2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Large-scale Gesture Recognition with a Fusion of RGB-D Data Based on the C3D model

Yunan Li, Qiguang Miao*, Kuan Tian, Yingying Fan, Xin Xu, Rui Li and Jianfeng Song
School of Computer Science and Technology, Xidian University,
Xi'an, PR China
*Corresponding author E-mail: qgmiao@126.com

***Abstract*—The gesture recognition has raised attention in computer vision owing to its many applications. However, video-based large-scale gesture recognition still faces many challenges, since many factors like background may disturb the accuracy. To achieve gesture recognition with large-scale videos, we propose a method based on RGB-D data. To learn gesture details better, the inputs are expanded into 32-frame videos first, and then the RGB and depth videos are sent to the C3D model to extract spatiotemporal features respectively. Next these features are combined to boost the performance, which can also avoid unreasonable synthetic data due to the uniform dimension of C3D features. Our approach achieves 49.2% accuracy on the validation subset of the Chalearn LAP IsoGD Database just with a linear SVM classifier. It also outperforms the baseline and other methods in the challenge and wins the first place at 56.9% on testing set.**

I. Introduction

The gesture recognition, of which the intention is interpreting human gestures to machines via some algorithms, is of great importance in computer vision since many applications, such as video surveillance, sign language comprehension, virtual reality and most commonly, the human computer interaction (HCI) are on the basis of it.

Although gesture recognition seems just recognizing and understanding the physical movements of human body, there are many challenges still associated with the accuracy and practicability of it. First, gestures are ambiguous since a gesture may map to different meaning along with the variation of situation, and vice versa. For example, the gesture of stretching two fingers means "victory" , but it can also mean the number "2" in some conditions. Meanwhile, the concept of "stop" , may match a palm facing forward in the traffic gesture in China, while it can also map to a finger vertically touching another palm. Second, the environment, such as background, performers' clothes and skin color may disturb the recognition since these variants are uncorrelated with gestures. Compared with general gesture recognition via images, video-based recognition may face more challenges. As it detects gestures in video, the motion instead of simply posture recognition is necessary. Then the reaction time, amplitude of movement and video quality may affect the accuracy as well.

In this paper, we propose a video gesture recognition method based on RGB-D data, i.e. RGB and depth data that captured simultaneously via a Kinect sensor. The goal of ChaLearn LAP Large-scale Isolated Gesture Recognition Challenge [1] is to solve large-scale gesture learning and recognizing with user independent RGB-D videos. As similar gestures are integrated with the unique label [2], the challenges we encounter are all aforementioned except for the gesture ambiguity. To learn the details of gesture better, we first conduct a pre-processing on the input videos and convert them to 32-frame videos. Then the features of RGB and depth videos are extracted respectively by the C3D model, a 3D convolutional network model that learns spatiotemporal features. Next these features are blended to boost the performance. The final classification is implemented by a linear SVM classifier. Our approach achieves the accuracy of 49.2% on validation subset and 56.9% on the testing subset of the Chalearn LAP IsoGD Database respectively, and takes the first place of ChaLearn LAP Large-scale Isolated Gesture Recognition Challenge ultimately. It benefits mainly from two contributions as summarized below:

- The expansion to 32-frame videos. The 16-frame input required by the original the C3D model seems to remove too much information contained in the original videos. Therefore some similar gestures may be misunderstood and put into one category. This expansion helps with increasing the information of inputs and makes it easier to track the path of gestures.

- The fusion scheme of features. Since the RGB and depth videos are not matched well (objects in RGB video are a little bigger), the fusion is employed in the later stage with the extracted features. The fusion is processed by either averaging the two features or integrating them to generate a feature with higher dimension. The experiments prove that both of our fusion schemes are beneficial for boosting the final performance to a large extent.

The remainder of this paper is organized as follows. In Section II, the development of video gesture recognition is briefly reviewed and the Chalearn LAP IsoGD Database is also introduced. Subsequently, our proposed video gesture recognition method with RGB-D data based on the C3D model is described in Section III. Following in Section IV, the experiments that prove the effectiveness of our approach and the comparisons among our approach, baseline method and other available methods are given. Finally our approach is concluded and the future work is given in the last section.

