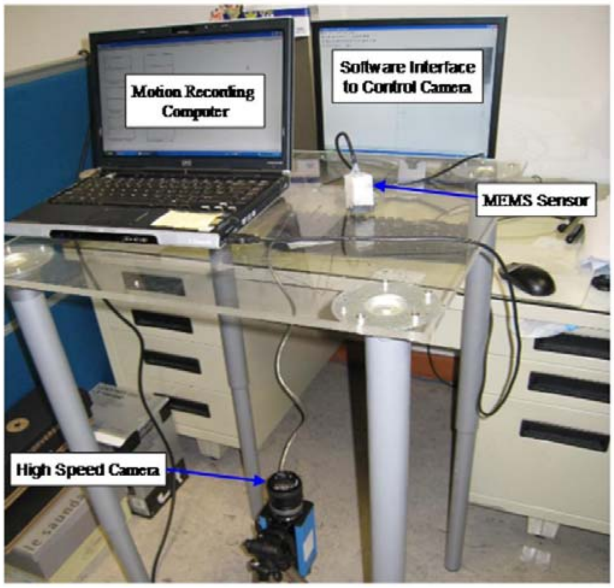


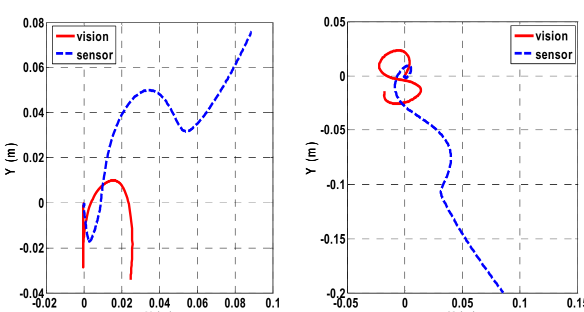
Figure 1. Experimental setup for character recognition

B. Sampling Rate

The frequency of our micro processor is set at 200Hz, so in order to obtain proper reference to compare with, the same sampling rate of the camera is necessary. Our camera has the maximum sampling rate at 1,000Hz [9]. Then the comparison of synchronizing trajectories is able to be done.

C. Track Comparison Result

The trajectory reconstructions of letter *n* and *s* are shown in Fig.2. Apparently, comparing with the curves in red, which is reconstructed by camera, the tracks in blue cannot describe clearly what characters have been written. The blue curves are reconstructed directly from accelerometer output after twice order integral. For this reason, we have developed another



(a) (b)

Figure 2. Comparison between original letters and reconstructed ones: (a) letter *n*; (b) letter *s*

method to recognize characters. Similar experiments have been performed to realize character recognition based on accelerometer [11].

III. Experiment Results

Since the experimental result in the previous experiment proves that reconstruction of trajectory through integration of acceleration is not very desirable for character recognition, we conducted the following experiment to demonstrate the possibility of recognize hand written characters using simply acceleration values.

A. Experimental Setup

Fig.3 shows the new experimental setup and Fig.4 describes the schematics of this work. The Mote sensor sponsored by Virtus Asia Ltd. is used to sense and transmit the acceleration value of to the remote computer. The Bluetooth receiver receives the data from Mote and then the data capture program will store the data. These data will then go through the data auto cut program which will automatically recognize the start and end of a character. Data processing module will then process these cut data and run it through recognition module in order to give out the end result.

