Gesture Recognition Benchmark Based on Mobile Phone

Conference Paper *in* Lecture Notes in Computer Science · October 2016

DOI: 10.1007/978-3-319-46654-5_48

Chunyu Xie
Beihang University (BUAA)
10 PUBLICATIONS **342** CITATIONS

Shangzhen Luan
Beihang University (BUAA)
7 PUBLICATIONS **438** CITATIONS

Baochang Zhang
Beihang University (BUAA)
304 PUBLICATIONS **8,435** CITATIONS

Gesture Recognition Benchmark Based on Mobile Phone

Chunyu Xie[1], Shangzhen Luan[1], Hainan Wang[1,2], and Baochang Zhang[1(✉)]

[1] School of Automation Science and Electrical Engineering, Beihang University, Beijing, China
bczhang@buaa.edu.cn
[2] School of Mechanical Engineering, Guizhou University, Guiyang, China

Abstract. Mobile phone plays an important role in our daily life. This paper develops a gesture recognition benchmark based on sensors of mobile phone. The built-in micro gyroscope and accelerometer of mobile phone can efficiently measure the accelerations and angular velocities along x-, y- and z-axis, which are used as the input data. We calculate the energy of the input data to reduce the effect of the phone's posture variations. A large database is collected, which contains more than 1,000 samples of 8 gestures. The Hidden Markov Model (HMM), K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) are tested on the benchmark. The experimental results indicated that the employed methods can effectively recognize the gestures. To promote research on this topic, the source code and database are made available to the public. (mpl.buaa.edu.cn or correspondence author)

Keywords: SVM · HMM · KNN · Gesture recognition · Mobile phone

1 Introduction

With the advance of electronic technology, daily activities are closely related to computer and network technology. Human-computer interaction (HCI) becomes an important part of our daily life. The emergence of smart phones allows HCI to be performed effectively through touch screen. However, the touch screen, affected by environmental constraints, is prone to many uncontrollable problems. Therefore, other types of interaction, such as gesture input [1], attract the attention of researchers.

For HCI using gesture input, the key component is the recognition of gestures. As an efficient way of communication, human gesture is easy to implement and understand [2]. Currently, there are two kinds of gesture recognition methods, including vision-based recognition and inertial sensor-based recognition [14–17]. However, because of

B. Zhang—This work was supported in part by the Natural Science Foundation of China under Contract 61272052 and Contract 61473086, in part by PAPD, in part by CICAEET, and in part by the National Basic Research Program of China under Grant 2015CB352501. The work of B. Zhang was supported by the Program for New Century Excellent Talents University within the Ministry of Education, China, and Beijing Municipal Science & Technology Commission Z161100001616005. Baochang Zhang is the corresponding author.

Z. You et al. (Eds.): CCBR 2016, LNCS 9967, pp. 432–440, 2016.
DOI: 10.1007/978-3-319-46654-5_48

the diverse gestures and demanding environment, the first method is relatively time consuming. For the second method, it is in constant development in recent years [3–5].

In the last two decades, researchers have carried out experiments by using inertial sensors to identify simple operations and achieved some good results. Due to the huge volume and expense of related equipment, the device cannot be used in individual level. Therefore this technology has been difficult to make progress. At the beginning of this century, the rapid development of mobile phones provided hardware platforms, which made gesture recognition by acceleration usher in the dawn. In 2000, Rekimoto tried to use a specific type of wearable device to detect the motion of the arm, but the size of the equipment is too large, and the precision obtained is very low, making this method not suitable in practice [6]. In 2004, the theory of static acceleration and dynamic acceleration was published by Jang and Park, and at the same time they put forward their own views about recognition issues [7]. Fresca and Resmritas categorized gestures through a law developed, including two categories of basic gestures and combined gestures. They established a database of gestures, and classified gestures based on the database [8]. Juha put forward the concept of base interfaces, and obtain different features of different gestures [9]. F. G. Hofmann proposed a recognition scheme based on the acceleration vectors from different gestures [10]. Baek, who further refined gestures and obtained better results [11].

