Hand Gesture Recognition with Inertial Sensors

Krittameth Teachasrisaksakul[1,2], Liqun Wu[1], Guang-Zhong Yang[1,2], Benny Lo[2]
[1] Department of Computing and [2]The Hamlyn Centre
Imperial College London, London, UK
{kt1312, liqun.wu15, g.z.yang, benny.lo}@imperial.ac.uk

***Abstract*— Dyscalculia is a learning difficulty hindering fundamental arithmetical competence. Children with dyscalculia often have difficulties in engaging in lessons taught with traditional teaching methods. In contrast, an educational game is an attractive alternative. Recent educational studies have shown that gestures could have a positive impact in learning. With the recent development of low cost wearable sensors, a gesture based educational game could be used as a tool to improve the learning outcomes particularly for children with dyscalculia. In this paper, two generic gesture recognition methods are proposed for developing an interactive educational game with wearable inertial sensors. The first method is a multilayered perceptron classifier based on the accelerometer and gyroscope readings to recognize hand gestures. As gyroscope is more power demanding and not all low-cost wearable device has a gyroscope, we have simplified the method using a nearest centroid classifier for classifying hand gestures with only the accelerometer readings. The method has been integrated into open-source educational games. Experimental results based on 5 subjects have demonstrated the accuracy of inertial sensor based hand gesture recognitions. The results have shown that both methods can recognize 15 different hand gestures with the accuracy over 93%.**

I. Introduction

Gesture recognition involves comprehending and classifying meaningful motion of human's arms, hands, or head [1]. Using hand gesture for human-machine interaction will facilitate the interaction process [2]. Hand gesture recognition can find applications in domains including smart device control [3] and robot-assisted living [4]. One of the areas that can be benefitted by the application of hand gesture recognition is education for children with learning difficulties.

Intellectual Disability (ID) is the most prevalent neurodevelopmental disability in the world [5, 6]. A total of 0.8 million young people, aged 18 and younger in the UK have identified with one or more learning disabilities [6]. Learning disabilities can cause difficulties in learning new skills, understanding complex or new information, and hinder independence [7]. Learning disability has significant impact on children learning [7].

Dyscalculia is a learning difficulty in mathematics affecting basic arithmetical skills, e.g. number cognition, counting, and manipulation [8]. Although the exact causes of dyscalculia have not been identified, several hypotheses on the causes have been proposed. The fundamental cause is found to be genetics [9, 10]. Other factors that may cause dyscalculia include brain development, low birth weight, alcohol use during pregnancy, and brain injuries [9, 11]. The symptoms of dyscalculia differ at different age ranges. Preschool children may struggle to count, recognize numeric symbols, and arrange numbers in the correct order [9]. Middle or high school children may struggle to identify and use mathematical signs, memorize fundamental math concepts and associate them with everyday life, and comprehend math-related words including greater than and less than [9, 11].

Traditional teaching approaches are often found to be ineffective in engaging children with dyscalculia particularly in traditional classroom settings. Educational game is an attractive alternative, since it could lead to better engagement and it can be tailored to address individual learning needs. In order to improve interaction and learning outcomes, gesture should be used as the control input of educational games due to two major advantages. First, gestures enable children to devise different problem-solving methods and therefore help them to learn new knowledge [12]. For example, children may describe length using hand gesture allowing them to relate the knowledge in the physical world. Second, it has been demonstrated that gestures are an effective device for children's math study [13]. The study at University of Chicago aimed at assessing children's knowledge generalization ability. Three groups of children were asked to solve equivalence problems using different means of interacting methods. The only group that was able to generalize problems was those who used gestures to solve the problems [13].