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

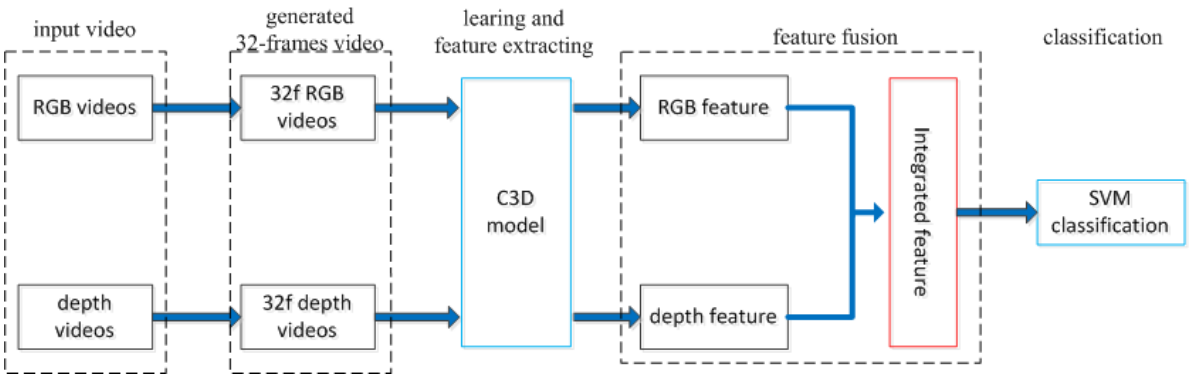


Fig. 1. The flowchart of our approach. First the input videos are converted into 32-frame ones. The two kinds of videos are sent to the C3D model respectively and features of each one are extracted. The features are blended to improve the performance and the integrated feature are used for classification via linear SVM classifier.

II. Related work

A. Development of gesture recognition

Gesture recognition has been studied for decades. There are various approaches to handle gesture recognition problem. In the early stage, the hand-crafted features like histogram of oriented gradients (HOG) and histogram of optical flow (HOF) are extracted [3] [4] [5], and the finite-state machine (FSM) [6], [7], hidden Markov model (HMM) [5] [8], and dynamic time warping (DTW) [9] are commonly applied in modeling human gestures. Then the traditional features are extended into sptiotemporal domain to generate more effective features for video data. For example, Klaser *et al.* [10] propose a 3D HOG feature for action recognition. Wan *et al.* extend the scale invariant feature transform (SIFT) and propose 3D enhanced motion SIFT (3D EMoSIFT) [11] and 3D Sparse Motion SIFT (3D SMoSIFT) [12] to extract features with the fusion of RGB-D data. Then they propose the mixed features around sparse keypoints (MFSK) [13] specifically for one shot learning of gesture recognition.

With rapid development of deep learning and powerful hardware like GPU, the Convolutional Neural Network has made a great breakthrough on visual recognition, including gesture recognition. Le *et al.*[14] use stacked ISA to learn spatiotemporal features in videos. Karpathy *et al.*[15] propose a CNN-based model to classify videos on large-scale datasets. Ji *et al.*[16] employ a hardwired layer to extract some hand-crafted features like optical flow and gradient first and then send them all into a 3D CNN to extract features. Simonyan and Zisserman [17] increase the amount of training data by multi-task learning and extract the spatial concurrent with temporal features by two-stream convolutional networks. Tran *et al.*[18] propose a C3D model, a 3D CNN as well which is a modified version of BVLC caffe [19] and processes directly on video clips. This method achieves a promising accuracy even on large-scale datasets, and recently many a method are on the basis of it like [20] and [21].

B. The Chalearn LAP IsoGD Database

The Chalearn LAP IsoGD Database is built by Wan *et al.* [2]. which is derived from the CGD dataset [22]. The goal of this dataset is to complete the task of user independent recognition - the person performs in training data will not appear in validation or testing data. There are 47933 gestures solely exist in the same amount of videos in this dataset, which are provided with both RGB and depth video and can be divided into training (35878 videos), validation (5784 videos) and testing (6271 videos). The gestures are labeled ranging from 1 to 249.

III. The video gesture recognition with RGB-D data based on the C3D model

As mentioned by Wan *et al.*[2], gesture recognition experiences many difficulties in feature extraction. When the input is video rather than still image, this task needs more endeavors since the temporal features also need to be learned. Thanks to the bloom of deep learning, the features can be learned automatically from the temporal and spatial domains simultaneously. The flowchart of our approach is depicted in Fig.1. We first convert the input RGB-D data into 32-frame videos. Then the features of these videos are extracted respectively by a 3D Convolutional Neural Network - the C3D model, and blended together later. The integrated feature is sent to a linear SVM classifier for the finally result. The details of our C3D model implementation, the 32-frame unification strategy, and the fusion schemes will be introduced in following subsections.