Most studies of gesture recognition are based on an accelerometer, which pose restriction on the type of gestures and the recognition accuracy. With MEMS sensors are first built in iphone4, including linear accelerometers and micro-gyroscopes, various Android phones also include these basic functions, which provide more excellent hardware conditions of mobile phone gesture recognition. Based on inertial sensors such as accelerometers and gyroscopes, mobile phones can measure gesture information directly. At present, the main ways of gesture recognition are: decision trees, dynamic time warping technique, support vector machines, hidden Markov models, etc.

Similar to other areas of recognition, public databases are very important to researchers in the field of gesture recognition. However, as far as we know, there is no such database related to gesture recognition based on inertial sensors of mobile phones, while the benchmark databases in other biometric problems are normally ample. For example, the FERET and FRGC databases are commonly used in face recognition [12, 13].

Our database was created to enable more researchers and practitioners to evaluate and compare their algorithms by using the same data. We record the inertial sensor data of each gesture in a text file. It is worth noting that most of the previous experiments were conducted in a laboratory environment, using their own data. Therefore creating an open and comprehensive database for gesture recognition is of benefit to the research community.

In this paper, we focus more on data preprocessing, feature extraction, and recognition algorithms of HMM, KNN and SVM. The rest of the paper is organized as follows. In Sect. 2, we introduce the gesture recognition system, describing the basic theory and processes. The specific methods of preprocessing and feature extraction are presented in Sects. 3 and 4. Section 5 displays our experiment results of HMM, KNN and SVM. Section 6 concludes this paper and makes some discussions on future work.

2 Gesture Recognition Based on Mobile

By using mobile phone, we obtain the original data from the phone's built-in accelerometer and gyroscope. We carry out data preprocessing and feature extraction. The gestures include English alphabet and Arabic numerals, which are very common characters and have a standard way of writing. This paper selects four representative capital letters of A, B, C, D and four Arabic numerals of 1, 2, 3, 4 to carry out gesture recognition research.

The flowchart of the gesture recognition approach is shown in Fig. 1. When the phone records gestures, we transfer data into digital signal of three-dimensional acceleration and angular velocity by A/D conversion. Then we train an HMM or SVM for each type of gesture through the preprocessing and feature extraction, and finally make a classification decision.

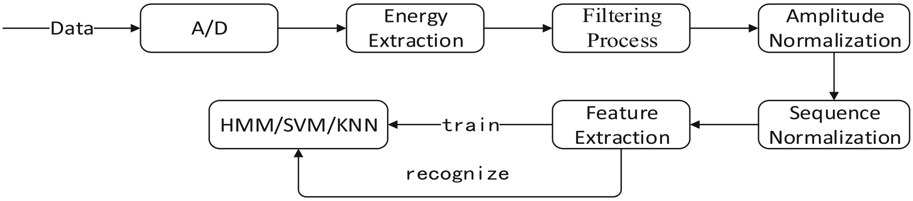

Fig. 1. Flowchart of the gesture recognition system

3 Data Preprocessing

3.1 Energy Calculation

In the experiments, we collect gestures of the capital letters C in different ways. Figure 2 shows angular velocity and acceleration. The angular velocity and acceleration have three directions, X, Y and Z. Z-axis direction is perpendicular to the screen of the phone. The X-axis direction represents right direction of the phone, and Y-axis means upward direction. The data curves are quite different when the phone does not have the same attitude, even if it has the same action, which makes it difficult for accurate recognition.

If we denote the XYZ-axis acceleration data as accx, accy and accz, the XYZ-axis angular velocity data as gyrox, gyroy and gyroz, and the energy signal of the acceleration data and angular velocity data as accs and gyros, acceleration data, angular velocity data and energy signal will have the following relationships:

$$accs = accx^2 + accy^2 + accz^2. \tag{1}$$

$$gyros = gyrox^2 + gyroy^2 + gyroz^2. \tag{2}$$

Using Eqs. (1) and (2), the data in Fig. 2 is converted to the data shown in Fig. 3. As it is apparent from Fig. 3, different postures we draw C, using a mobile phone, have

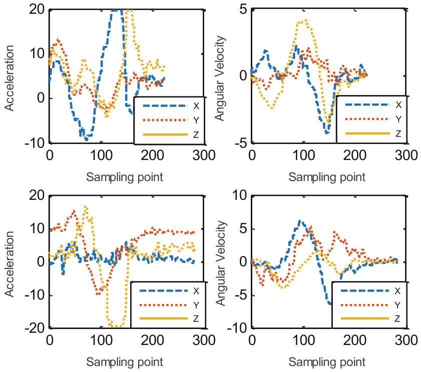

Fig. 2. Different postures for Gesture C in each row.

almost the same energy signal of the acceleration and angular velocity. The energy signal of angular velocity and acceleration have a clear advantage on gesture clustering and recognition, which can greatly increases the recognition rate.

