Fusion of Deep Learning Descriptors for Gesture Recognition

Edwin Escobedo Cardenas(✉) and Guillermo Camara-Chavez

Federal University of Ouro Preto, Ouro Preto, MG, Brazil
edu.escobedo88@gmail.com, gcamarac@gmail.com

Abstract. In this paper, we propose an approach for dynamic hand gesture recognition, which exploits depth and skeleton joint data captured by Kinect™ sensor. Also, we select the most relevant points in the hand trajectory with our proposed method to extract keyframes, reducing the processing time in a video. In addition, this approach combines pose and motion information of a dynamic hand gesture, taking advantage of the transfer learning property of CNNs. First, we use the optical flow method to generate a flow image for each keyframe, next we extract the pose and motion information using two pre-trained CNNs: a CNN-flow for flow-images and a CNN-pose for depth-images. Finally, we analyze different schemes to fusion both informations in order to achieve the best method. The proposed approach was evaluated in different datasets, achieving promising results compared to other methods, outperforming state-of-the-art methods.

Keywords: Keyframe extraction · Hand gesture recognition
Pose and motion information · Convolutional neuronal networks
Fusion methods

1 Introduction

Actually, with greatest technological advance, more emphasis is being placed on non-verbal communication due to its speed and expressiveness in interaction, it is performed through gestures involving the gesture recognition area, which is one of the main component in the recent research field of human computer interaction (*HCI*) and recognized as a valuable technology for several applications due to its potential in areas such as video surveillance, robotics, multimedia video retrieval, etc. [9,13]. Hand gestures are one of the most common categories of gesture recognition used for communication and interaction. Furthermore, hand gesture recognition is seen as a first step towards sign language recognition, where each little difference in motion or hand configuration can change completely the meaning of a sign. So, the recognition of fine-grained hand movements represent a major research challenge.

Hand-crafted spatio-temporal features were widely used in gesture recognition [19]. Many vision-based algorithms were introduced to recognize dynamics

M. Mendoza and S. Velastín (Eds.): CIARP 2017, LNCS 10657, pp. 212–219, 2018.
https://doi.org/10.1007/978-3-319-75193-1_26

hand gestures [9,14]. Others well-known feature detection methods used were HOG/HOF [11], HOG3D [10], SIFT [12], etc. In the same way, to exploit the trajectory information, Shin et al. [17] proposed a geometric method using Bezier curves. Escobedo *et al.* [3,4] proposed convert the trajectory to spherical coordinates to describe the spatial and temporal information of the movements and to avoid problems with the user position changes.

Recent studies have demonstrated the power of deep convolutional neural networks(CNNs), it become an effective approach for extracting high-level features from data [16]. Wu *et al.* [21] proposed a novel method called Deep Dynamic Neural Networks for multimodal gesture recognition using deep neural nets to automatically extract relevant information from the data, they integrated two distinct feature learning methods, one for processing skeleton features and the other for RGB-D data. Besides, they used a feature learning model with a HMM to incorporate temporal dependencies. An interesting property of the CNNs, is the transfer of pre-trained network parameters to problems with limited training data, this has show success in different computer vision areas, achieving equal or better results compared to the state-of-the-art methods [15].

Based on the previous study, we propose a dynamic hand gesture recognition approach combining motion and pose information computed from depth and skeleton data captured by a Kinect^{TM}device. In contrast to previous works, we use the method proposed in [3] to extract keyframes, this method exploits the spatial information of both arms, detecting the dominant hand. This method analyses the *3D* trajectory skeleton to detect points which represent frames with more differentiated pose. As we have a fixed keyframe number, the proposed method becomes independent of the repeated use of time series techniques as Hidden Markov Model (HMM) or Dynamic Time Warping (DTW), reducing the processing time for a gesture. Finally, in this paper we investigate different fusion methods for human pose and motion features computed from two pre-trained CNNs, the hand user posture together with the motion information are decisive to determine a good classification. We report experimental results for different datasets composed of a set of fine-grained gestures.

The remainder of this paper is organized as follows. In Sect. 2, we describe and detail our proposed hand gesture recognition system. Experiments and Results are presented in Sect. 3. The Conclusions and future works are presented in Sect. 4.

2 Method Overview

Our approach consists of four main stages, as shown in Fig. 1. In the first stage, hand gesture information is captured by Kinect^{TM}device. In the second stage, we preprocess the trajectory information to obtain the keyframes. In the third stage, we compute motion and pose features using two CNNs and finally, these features are fused by different methods to generate a unique feature vector, which is used as an input into our classifier.