A. Feature extraction model

Since gestures are presented in videos, the gesture recognition is essentially to recognize motions. Thus the features among frame sequences are also required. To this end, we employ the C3D model [18] to extract such spatiotemporal features. The C3D model is a kind of 3D Convolutional Neural Network. Compared with the traditional 2D CNN, the C3D model is able to model temporal information with the scheme of 3D convolution and 3D pooling. Hence it achieves better result when the inputs are video clips.

The ensemble architecture of the C3D model is illustrated in Fig.2. This model consists of 8 convolution, 5 pooling, 2 fully connected layers to learn features and a softmax layer to provide predicted label. All of convolution layers are with the same kernel size of $3 \times 3 \times 3$ with stride 1, whereas the kernel size of pooling layers is $2 \times 2 \times 2$ except for the pooling 1 layer, of which the kernel size is $1 \times 2 \times 2$ to keep the temporal information in the early stage of the network. After times of

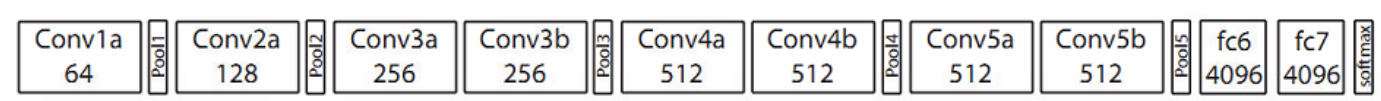
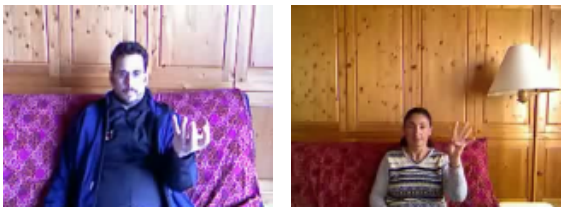


Fig. 2. The architecture of the C3D model (from Tran *et al.*'s paper [18]). It consists of 8 convolution layers, 5 pooling layers, 2 fully-connected layers and a softmax loss layer. The feature we extract is from fc6 layer, i.e., the first fully-connected layer.

convolution and pooling, the input video is converted into a 4096-dim vector in the first fully connected layer - fc6, and mapped to the predicted label after another fully connected layer. The input of the C3D model is required to be split into 16-frame clips and randomly cropped into 112×112 to match the structure of network according to Tran *et al.*[18].

However, training a deep network is very time-consuming since there are millions of parameters waiting for adjustment. It seems more arduous for the large-scale datasets like the ChaLearn LAP IsoGD Database, which has 249 categories of gestures. As the C3D model that pre-trained on Sports-1M dataset (which is the largest video classification benchmark with 1.1 million sports videos in 487 categories) is accessible, we can directly finetune with videos from the ChaLearn LAP IsoGD Database. The effectiveness of such a pre-trained model has been verified by Tran *et al.*[18]. As both RGB and depth video are available, we finetune that model with them respectively. The features of fc6 are extracted after 28 epochs.



(a) (b)

Fig. 3. The similar gestures. (a) the video 03138 that wrongly classified into category 10 by the original C3D model. (b) the video 01006 which exactly belongs to category 10. These two videos are similar in the ultimate posture.

B. Frame unification of input videos

After an analysis of the wrongly classified videos given by the original C3D model, we find one factor hampering the accuracy is that the frames of input videos are clipped to meet the demand of C3D input so that some of the motion details are lost. As shown in Fig.3, the video 03138 is given the same label with video 01006, which belongs to the category 10. These two videos are really similar in the final posture (four fingers stretched without occlusion), so the motion path is important for distinguishing.

After analyzing the frame numbers of all 35878 training videos, we find they distribute like Fig.4. The frame numbers range from 1 to 405, in which most videos are with 29-39 frames and the peak is 33-frame - 1202 videos have 33 frames. For easier processing, we choose 32 as a benchmark frame number and unify all the videos with it. Videos that have frames more than 32 are sampled with the dynamic ratio according to their numbers of frames, while videos with frames less than 32 are extended by interpolation. Then over 98% videos are sampled at least per 3 frames, which guarantees most information of motion path is available for the extraction and thus distinguishing gestures like 03138 and 01006 in Fig.3 becomes easier.