Despite the benefits of games to education, there is a lack of online mathematical games for children with dyscalculia. With the widely available wearable devices, hand gesture recognition educational games can be developed as an interactive learning tool to improve the learning outcomes for children with dyscalculia. Games using wearable sensors require high degree of user involvement [14] and lead to better student engagements

In this paper, we present an inertial sensor based hand gesture recognition approach for mobile educational game development. The approach consists of two methods. The first method is based on a multi-layered perceptron classifier for devices with both accelerometer and gyroscope. However, as most very low-cost wearable devices do not have a gyroscope, we have extended our work and developed the second method. The second method is based on a nearest centroid classifier for accelerometer only wearable devices. The method has been integrated into a few open-source educational math games for children with dyscalculia [15]. The following sections will describe the details of the two methods.

II. Hand Gesture Recognition: Method 1

The gesture recognition method consists of data preprocessing, feature extraction and selection, and gesture classification. These steps will be described after the brief description of hand gestures and sensing system.

A. Hand Gesture Description

In this paper, 15 gesture classes (GC) (see Fig. 1) are defined. These gestures are hand movements along different predefined trajectories on a flat table surface. The trajectories include shapes (circle, triangle, square), numerals (0 to 9), and simple movements (moving left, moving right). Each subject was asked to trace each trajectory 10 times with 2 – 3 seconds rest period after each repetition.

B. Sensing System and Sensor Placement

One MTx sensor module by Xsens is used to collect experimental data. The wearable sensor module is equipped with a 3D accelerometer, 3D gyroscope, and 3D magnetometer. The raw sensor data are recorded by the sensor node. The sensor module is physically linked to the Xbus master that synchronizes and sends real-time data to a PC via a wireless connection. The sampling rate of the output signals is 100 Hz. During data acquisition, the sensor module is attached to a glove, which is worn on the back of a subject's right hand.

C. Data Preprocessing

Data preprocessing consists of segmentation, smoothing, and normalization. The signal of each GC contains multiple repetitions of the same gesture. At first, the signal of each GC is manually segmented based on the assumption that the subject's hand stays still for longer than or equal to 2 seconds before and after each gesture repetition. The segmentation step results in the time indices before the first repetition and after the last repetition of each GC. Then signal smoothing is performed to remove high frequency noises. Given a data sample V_t, smoothing results in a smoothed value:

$$S(V_t) = \begin{cases} V_t, & \text{if } t = 0 \\ S(V_{t-1}) + \delta\big(V_t - S(V_{t-1})\big), & \text{if } 0 < t \le T \end{cases} \tag{1}$$

where $t = 0, \cdots, T$ is the time index of a sample; T denote the time point of the last sample in a segmented gesture signal sequence; and δ denote a smoothing factor, which ranges from 0 to 1. δ is set to 0.6 in this method. The lower δ yields a smoother trajectory. Finally, filtered signals are normalized by subtracting them with their means and dividing by their standard deviations.

D. Feature Extraction and Feature Selection

For feature extraction, a 6.40-second window with shifting size of 80 ms is applied to each segmented gesture sequence from the signals of two sensors (accelerometer and gyroscope) each with three axes. A total of 36 ($2 \times 3 \times 6$) features are extracted from each signal window. The six groups of features include mean (f_1), median (f_2), skewness (f_3), standard deviation (f_4), mean absolute deviation (MAD, f_5), and root mean square level (RMSL, f_6). Given a vector $R = [r_0, r_1, r_2, \cdots, r_{P-1}]$ of length P, MAD is defined as

$$MAD = \frac{1}{P}\sum_{j=0}^{P-1}\left| r_j - \frac{1}{P}\sum_{j=0}^{P-1} r_j \right| \tag{2}$$

where $j \in \{0,1,2,\cdots,P-1\}$ denote a time index of R. RMSL is defined as

$$RMSL = \sqrt{\frac{1}{P}\sum_{j=0}^{P-1}\left|r_j\right|^2}. \tag{3}$$

Feature selection was performed by using Wrapper Subset Evaluator, which evaluates attribute sets by using a learning scheme. The search method used during feature selection was greedy stepwise, which performs a greedy forward search through attribute subset space. The feature selection process resulted in the subset of 20 features as described in TABLE I.