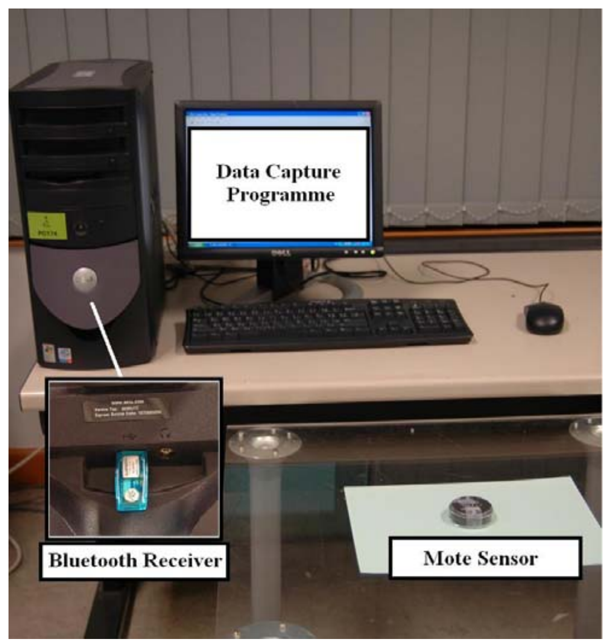


Figure 3. Experimental setup for written-character recognition

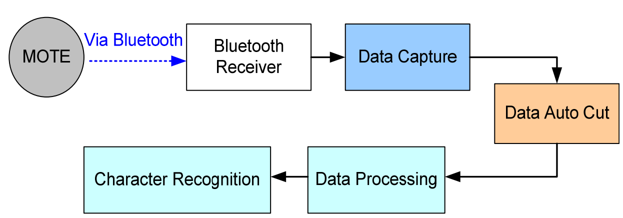


Figure 4. Flowchart of written-character recognition

B. *Algorithm*

a) Auto cut Algorithm

In order to realize real-time character recognition, all the characters should be cut automatically. Table I shows the algorithm of character auto cut program. In this program, we define three variables: flag1, flag2 and flag3 to monitor the state of the matrix at three different moments: T-2, T-1 and T, respectively. For example, at time T, if the matrix belongs to a gesture, flag3 will be given a value 1, otherwise flag3 will be 0. The starting point of data recording is when both flag1 and flag2 equal 1, flag3 equals 1, and the ending signal is when both flag2 and flag3 equal 0, flag1 equal 1. If flag2 equals 1while flag 3 equals 0, the current matrix will be saved in a temporary matrix, and the program will wait for the nest state to decide whether the saved matrix acceptable or not.

In this experiment, since the characters are written on a flat plane, the movement of the input device (i.e. Mote Sensor) is two dimensional. The change of the acceleration in the vertical direction (A_z) will be much smaller than the change of accelerations in the horizon plane (A_x, A_y). According to this idea, the character automatically cut output is shown in Fig. 5. If A_z goes beyond a range, here we define the range is from 330 to 500, no matter how big A_x and A_y are, the system will not seen this gesture as character writing. This is very useful in practical application, especially when we pick the input device up and put it down.

b) Character Recognition Algorithm

i. Data Processing

The data we get directly from the sensor are all in time domain, and the gesture length of each character is different. The accelerometer output is a serial of discrete acceleration values. For reasons of accelerometer structures and random vibrations, there exists some noise in the output [12]. Therefore, some methods must be applied to filter the noise. As we all known, both FFT and DCT can compress the data and filter the noise. In the last paper, we have demonstrated that DCT is better than FFT not only because DCT is better in energy compaction, but also because the data processed by DCT can provide more information than processed by FFT. Thus, in this paper, the data processing method we applied is still discrete cosine transform. The governing mathematical equation for the DCT algorithm is shown in Eq. 1.

It is known that the data in low frequency always represents signal, whereas the high frequency always represents noise. Therefore, after the data processed by DCT, we just take several data in low frequency; the other data are thrown way. In this way, several noise has been got rid of, the data have also been compressed.

$$X_{kdct} = \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi k}{N}\left(n+\frac{1}{2}\right)\right) \qquad (1)$$

ii. Introduction to Hidden Markov Models

TABLE I

GESTURE AUTOMATICALLY CUT ALGORITHM

	Flag1	Flag2	Flag3	Program action
States	0	0	0	No action
	0	0	1	Beginning, recording
	0	1	0	Waiting and saving current matrix
	0	1	1	Recording
	1	0	0	Ending. Determining whether it is acceptable by calculating its length
	1	0	1	Recording the saved and current matrix
	1	1	0	Waiting and saving current matrix
	1	1	1	Recording

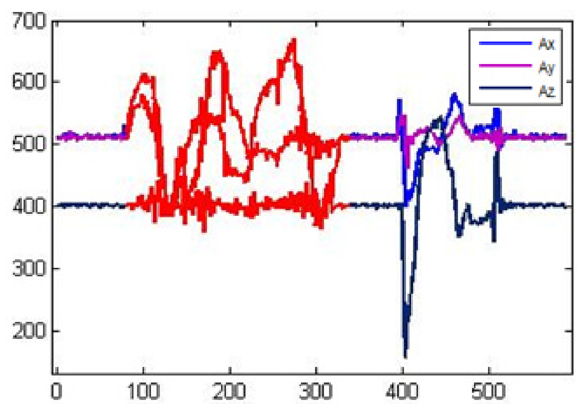


Figure 5. Character automatically cut output of number *'4'*

The classifier we applied in this experiment is Hidden Markov Model. It is a rich tool used for hand gesture recognition in diverse application domains [13]. HMM process is a double stochastic process governed by an underlying Markov chain with finite number of states, each of which is associated with a probability distribution. It includes the initial parameter π, a transition matrix A and an observation matrix B, being denoted as $\lambda = (A, B, \pi)$ [14]. In hidden Markov model, the state is not directly visible, but variables influenced by the states are visible. In the testing process, the network will first check the dimension of the new input, and then calculate the density value of the input. After comparing with the density values in each state, a probability distribution will be generated. In the final stage, the system will give an output according to the probability distribution.