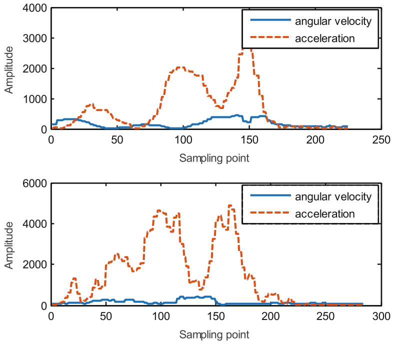

Fig. 3. Energy signals for Fig. 2

3.2 Wave Filtering

Affected by laboratory equipment and the environment, the data from the sensor inevitably contain noise. Therefore, we further carry out a filtering process to reduce noise. For time-domain signal, we can use Average Filter or Median Filter to remove high-frequency interference of background signal in the data. Since the gesture signals have relatively low frequency in the frequency domain, so we should use a low-pass filter.

Figure 4 shows the original data and the processed data by the Average Filter, Median Filter and Butterworth Filter. We can obviously find the Average Filter and Butterworth Filter are better. However, it should be noted that Butterworth Filter has the process of DFT or FFT, which cannot meet the requirement of less calculation cost. Thus, we choose the Average Filter which is relatively simple and efficient.

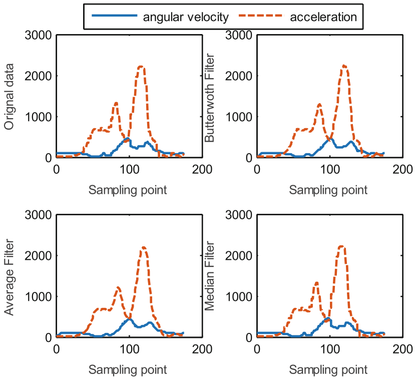

Fig. 4. From top (left) to bottom (right): the original data, and the processed data by Average Filter, Median Filter, Butterworth Filter

3.3 Amplitude Normalization and Sequence Normalization

Different gestures may have different speeds and intensities. The difference of intensity has an impact on the amplitude of data. Meanwhile, when the speeds of action are different, gestures completed relatively fast have fewer sampling points since the sampling frequency (200 Hz) of acceleration and the angular velocity signals is fixed for a mobile phone. So we need to carry out amplitude normalization to mitigate interference caused by the differences in action's intensity. In addition, we normalize the data obtained at different speeds to have the same length, which eliminates the impact of action's speed by sequence normalization.

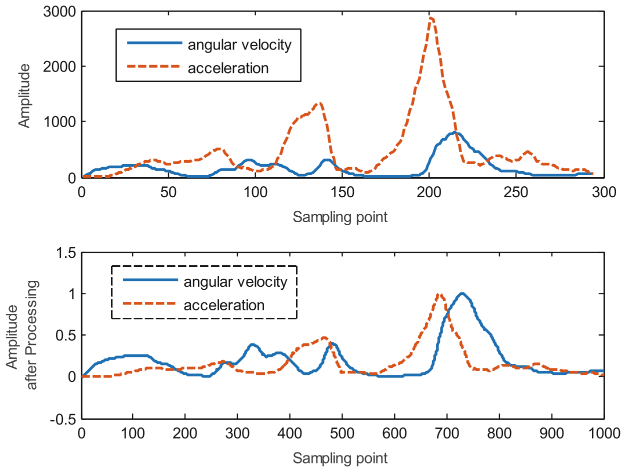

Fig. 5. The original data (top) and the preprocessed data (bottom).

This paper uses cubic spline interpolation to set the sampling point of data to the same number. Figure 5 shows the waveform of data after the process of amplitude normalization and sequence normalization, which has effectively eliminated the influence of intensity and speed variations.