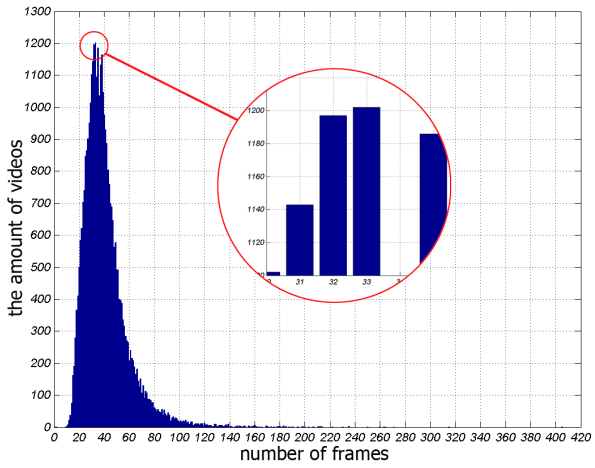


Fig. 4. The distribution of frame numbers of training videos. The peak value is 1202 of 33-frame video, and there are very few videos containing frames less than 10 or more than 100.

C. Fusion scheme

Fig. 5. The video-level fusion result. Since the object size in RGB and depth video is not the same, the fusion video of RGB and depth may result in unreasonable input.

Tran *et al.* [18] use 3 different nets and combine the features extracted by them to boost the accuracy. That inspires us to employ a fusion scheme for better performance. As RGB and depth videos are both available, we try to blend them together to benefit from both of them. Although RGB and depth video are obtained concurrently, the objects in them are with different sizes. As can be seen in Fig.5, the fusion in video-level may result in videos that make no sense. Meanwhile, the cost of frame-by-frame registration for these two kinds of videos is high. Therefore we process the fusion scheme in the later stage in feature-level, i.e., blend the features rather than videos of RGB and depth data. The features are abstracts of videos, which are the best illustrations of the character of gestures. Furthermore, all the features are with the same dimension, thus the fusion at this stage is reasonable. We

have two strategies for fusion: One is averaging the features, the other is integrating them to obtain a feature with higher dimension. The final classification is conducted by a linear SVM classifier with the blended feature, which is implemented by libsvm, a toolkit developed by Chang and Lin [23]. The results of these strategies will be given in the next section.

IV. Experiment Results

In this section, we demonstrate how our strategies work by groups of experiments. First, we discuss the parameter setting in our finetuning stage in Section IV-A, and then the experiments that show the effect of the unification of 32-frame strategy, different fusion schemes, and the comparisons among the baseline method, original C3D method and our method are given in Section IV-B to IV-D. The runtime analysis is illustrated in the last subsection. As the testing label is unavailable, all the experiments are conducted on the validation subset of the Chalearn LAP IsoGD database.

A. Parameter setting

As the model we use for feature extraction is C3D, the common network settings are the same as [18]. However, for a better finetuning result and adapting to our experiment environment, we use mini-batches of 10 clips, with the initial learning rate of 0.0001. The learning rate is reduced to the 0.9 times of antecedent after about 1.5 epochs (5000 batches). The finetuning process stops after about 28 epochs (10000 batches).

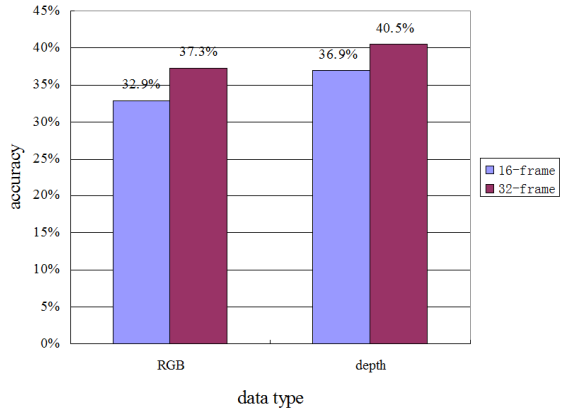


Fig. 6. Result comparison between the C3D model with 16-frame and 32-frame input. The 32-frame input is superior to the other one with both RGB and depth data on the validation subset of the Chalearn LAP IsoGD database.

B. The effect of 32-frame strategy

In this subsection, we verify the effectiveness of our 32-frame strategy. We compare the accuracy of classification with either 16-frame or 32-frame input. Both inputs are finetuned with 28 epochs, and the results are shown in Fig.6.