E. Classification

A Multi-layer Perceptron (MLP), a variant of Artificial Neural Network, contains multiple layers of neurons, with each layer fully connected to the subsequent layer with directed connections [2]. In this study, a three-layer MLP classifier – containing an input layer, a hidden layer, and an output layer – was implemented. The output of the m-th node of the output layer is defined as

$$y_m = h\left(\alpha_m + \sum_{l=1}^{N_h} p_{ml}\, h\left(\beta_l + \sum_{k=1}^{N_i} q_{lk} x_k\right)\right) \tag{4}$$

where $m = 1, \dots, N_o$. N_h, N_i, N_o are the number of nodes in a hidden layer, an input layer, and an output layer, respectively. x_k is the input of the k^{th} node of the input layer, q_{lk} is the weight between an input layer node and a hidden layer node, p_{ml} is the weight between a hidden layer node and an output layer node, and α_m and β_l are bias terms [16]. Each node uses a non-linear activation function, which is a sigmoid function and is defined as

$$h(a) = \frac{1}{1 + e^{-a}} \tag{5}$$

where a is the weighted sum of the neuron's inputs. The number of nodes in the hidden layer was the mean of the number of attributes and classes. After the MLP classifier is trained, it can be used to classify hand gestures.

III. Hand Gesture Recognition: Method 2

A. Hand Gesture Description

The hand gestures used in this method is the same as those of the first method (Sect. II-A) except that, in the second method, the subject was asked to press a button to signal data recording before and after executing each gesture.

B. Sensing System and Sensor Placement

One TI SensorTag CC2650 sensor node was used to acquire gesture data. It incorporates an MPU-9250 motion sensor and supports Bluetooth Low Energy (BLE) protocol. The output sampling rate is 10 Hz. The sensor readings are streamed to a PC via the BLE connection. During data collection, the subject was asked to hold the shorter sides of the sensor with the thumb and index finger of the right hand.

C. Data Preprocessing

Data preprocessing consists of three steps: data segmentation, smoothing, and linear interpolation. First, data segmentation aims at finding the starting point and ending point of each gesture sequence. A simple button triggering based segmentation approach is used, i.e. pressing a button before and after performing a gesture in order to signal data recording. However, this approach still requires further segmentation in the head and tail of gesture signals. In this study, data samples of length 0.2 – 0.3 seconds were removed from both the head and tail of each gesture sequence. Signal smoothing (as described in Eq.(1)) is performed with $\delta = 0.3$. Finally, linear interpolation is applied to remove the effect of phase differences so that gesture signals can be compared regardless of motion speed. Linear interpolation generates new data points using existing data points. In this method, gesture signal is linearly interpolated to the length of 10 seconds, i.e. 100 samples.

D. Classification

Nearest centroid classification is a simplified variant of $k-$means classifier. A k-mean classifier defines multiple clusters and their centroids; classifies an arbitrary input into a cluster with the shortest relative distance; and continue repeating these steps until the total distance is below a threshold. In contrast, nearest centroid method performs cluster sorting only once and k is defined as a gesture database's size.

To construct a nearest centroid classifier, two steps are required. First, the gesture database is constructed by acquiring standard gesture classes. For each gesture, the standard gesture class is the mean value of 50 gesture sequences, obtained from 5 subjects performing the same gesture 10 times. Second, in order to recognize gesture, a Euclidean distance between the mean value of an input gesture sequence and each standard gesture class in the database is computed:

$$d_{i,c} = \sqrt{\sum (u_i - u_c)^2} \tag{6}$$

where $u_i = [u_{i,x} \quad u_{i,y} \quad u_{i,z}]^T$ denote the vector of mean acceleration values of an input gesture, and $u_c = [u_{c,x} \quad u_{c,y} \quad u_{c,z}]^T$ denote the vector of mean values of standard gesture class. The classification selects the class with the smallest distance as a result.

TABLE I
Selected Features of Method 1 (All denote all three axes of the feature are selected).