4 Feature Extraction

After preprocessing, the length of each sample is of 1000 sampling points. For a gesture, if we simply use these data, the difference between the samples corresponding to the same gesture is large, thus we cannot achieve a good recognition performance. We have to extract compact and discriminative features, not only effective for gesture recognition, but also for compression.

We utilize the method of computing the mean of the energy signal. The segment length is set to be 30 sampling points, and the segment shift has 20 sampling points, which can guarantee the continuity of data. We calculate the mean of every segment's acceleration energy and angular velocity energy to obtain their characteristic sequences. Figure 6 shows characteristic curves of the capital letters of A in different poses, intensities and speeds.

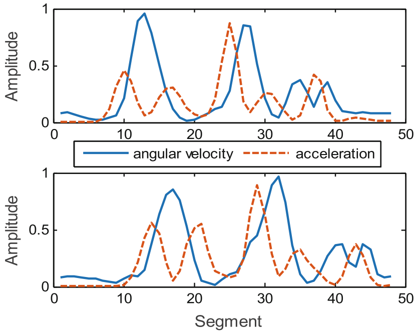

Fig. 6. The curve of different As

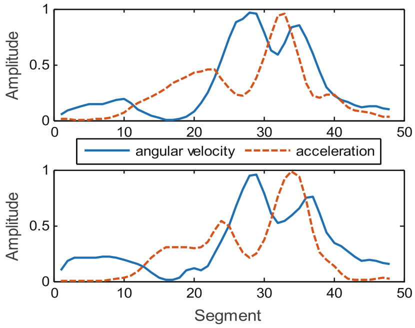

Fig. 7. The curve of different Cs

5 Baseline Algorithms and Results

There are several methods widely used in statistical classifications. In this paper, we use 3 state-of-the-art algorithms: SVM, KNN and HMM based methods. The SVM solution uses maximum-margin hyperplanes to classify, while the KNN solution selects the nearest Euclidean distance and the HMM uses the maximum of likelihood as a decision rule. The details of the parameters of the baseline algorithms can be found in the source codes that will be publically available. In the following, we use a 14-fold cross validation, where each category training set has 130 samples and test set has 10 samples. Based on these algorithms, the paper obtains the overall recognition accuracy rate (see Table 1).

Table 1. Recognition results comparison on the BUAA mobile gesture dataset

Algorithm	SVM	KNN	HMM
A	97.14	92.14	95.00
B	88.57	82.14	95.71
C	89.29	89.29	84.29
D	93.57	89.29	80.00
1	97.86	95.00	97.14
2	90.00	92.14	93.57
3	97.86	96.43	96.43
4	86.43	90.71	94.29
Average	93.84	90.54	91.79

6 Conclusion

This paper introduces a mobile based gesture recognition benchmark by building a gesture database of three-dimensional acceleration and angular velocity. Data preprocessing is performed including energy signal calculation, filtering, normalization and feature extraction. To evaluate the recognition performance, we use the algorithms of HMM, SVM and KNN. Our future work will focus on boosting the classifiers and finding more powerful features to improve the performance. Related work can also refer to [18–21].

The BUAA Mobile Gesture Database

This paper uses an Android 5.1 platform mobile phone to collect data, and the sampling frequency of data is taken as 200 Hz. For gestures of A, B, C, D, 1, 2, 3, 4 we collected 1,120 samples, and each sample includes three-dimensional acceleration and angular velocity of the mobile phone. All data is saved in text file, which are originally collected without any manual modification. The data is classified and stored in different folders, named by the type of gesture. To download the database for research purpose, one can send an email to the corresponding author or visiting mpl.buaa.edu.cn.