Fig.6 reports that compared with the original 16-frame strategy, our 32-frame strategy is more reasonable and achieves a great promotion on both RGB and depth data at about 4%.

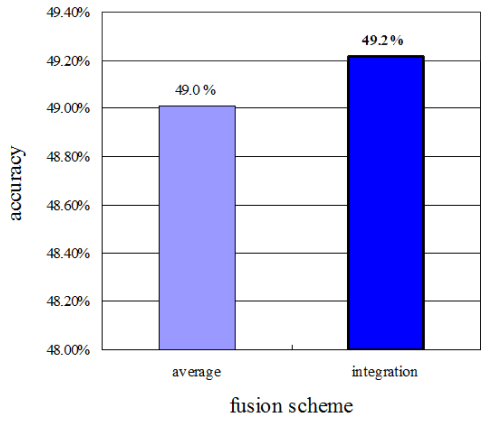


Fig. 7. Result comparison between different fusion schemes on the validation subset of the Chalearn LAP IsoGD database. The integration is better since the advantages of both of single features can be combined in the fusion feature.

C. Feature fusion result

We have two schemes for fusion as aforementioned: averaging the two features and integrating them to obtain a feature with higher dimension. In this subsection, we denote them as average and integration for simplicity. To make a fair comparison, all the other factors, such as the frame numbers of input videos and feature extraction parameters are all the same. The results of those schemes on 32-frame data are shown in Fig.7.

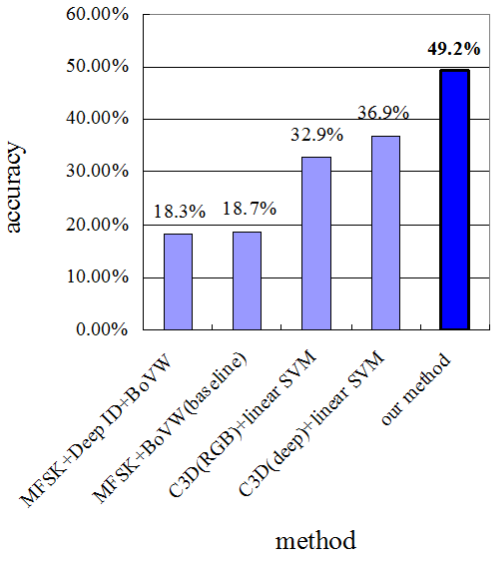


Fig. 8. Result comparison between different fusion schemes. All the results are given with data of the validation subset of the Chalearn LAP IsoGD database. It is apparent that our method far outperforms the others.

There is no doubt that the improvements on performance of both of fusion schemes are significant compared with any single feature as indicated in Table I. Although the differences among those schemes are slight, the integration scheme has the best performance as the advantages of all input features can be taken, while the other is more like striking a balance