Features	f_1	f_2	f_3	f_4	f_5	f_6
Accelerometer	*y*	all	all	all		*x, y*
Gyroscope	*x*		all	*y*	*y*	*x, z*

IV. Experimental Results and Discussions

A. Experimental Results of Method 1

The multi-layer perceptron model of Method 1 as trained using gesture data of 5 subjects tracing 15 hand gestures (Sect. II-A). Each gesture was repeated 10 times. The classifier was implemented in Weka [17]. The model's learning rate, momentum, and batch size was set to 0.01, 0.9, and 100, respectively. The learning rate decay was not applied. The model was evaluated with the same dataset but different evaluation approaches and training data percentage. The gesture recognition accuracy of Method 1 against different evaluation approaches is listed in TABLE II. The accuracy of the approach exceeded 93.12%.

B. Experimental Results of Method 2

The nearest centroid classifier was constructed from gesture data of 5 subjects performing 15 gestures (Sect. II-A) for 10 times each. The recognition accuracy of different gestures is listed in TABLE III. The overall gesture recognition accuracy was 93.60%. However, the accuracy of the digit 6 was 62.00% and the lowest of all gestures because these gestures were misclassified as circle gestures and vice versa. This was due to similar movements of the digit 6 and the circle. Next, we aim to investigate whether using a 2D Euclidean distance, by excluding the *z*-axis readings, instead of the 3D distance during classification affects gesture recognition results. The rationale behind this investigation is that the hand movements are approximately on a 2D plane, i.e. a flat table surface. The accuracy of using 2D distance for classification was 93.33%. Hence, using 2D distance did not have a significant effect on recognition accuracy.

TABLE II
Overall Gesture Recognition Accuracy of Method 1 Against Different Evaluation methods (* denote Data Randomized).

Evaluation Methods	Training Data (%)	Number of repetitions	Mean Accuracy (SD)
10-fold cross validation	90%	3	95.36 (0.65)
Percentage split (*)	20%	10	94.64 (0.99)
	10%	10	93.12 (0.70)

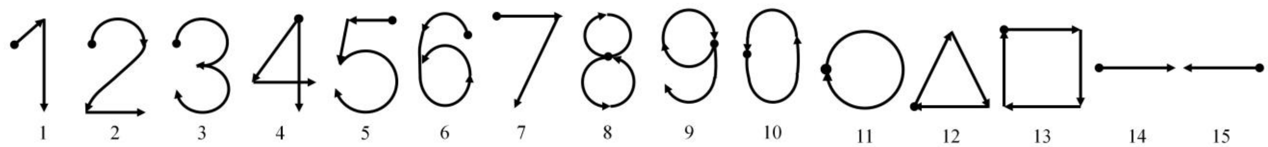

Fig. 1: Trajectories of fifteen gesture classes with their corresponding numbers. The dots denote the starting point of the trajectory. The digit 4 is traced with two strokes with the same starting point while the other gestures are traced with one stroke.

TABLE III
GESTURE RECOGNITION ACCURACY OF METHOD 2 WHEN USING 3D (WITH Z-AXIS) AND 2D (WITHOUT Z-AXIS) EUCLIDEAN DISTANCE (NOTATION: D = DIGIT, ML = MOVING LEFT, MR = MOVING RIGHT, S = SQUARE, C = CIRCLE, T = TRIANGLE).

Gestures	**D0**	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**	**D7**	**D8**	**D9**	**ML**	**MR**	**S**	**C**	**T**	**Overall**
With *z*-axis	100	94	98	92	96	100	62*	100	94	100	100	96	98	80	94	93.60
Without *z*-axis	100	92	98	92	96	100	62*	100	94	100	100	96	98	76	96	93.33

C. Discussions

Both methods have comparable overall recognition accuracy. However, Method 2 involves interpolation during preprocessing; and thus requires a gesture data template and the template's length. If the number of classes increases, resolution of the template needs to be re-determined prior to classification. In contrast, Method 1 extracts features based on a sliding window; and can be extended to classify more gestures without requiring the template and its length. However, Method 1 requires the use of both gyroscope and accelerometer which are not often available in low cost wearable sensors. Method 2, on the other hand, is a light-weight approach achieving similar accuracy as per Method 1 by using only the accelerometer.