References

1. Lane, N.D., Miluzzo, E., Lu, H., et al.: A survey of mobile phone sensing. J. IEEE Commun. Mag. **48**(9), 140–150 (2010)
2. Choi, E.S., Bang, W.C., Cho, S.J., et al.: Beatbox music phone: gesture-based interactive mobile phone using a tri-axis accelerometer. In: 2005 IEEE International Conference on Industrial Technology, pp. 97–102. IEEE Press, Hong Kong (2005)
3. Mantyla, V.M, Mantyjarvi, J., Seppanen, T., et al.: Hand gesture recognition of a mobile device user. In: 2000 IEEE International Conference on Multimedia and Expo, pp. 281–284. IEEE Press, New York (2000)
4. Liu, J., Zhong, L., Wickramasuriya, J., et al.: uWave: accelerometer-based personalized gesture recognition and its applications. Pervasive Mob. Comput. **5**, 657–675 (2009)
5. Yazdi, N., Ayazi, F., Najafi, K.: Micromachined inertial sensors. Proc. IEEE **86**, 1640–1659 (1998)
6. Rekimoto, J.: Gesturewrist and Gesturepad: unobtrusive wearable interaction devices. In: Fifth International Symposium on Wearable Computers, pp. 21–27. IEEE Press, Zurich (2001)
7. Jang, I.J., Park, W.B.: Signal processing of the accelerometer for gesture awareness on handheld devices. In: The 12th IEEE International Workshop on Robot and Human Interactive Communication, pp. 139–144. IEEE Press, Roman (2003)
8. Ferscha, A., Resmerita, S.: Gestural interaction in the pervasive computing landscape. e & i Elektrotechnik und Informationstechnik. **124**, 17–25 (2007)
9. Kallio, S., Kela, J., Mantyjarvi, J.: Online gesture recognition system for mobile interaction. In: IEEE International Conference on Man and Cybernetics, pp. 2070–2076. IEEE Press, Washington, D.C (2003)
10. Hofmann, F.G., Heyer, P., Hommel, G.: Velocity profile based recognition of dynamic gestures with discrete hidden markov models. In: Wachsmuth, I., Fröhlich, M. (eds.) GW 1997. LNCS (LNAI), vol. 1371, pp. 81–95. Springer, Heidelberg (1998)
11. Baek, J., Jang, I.-J., Park, K., Kang, H.-S., Yun, B.-J.: Human computer interaction for the accelerometer-based mobile game. In: Sha, E., Han, S.-K., Xu, C.-Z., Kim, M.-H., Yang, L.T., Xiao, B. (eds.) EUC 2006. LNCS, vol. 4096, pp. 509–518. Springer, Heidelberg (2006)
12. Phillips, P.J, Flynn, P.J, Scruggs, T., et al.: Overview of the face recognition grand challenge. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), pp. 947–954. IEEE Press, San Diego (2005)
13. Phillips, P.J., Moon, H., Rizvi, S.A., et al.: The FERET evaluation methodology for face-recognition algorithms. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1090–1104 (2000)
14. Chen, C., Jafari, R., Kehtarnavaz, N.: Improving human action recognition using fusion of depth camera and inertial sensors. IEEE Trans. Hum.-Mach. Syst. **45**, 51–61 (2015)
15. Chen, C., Jafari, R., Kehtarnavaz, N.: A real-time human action recognition system using depth and inertial sensor fusion. IEEE Sens. J. **16**, 773–781 (2016)
16. Liu, K., Chen, C., Jafari, R., et al.: Fusion of inertial and depth sensor data for robust hand gesture recognition. IEEE Sens. J. **14**, 1898–1903 (2014)
17. Chen, C., Jafari, R., Kehtarnavaz, N.: A survey of depth and inertial sensor fusion for human action recognition. Multimedia Tools Appl. 1–21 (2015)
18. Chen, C., Liu, M., Zhang, B., Han, J., Jiang, J., Liu, H.: 3D action recognition using multi-temporal depth motion maps and fisher vector. In: International Joint Conference on Artificial Intelligence, pp. 3331–3337 (2016)
19. Zhang, B., Perina, A., Li, Z., Murino, V., Liu, J., Ji, R.: Bounding multiple gaussians uncertainty with application to object tracking. Int. J. Comput. Vis. **118**, 364–379 (2016)

20. Zhang, B., Li, Z., Perina, A., Del Bue, A., Murino, V.: Adaptive local movement modeling (ALMM) for object tracking, IEEE TCSVT (2016). doi:10.1109/TCSVT.2016.2540978
21. Zhang, B., Perina, A., Murino, V., Bue, A.D.: Sparse representation classification with manifold constraints transfer. In: Computer Vision and Pattern Recognition, pp. 4557–4565. IEEE press, Boston (2015)