C. *Results*

In our experiments, there are only 5 Arabic numbers are written, which are *0, 1, 2, 3* and *4*. The accelerations of these five Arabic numbers are shown in Fig. 6. Due to the two dimensional movement of the sensor, only two accelerations in each movements are plotted. From Fig. 6, we can find, the movements are totally different except number *2* and number *3*. This is possible! Because the way we write number *2* is similar as the way we write number *3.* Therefore, there may be some problems in the recognition of these two numbers.

As mentioned above, the sampling frequency of the sensor is 200 Hz. In the training data collection stage, we collected data from two different persons. Both of these two persons wrote the characters 20 times, so the total training samples of each character were 40. After trained by these data, we gave the network 29 new inputs for recognition. We were delighted to find there were 27 Arabic characters could be recognized, that was the correct rate was 93%. The character that was wrongly recognized were one *2* and one *3.* As discussed above, this is possible.

Fig. 7 shows the frequency of human motion when characters are written. From Fig. 7, we may see that the frequency of human motion is no more than 5 Hz. whereas the sampling frequency of our sensor is 200Hz. Theoretically speaking, the data taken at 200Hz of sampling frequency provides far more than enough information for motion recognition. Another fact is that the higher the sampling frequency is, the slower the system responds. Hence out group starts to investigate on whether lowering the sampling frequency will have any big influence on the recognition results. In the later experiments, we lowered the 200Hz sampling frequency to 40Hz and 20Hz respectively and obtain the corresponding training database and testing data. During the experiments our group is able to obtain an accuracy of 80% and 70%. Although the accuracy is not very high, still, we believe that this experimental result has fully demonstrated that 20Hz or 40Hz could be enough to realize character recognition in the future.

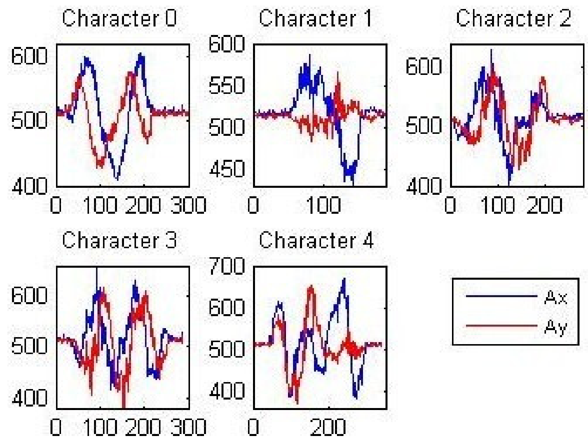


Figure 6. Sensor outputs of characters: *0, 1, 2, 3 and 4*

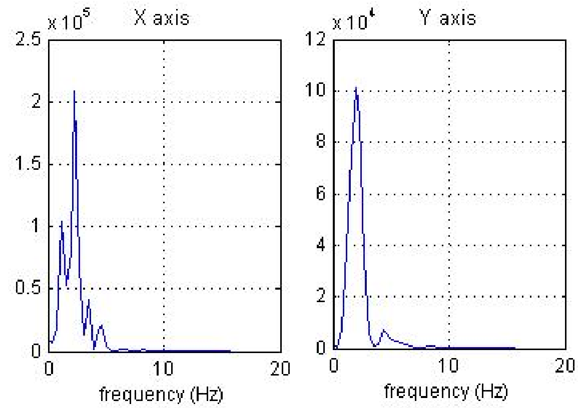


Figure 7. Frequency of human motion

IV. Conclusion

From our previous work we can say that it is hardly for us to construct character trajectory directly from integrals of accelerometer's acceleration information after a hand-writing motion is complete. Thus, another method is needed in order to realize the recognition of characters and hand gestures through acceleration values. In this paper, DCT is used to process the data of quantities of acceleration information from writing numbers *0*, *1*, *2*, *3* and *4* at the beginning, then put these samples into HMM network and have them trained. Then, another 29 new testing samples of the same numbers are tested to recognize. The result is encouraging that 27 out of 29 characters are recognized correctly by this means. Furthermore, we propose a possibility to improve the performance of recognition by lowing sample frequency to synchronize human writing motion better.