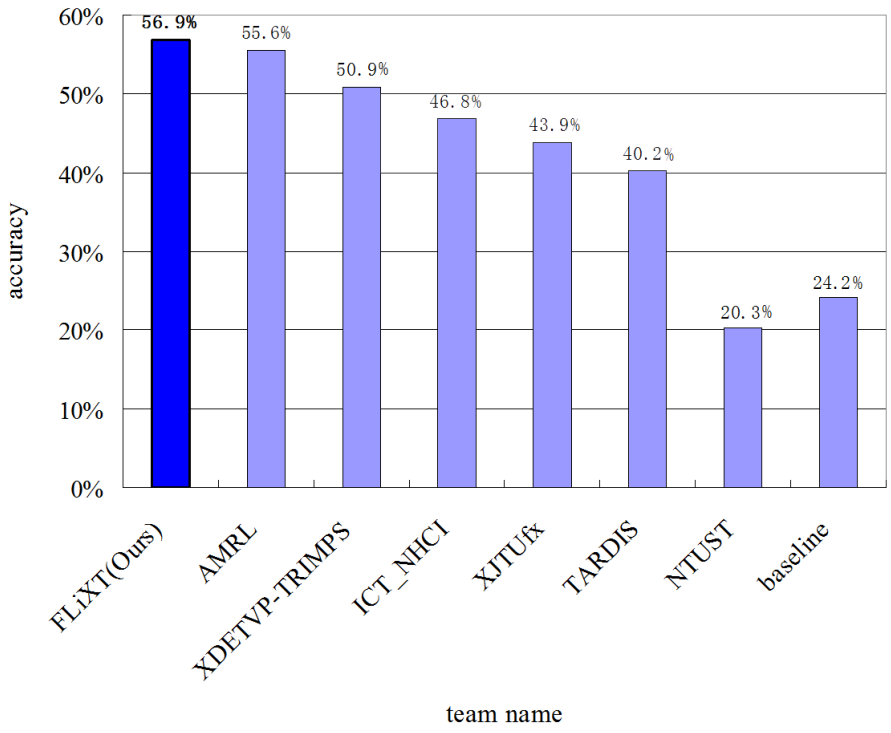


Fig. 9. The top 7 results and baseline of the challenge.

TABLE I. RESULT COMPARISON AMONG SINGLE FEATURES AND DIFFERENT FUSION SCHEMES

data	RGB	depth	**fusion-average**	**fusion-integration**
accuracy	37.3%	40.5%	**49.0%**	**49.2%**

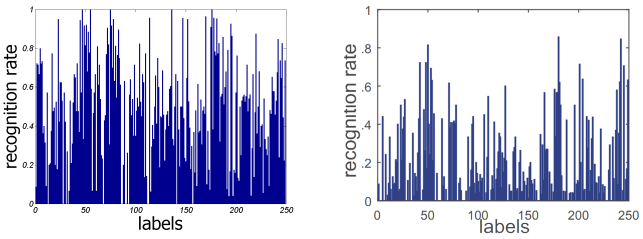
among those features, which may weak the benefits of fusion.

D. The final comparison

We illustrate the final comparison with the baseline method and original C3D result in Fig.8.

The results shows that compared with hand-crafted features, the deep learning is more promising to extract features and our approach can learn the details of gestures more clearly, and correspondingly yields better result to win in the challenge. The final score of top 7 results, including [24] and [25] (which wins the second and third place) and the baseline method [2] on testing data are shown in Fig.9.

We then analyze the result of each category of gestures as well. we take the result of integration scheme on validation dataset as an example, and as shown in Fig.10, almost each category evidences a great improvement on the baseline method, and some categories like 34, 57 and 61 are even fully recognized. We also reduce the number of unrecognized categories from 70 as reported in [2] to 14. These gestures are really hard to recognize, for example the gesture 13 with the performer stretching the single little finger, is always confused with the gesture 97, in which the performer stretches single index finger. Therefore more elaborate features may be required to deal with such a subtle difference.



(a) (b)

Fig. 10. The comparison between recognition rate of our method and baseline method for each category. The horizonal axis indicates the category numbers ranging from 1 to 249, while the vertical axis indicates the rate of correctly recognized gestures of each category. (a) Our recognition rate. (b) The recognition rate of baseline method (from Wan *et al.*'s paper [2]). It is obvious our approach outperforms the baseline method in many categories. However, like the baseline one, ours is also failed for some hard-to-recognize categories.

E. Runtime analysis

Our experiments are processed on a PC with Intel Core i7-6700 CPU @ 3.40GHz × 8, 16 GB RAM and Nvidia Geforce GTX TITAN X GPU. The experiments of the C3D model training and feature extracting are processed under caffe framework on Linux Ubuntu 14.04 LTS, others including 32-frame video generation, feature fusion and SVM classification are implemented by matlab R2012b on 64-bit Windows 7.

Thanks to the quickness of C3D implementation, the processing of feature extraction can reach 656 fps. For the SVM classification, since it runs only with CPU, it reaches about 3 - 5 seconds to finish the whole training and classifying process

for each video, which depends on the amount of training data and the rate that CPU is occupied.

V. Conclusion

In this paper, we propose a gesture recognition method with both RGB and depth data on a 3D convolutional network. We first convert the input data into 32-frame videos to learn the details of motion better, then the features of RGB and depth videos are extracted from the C3D model respectively and blended together to boost the performance. The final classification is implemented with a linear SVM classifier. The experiments verify the effectiveness of our strategy.

However, there are still many factors that affect the accuracy of recognition. The skin color, clothes of performers may disturb the gesture recognition. Therefore how to weak the influence of those gesture-irrelevant factors is what we intend to study in the future. Meanwhile, the other deep learning architectures, like the deep belief networks, also show great promising for object recognition. The extension of these architectures to video based gesture recognition may be also worthy to be studied.