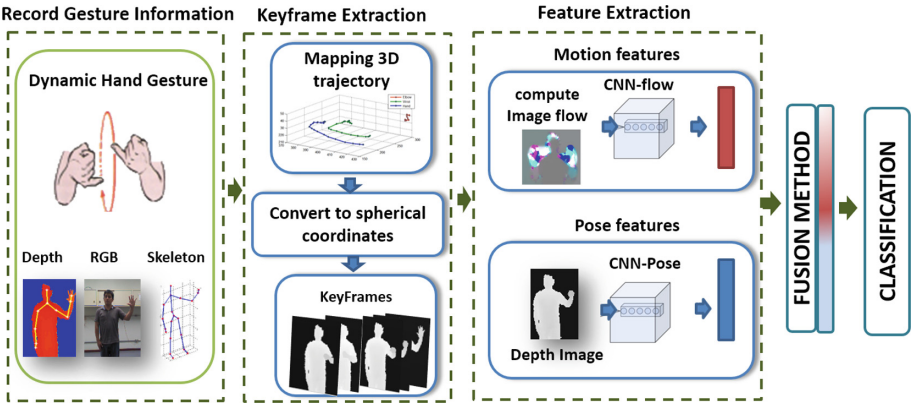

Fig. 1. Hand Gesture Recognition proposed model.

2.1 Record Gesture Information

The Kinect^{TM} sensor *V1* was used to capture hand gestures. The key of gesture recognition success is the depth camera device, which consists of an infrared laser projector and an infrared video camera. Furthermore, this device provides intensity and depth data, and Cartesian coordinates of 20 human body skeleton joints.

2.2 Keyframe Extraction Method

One of the principal challenges in hand gesture video analysis is the time variability that arises when every user makes a gesture with different speeds. Work with all frames is inefficient and take a long time, so it is necessary to choose some keyframes. Therefore in this approach, we used the method proposed in [3], which is an improve of [4] to extract these keyframes. These make our hand gesture recognition system invariant to temporal variations and avoid the repeated use of time series techniques.

2.3 Feature Extraction

A dynamic hand gesture has two essential parts: the body pose information and its motion. According to this, we further borrow inspiration from [2,5] to represent our dynamic hand gesture approach. Unlike [2], we do not create body regions since this step is unnecessary and only increases the processing time. Another difference is that we use depth image keyframes instead of the RGB, avoiding illumination changes and complex background interference. To generate our final feature vector, we first compute optical flow. For that, we apply the method used in [6] for each consecutive pair of keyframes. According to [2], the values of the motion field v_x, v_y are transformed to the interval $[0; 255]$ by $\tilde{v}_{x|y} = av_{x|y} + b$ where $a = 16$ and $b = 128$. The values below 0 and above

255 are truncated. We save the transformed flow maps as images with three channels corresponding to motion $\tilde{v}_x, \tilde{v}_y$ and the flow magnitude. So far, we have N keyframes and $N-1$ flow images. Figure 2 shows an example from a dynamic gesture with its N keyframes and its corresponding flow images.

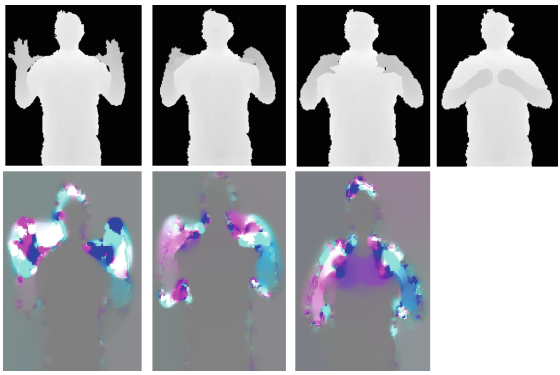

Fig. 2. Example from a dynamic gesture with its $N = 4$ keyframes and its corresponding $N-1$ flow images.

Now, given a keyframe and its corresponding flow image, we use two distinct CNNs to compute pose and motion features. Both networks contain 5 convolutional and 3 fully-connected layers. The output of the second fully-connected layer with $k = 4096$ values is used as a keyframe descriptor.

For depth images, we use the publicly available *imagenet-vgg-f* network from [1] (our CNN-pose). For flow images, we use the motion network provided by [6] (CNN-flow). Computing at the end, two vector of $N \times 4096$ and $(N-1) \times 4096$ to be fused.

2.4 Fusion Methods

Finally, we follow the ideas proposed in [2,5]. We considered and analyzed different schemes for fusing both informations. We distinguished two different fusion methods: direct fusion and fusion by aggregation.