References

[1] The Duy Bui, and Long Thang Nguyen, "Recognition of Vietnamese Sign Language Using MEMS Accelerometers", *in Proc. IEEE International Conference on Sensing Technology,* pp. 118-122, November 21-23, 2005.

[2] Won-Chul Bang, Wook Chang, Kyeong-Ho Kang, Eun-Seok Choi, Alexey Potanin, and Dong-Yoon Kim, "Self-contained Spatial Input Device for Wearable Computers", *in Proc. the Seventh IEEE International Symp . on Wearable Computers*, pp. 26-34, 2003.

[3] A. D. Cheok, K. G. Kumar, and S. Prince, "Micro Accelerometer Based Hardware Interfaces for Wearable Computer Mixed Reality Applications", *in Proc. of the 6th IEEE International Symposium On Wearable Computer*s, pp. 223-230, October. 7–10, 2002.

[4] H. Ishikawa, Y. Yonezawa, K. Itoh, and M. Hashimoto, "Extraction of Character Information from Acceleration of Handwriting Pen", *Trans. Society of Instrument and Control Engineers,* vol. J76D-I, pp. 541–543, 1993.

[5] D. Strickland, A. Patel, C. Stovall, J. Palmer, and D. McAllister, "Self Tracking of Human Motion for Virtual Reality Systems", *in Proc. SPIE Int. society of Instrument and Control Engineers,* vol. 2177, pp. 278–287, 1994.

[6] Lee, G.-J. Nam, J. Chae, H. Kim, and A. J. Drake, "Two-Dimensional Position Detection Systems with MEMS Accelerometer for Mouse Applications", *in Proc. of the 38th IEEE International Conference on Design Automation,* pp. 852–857, June 2001.

[7] B. Milner, "Handwriting Recognition Using Acceleration Based Motion Detection", *in Proc. IEEE Coll. Document Image Processing and Multimedia,* London, UK, pp. 5/1–5/6.

[8] Chi Chiu Tsang, Gary Chun Tak Chow, Guanglie Zhang, Yilun Luo, Zhuxin Dong, Guangyi Shi, and Wen J. Li, etc., "A Novel Real-Time Error Compensation Methodology for μIMU-based Digital Writing Instrument", *in Proc. of 2006 IEEE International*

Conference on Robotics and Biomimetics, pp. 678-681, Kunming, China, 17-20 December 2006.

[9] Zhuxin Dong, Guanglie Zhang, Yilun Luo, Chi Chiu Tsang, Guangyi Shi, and Wen J. Li , etc., "A Calibration Method for MEMS Inertial Sensors Based on Optical Tracking", in *Proc of 2007 IEEE International Conference on Nano/Micro Engineered and Molecular Systems,* Bangkok, Thailand, 16-19 January 2007.

[10] Shengli Zhou, Zhuxin Dong and Wen J. Li, and Chung Ping Kwong, "Hand-Written Character Recognition Using MEMS Motion Sensing Technology", *in Proc. IEEE/ASME International Conference on Advanced Intelligent Mechatronic*， 2-5 July 2008, Xi'an, China, pp. 1418-1423.

[11] Sung-Do Choi, Alexander S. Lee, and Soo-Young Lee, "On-Line Handwritten Character Recognition with 3D Accelerometer", in *Proc. of IEEE International Conference on Information Acquisition,* August 20-23, 2006.

[12] Shiqi Zhang, Chun Yuan, Yan Zhang, "Handwritten Character Recognition Using Orientation Quantization Based on 3D Accelerometer", *the 5th Annual International Conference on Mobile and Ubiquitous Systems,* Dublin, Ireland, 25 July 2008.

[13] Sushmita Mitra and Tinku Acharya*, "*Gesture Recognition: A Survey*", IEEE Transactions on systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol. 37, No. 3, May 2007.

[14] Nianjun Liu, Brian C. Lovell, Peter J. Kootsookos, and Richard I.A. Davis, "Model Structure Selection and Training Algorithms for a HMM Gesture Recognition System", International Workshop in Frontiers of Handwritten Recognition, pp. 100-106, October 2004.