Acknowledgment

The work was jointly supported by the National Natural Science Foundations of China under grant No. 61472302, 61272280, U1404620, and 41271447; The Open Projects Program of National Laboratory of Pattern Recognition(201600031): The Program for New Century Excellent Talents in University under grant No. NCET-12-0919; The Fundamental Research Funds for the Central Universities under grant No. K5051203020, K5051303018, JB150313, JB150317, and BDY081422; Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase)Natural Science Foundation of Shaanxi Province, under grant No. 2010JM8027: The Creative Project of the Science and Technology State of Xi'an under grant No. CXY1441(1); The State Key Laboratory of Geo-information Engineering under grant No.SKLGIE2014-M-4-4; The International Exchange Program of Xidian University Graduate Innovation Fund.

References

[1] H. Escalante, L. Ponce-López, L. Wan, M. A. Riegler, B. Chen, A. Clapés, S. Escalera, I. Guyon, X. Baró, P. Halvorsen, H. Möller, and M. Larson, "Chalearn joint contest on multimedia challenges beyond visual analysis: An overview," in *ICPR Workshop*, 2016.

[2] J. Wan, S. Z. Li, Y. Zhao, S. Zhou, I. Guyon, and S. Escalera, "Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition," in *IEEE CVPR Workshop*, 2016.

[3] J. Konecnỳ and M. Hagara, "One-shot-learning gesture recognition using hog-hof," *Journal of Machine Learning Research*, vol. 15, pp. 2513–2532, 2014.

[4] M. R. Malgireddy, I. Inwogu, and V. Govindaraju, "A temporal bayesian model for classifying, detecting and localizing activities in video sequences," in *IEEE CVPR Workshops*. IEEE, 2012, pp. 43–48.

[5] M. R. Malgireddy, I. Nwogu, and V. Govindaraju, "Language-motivated approaches to action recognition." *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2189–2212, 2013.

[6] M. Yeasin and S. Chaudhuri, "Visual understanding of dynamic hand gestures," *Pattern Recognition*, vol. 33, no. 11, pp. 1805–1817, 2000.

[7] P. Hong, M. Turk, and T. S. Huang, "Gesture modeling and recognition using finite state machines," in *IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, 2000, pp. 410–415.

[8] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1992, pp. 379–385.

[9] H. J. Escalante, I. Guyon, V. Athitsos, P. Jangyodsuk, and J. Wan, "Principal motion components for one-shot gesture recognition," *Pattern Analysis and Applications*, pp. 1–16, 2015.

[10] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *BMVC 19th British Machine Vision Conference*. British Machine Vision Association, 2008, pp. 1–10.

[11] J. Wan, Q. Ruan, W. Li, and S. Deng, "One-shot learning gesture recognition from rgb-d data using bag of features." *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2549–2582, 2013.

[12] J. Wan, Q. Ruan, W. Li, G. An, and R. Zhao, "3d smosift: three-dimensional sparse motion scale invariant feature transform for activity recognition from rgb-d videos," *Journal of Electronic Imaging*, vol. 23, no. 2, pp. 3017–3017, 2014.

[13] J. Wan, G. Guo, and S. Li, "Explore efficient local features from rgb-d data for one-shot learning gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1626–1639, 2015.

[14] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 3361–3368.

[15] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[16] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[17] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[18] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *IEEE International Conference on Computer Vision*. IEEE, 2015, pp. 4489–4497.

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[20] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 4207–4215.

[21] H. Yang, B. Wang, S. Lin, D. Wipf, M. Guo, and B. Guo, "Unsupervised extraction of video highlights via robust recurrent auto-encoders," in *IEEE International Conference on Computer Vision*, 2015, pp. 4633–4641.

[22] I. Guyon, V. Athitsos, P. Jangyodsuk, and H. J. Escalante, "The chalearn gesture dataset (cgd 2011)," *Machine Vision and Applications*, vol. 25, no. 8, pp. 1929–1951, 2014.

[23] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 1–27, 2011.

[24] P. Wang, W. Li, S. Liu, Z. Gao, C. Tang, and P. Ogunbona, "Large-scale isolated gesture recognition using convolutional neural networks," in *ICPR Workshop*, 2016.

[25] G. Zhu, L. Zhang, L. Mei, J. Shao, J. Song, and P. Shen, "Large-scale isolated gesture recognition using pyramidal 3d convolutional networks," in *ICPR Workshop*, 2016.