V. CONCLUSION

In this paper, we have proposed a wearable inertial sensor based approach for hand gesture recognition and proposed 2 classification methods. The first method uses multilayered perceptron to recognize 15 different hand gestures, and the second method is based on a nearest centroid based recognition for gesture recognition. As opposed to Method 1, Method 2 does not require gyroscope readings, but Method 1 can be generated with more gestures for different learning games. The trained models of both approaches were evaluated on a hand gesture dataset of 5 subjects, performing 15 gesture classes. The experimental results demonstrate that the gesture recognition accuracies of both proposed approaches exceed 93%.

REFERENCES

[1] A. Akl, C. Feng, and S. Valaee, "A novel accelerometer-based gesture recognition system," *IEEE Trans. on Signal Process.*, vol. 59, pp. 6197-6205, 2011.

[2] R. Xie and J. Cao, "Accelerometer-based hand gesture recognition by neural network and similarity matching," *IEEE Sensors J.*, vol. 16, pp. 4537-4545, 2016.

[3] H. P. Gupta, H. S. Chudgar, S. Mukherjee, T. Dutta, and K. Sharma, "A continuous hand gestures recognition technique for human-machine interaction using accelerometer and gyroscope sensors," *IEEE Sensors J.*, vol. 16, pp. 6425-6432, 2016.

[4] C. Zhu and W. Sheng, "Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, pp. 569-573, 2011.

[5] American Academy of Pediatrics, 2015. Healthy children. Available: https://www.healthychildren.org/English/health-issues/conditions/developmental-disabilities/Pages/Intellectual-Disability.aspx

[6] C. Blackburn, J. Read, and N. Spencer, "Children with neurodevelopmental disabilities," *Annual Report of the Chief Medical Officer 2012*, 2012.

[7] NHS, 2015. What is a learning disability?, Available: http://www.nhs.uk/Livewell/Childrenwithalearningdisability/Pages/Whatislearningdisability.aspx

[8] SOAS University of London, Specific learning differences. Available: https://www.soas.ac.uk/studentadviceandwellbeing/information-for-staff/disabledstudents/learningdifficulties/

[9] C. Chakravarty, 2017. Dyscalculia or mathematics disorder in children. Available: http://www.momjunction.com/articles/dyscalculia-or-mathematics-disorder-in-children_00349446/

[10] J. B. Hale, 2017. Causes of dyscalculia & math learning difficulties. Available: http://www.dyscalculia.org/dyscalculia/math-ld-causes

[11] The Understood Team, 2016. Understanding Dyscalculia.

[12] B. M. Law, 2005. Gestures give learning a hand. Available: http://www.apa.org/monitor/nov05/gestures.aspx

[13] Association For Psychological Science, 2014. Gesturing with hands is a powerful tool for children's math learning. Available: http://www.psychologicalscience.org/news/releases/gesturing-with-hands-is-a-powerful-tool-for-childrens-math-learning.html

[14] F. P. Sze, H. C. Ping, and M. Y. Zhen, 2014. Motion sensing game. Available: http://www.slideshare.net/patsyfung/motion-sensing-game

[15] L.-Q. Wu, 2016. An interactive game for children with learning difficulties. Available: https://github.com/leowu4ever/IP

[16] C.-H. Chen, J.-C. Wu, and J.-H. Chen, "Prediction of flutter derivatives by artificial neural networks," *J. Wind Eng. Ind. Aerodyn.*, vol. 96, pp. 1925-1937, 2008.

[17] E. Frank, M. A. Hall, and I. H. Witten, "The WEKA Workbench," *Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*, M. Kaufmann, Ed., 2016