Fusion by Aggregation. Here, the two vectors do not need to have the same dimensions, since individual information is extracted separately through a function (max, mean, max-min) per column. At the end, both are concatenated in a single vector.

$$y = cat(f(x^a), f(x^b)) \tag{1}$$

where f can be a max, mean or max-min operator and cat is the concatenation operator.

Direct Fusion. In this case, both vector maps need to have the same size, because the function is applied in both vectors at the same time, following next rules:

$$y_{ij} = f(x^a_{ij}, x^b_{ij}) \tag{2}$$

where f can be a max, mean, sum or concatenation operator.

3 Experiments and Results

In our experiments we use three datasets: the UTD-MHAD [7] which contains 27 human actions performed by eight subjects, the Brazilian Sign Language - LIBRAS [4] which contains 20 gestures performed by two subjects and the recently SHREC-2017 [18] which contains sequences of 14 hand gestures performed in two ways: using one finger and the whole hand. Each gesture was performed between 1 and 10 times by 28 participants, resulting in 2800 sequences. Each dataset provides a different protocol to make the classification stage. Our experiments follow these specifications.

3.1 Fusion Methods Evaluation

To demonstrated the utility of the keyframe extraction process, we conducted two previous experiments. We randomly select 10 signals from the LIBRAS test dataset, we measure its processing time by using the keyframe extraction method and without it, *i.e.*, working with all frames. The results are shown in Fig. 3, where we can observe the processing time is almost constant in the 10 random gestures. In contrast, when we work with all frames, the processing time varies according to the frame number, thus we show that the process of keyframe extraction accelerates the execution time, remaining almost constant.

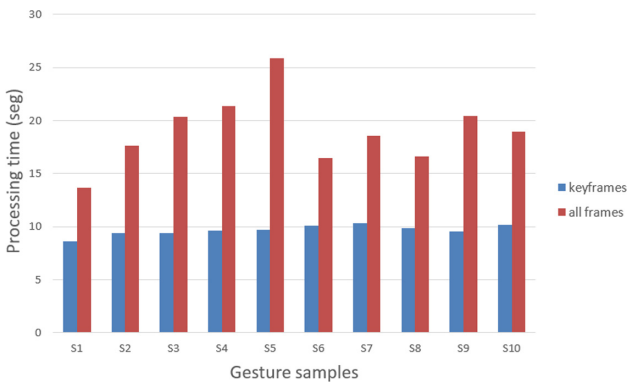

Fig. 3. Processing time recording to compare ten random S_i gestures from the LIBRAS dataset. Using the keyframe extraction algorithm we can wee that the processing time is almost constant.

We also make a comparison by measuring the overall performance obtained from the test dataset. The results are shown in Table 1, we observe that the results are almost similar for both cases, presenting a low standard deviation (SD). Thus, the performance is not significantly affected, we demonstrate the robustness of the keyframe extraction method used in our approach.

Table 1. Performance comparison between our approach to use the keyframe extraction algorithm and using all frames. We compared using different methods of fusion by aggregation. Results show that the difference is minimum.

Fusion method	Using keyframes	Using all frames	SD
Aggregation min-max	98.25	98.50	**0.18**
Aggregation max	95.99	96.25	**0.18**
Aggregation mean	96.74	97.50	**0.54**
SD	**1.15**	**1.13**	

After demonstrating the robustness of our approach, we conducted a set of experiments to find the best fusion method of flow and pose features, thereby we divided the experiments into two parts: first, we conducted experiments using fusion by aggregation, where we used the max, min and max-min operators. The second experiment was conducted by direct fusion. It is worth highlighting that these experiments can be performed only on keyframes, since this fusion method requires to have vectors with fixed size. The experiments were performed using the max, min, sum, mean and concatenation operators. All experiments were performed on the previously mentioned datasets. The results obtained are shown in Table 2 and we observe that in some cases the fusion by aggregation presents better results on UTD-MHAD and SHERC-2017 datasets, using the max-min operator. In the case of the LIBRAS dataset, the direct fusion presents better results when applying the max operator. This can be due to the LIBRAS dataset is formed from a set of sign language gestures with structured movements.

Table 2. Summary of our experiments using different schemes of fusion. The table shows the results in three different datasets.

	Fusion methods						
	Aggregation			Direct			
Dataset	Max	Mean	Max-min	Sum	Max	Mean	Concat
UTD-MHAD	93.54	93.73	**94.65**	91.30	94.63	92.49	93.75
SHREC-14	75.68	75.79	**75.97**	74.82	75.69	73.63	74.10
LIBRAS	96.74	95.99	98.25	97.74	**98.50**	96.24	98.00

Finally, we compared our best results with other methods that used the LIBRAS and UTD-MHAD datasets. Table 3 shows the results obtained and it is

deduced that the fusion of motion and pose features represent better a gesture. Here, emerge the importance of the fusion of features by searching a method that best suits the problem.

Table 3. Comparison of the our approach with other method in the literature

Method	UTD -MHAD	LIBRAS
Escobedo and Camara [3]	84.89	98.54
Chéron et al. [2] (Only Kinect data)	66.10	-
Wang et al. [20]	85.81	-
Hou et al. [8]	86.97	-
Aggregation (max-min)	**94.65**	**98.25**
Direct fusion (max)	**94.63**	**98.50**

4 Conclusion

In this paper, we propose a new approach that fuses pose and motion features of a dynamic hand gesture. We exploited the transfer learning property from CNNs by extracting information from two pre-trained models. We conclude that the fusion stage is very important and decisive to the correct performance of our method. Another important issue is the definition of a method to extract keyframes, the processing time is considerably decrease when the keyframe extraction process is used, without affecting the performance of our approach. This is a very important characteristic when applied in real time applications.

Finally, the robustness of our method was shown in the experiments when compared with other methods of the literature. As future works, we propose to exploit new fusion methods, research on new $3D$ CNN architectures to extract best features from depth images and apply it on more hand gesture datasets.

Acknowledgements. The authors thank UFOP and Pro-Rectory of Research and Post-Graduation (PROPP).

References

1. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: delving deep into convolutional nets. arXiv preprint arXiv:1405.3531 (2014)
2. Chéron, G., Laptev, I., Schmid, C.: P-CNN: pose-based CNN features for action recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3218–3226 (2015)
3. Escobedo, E., Camara, G.: A new approach for dynamic gesture recognition using skeleton trajectory representation and histograms of cumulative magnitudes. In: 2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 209–216. IEEE (2016)

4. Escobedo-Cardenas, E., Camara-Chavez, G.: A robust gesture recognition using hand local data and skeleton trajectory. In: 2015 IEEE International Conference on Image Processing (ICIP), pp. 1240–1244. IEEE (2015)
5. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1933–1941 (2016)
6. Gkioxari, G., Malik, J.: Finding action tubes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 759–768 (2015)
7. Han, F., Reily, B., Hoff, W., Zhang, H.: Space-time representation of people based on 3D skeletal data: a review. Comput. Vis. Image Underst. **158**, 85–105 (2017)
8. Hou, Y., Li, Z., Wang, P., Li, W.: Skeleton optical spectra based action recognition using convolutional neural networks. IEEE Trans. Circuits Syst. Video Technol. (2016)
9. Kim, D., Hilliges, O.D., Izadi, S., Olivier, P.L., Shotton, J.D.J., Kohli, P., Molyneaux, D.G., Hodges, S.E., Fitzgibbon, A.W.: Gesture recognition techniques. US Patent 9,372,544, 21 June 2016
10. Klaser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3D-gradients. In: 2008–19th British Machine Vision Conference on BMVC, p. 275:1. British Machine Vision Association (2008)
11. Laptev, I.: On space-time interest points. Int. J. Comput. Vis. **64**(2/3), 107–123 (2005)
12. Lowe, D.G.: Object recognition from local scale-invariant features. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157. IEEE (1999)
13. Pisharady, P.K., Saerbeck, M.: Recent methods and databases in vision-based hand gesture recognition: a review. Comput. Vis. Image Underst. **141**, 152–165 (2015)
14. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**(1), 1–54 (2015)
15. Sharif Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 806–813 (2014)
16. Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M.: Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans. Med. Imaging **35**(5), 1285–1298 (2016)
17. Shin, M.C., Tsap, L.V., Goldgof, D.B.: Gesture recognition using Bezier curves for visualization navigation from registered 3-D data. Pattern Recogn. **37**(5), 1011–1024 (2004)
18. SHREC-2017: 3D hand gesture recognition using a depth and skeletal dataset (2017). http://www-rech.telecom-lille.fr/shrec2017-hand/
19. Trindade, P., Lobo, J., Barreto, J.P.: Hand gesture recognition using color and depth images enhanced with hand angular pose data. In: 2012 IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 71–76. IEEE (2012)
20. Wang, P., Li, Z., Hou, Y., Li, W.: Action recognition based on joint trajectory maps using convolutional neural networks. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 102–106. ACM (2016)
21. Wu, D., Pigou, L., Kindermans, P.J., Nam, L., Shao, L., Dambre, J., Odobez, J.M.: Deep dynamic neural networks for multimodal gesture segmentation and recognition (2016)