MINISTERUL EDUCAȚIEI NAȚIONALE
ROMÂNIA
UNIVERSITATEA DE MEDICINĂ, FARMACIE, ȘTIINȚE ȘI TEHNOLOGIE „GEORGE EMIL PALADE" DIN TÂRGU MUREȘ

sciendo

Acta Marisiensis. Seria Technologica
Vol. 17 (XXXIV) no. 1, 2020
ISSN 2668-4217, ISSN-L 2668-4217

10.2478/amset-2020-0005

AN OPTICAL FLOW-BASED GESTURE RECOGNITION METHOD

Dániel Zoltán NAGY[1], Imre PILLER[2]

[1,2] *Mathematical Institute, University of Miskolc*
3515 Miskolc-Egyetemváros, Miskolc, Hungary

[1]nagy118@iit.uni-miskolc.hu
[2]imre.piller@uni-miskolc.hu

Abstract

The efficient human-machine interaction is an essential and current problem of computer science. The paper presents a gesture recognition method which applies optical flow calculation and an aggregation for obtaining a heatmap-like representation of the motion trajectories. After the overview of the image processing workflow, the paper introduces six symbols for providing some measurements. The described experiments show the robustness of the method against color, shape and time variance.

Key words: gesture recognition, optical flow, feature extraction, classification, human computer interaction

1. Introduction

The research of the proper human-computer interaction is one of the fundamental problems of computer science. The motivation behind them is to find the most natural way of the communication with the help of machines. The increasing number of Augmented Reality and Virtual Reality applications signs the relevance and actuality of the presented topic. Furthermore, many publications emphasize the importance of hand gesture recognition [1, 9, 10].

The goal of this paper is to provide an alternative method for solving the symbol recognition problem where the input is a video stream. In the considered scenario the user draws various symbols into the air in front of the camera. (Many researches use depth images as an input [3, 12, 15].) There is a set of predefined symbols. We have to provide an image processing method and a classification algorithm which can guess the symbol with sufficient accuracy.

The proposed symbol recognition method uses motion flow information for estimating the recently drawn symbols. Compared to the classical object tracking methods it has the following advantages and disadvantages.

The main benefit of this approach is the shape and texture invariance. Most of the available methods combines shape detection and object tracking techniques [4, 5, 14]. Therefore, the reliability of the symbol recognition depends on the accuracy of these two different steps.

An object detection and a pose estimation method can provide detailed information about the tracked object [7, 11]. In our use case these informations are ignored. It means that, our system is unable to differentiate between a human hand and an arbitrary pointer device on the images.

2. Video Processing Method

The processing of the video stream is a multi-step process. The video stream is the sequence of coloured images. We call these images as frames. The format and the resolution of all frames are the same. The elapsed time between two sequential frames can be considered as a constant value. Let see the overview of

the proposed method in Figure 1.

We consider only that use cases, where the hue of objects on the images do not provide significant additional information in the aspect of gesture recognition. Our experiment suggest that the usage of grayscale images is sufficient. There are different methods for the grayscale convertion. The selection of them does not play key role in the further processing steps. In Figure 1 the grayscale frames represent this processing step.

We would like to reduce the effect of light condition changes. Some of the video capturing devices have capability to adapt to them. It helps to improve the image quality in the typical use case, but also means that any local intensity changes affect intensity changes globally. Fortunately, the estimation of motion vectors will eliminate this kind of noise automatically.

On sequential frames we can observe that some regions of the image are changing their positions. It does not necessarily match with the position changes of the real objects of the captured scene. Theoretically, it is possible to estimate the motion vector for any point of a frame. Therefore, it results a vector field on the domain of the image.

The presented gesture recognition method based on the estimation of these motion vectors. The following paragraphs mention the reasons which makes the estimation task difficult.

The motion is not a well-defined term in the topic. It is an intuitive concept without a strict mathematical foundation. The estimation of a motion vector at a point requires the current frame and at least the previous one. On the analogy of interpolation methods, we could take into account more preceding frames. Expectedly, the larger count of frames are unable to cause significantly better precision.

The frames are rasterized images. In our case it is enough to estimate the motion vectors only in some specific positions. We have to select the tracked parts of the image. All parts should have exactly the same shape and size. The segmentation of the image by a squared, equidistant grid is appropriate. Moreover, it provides a more convenient calculation scheme. The grid size is a free parameter of the method.

The OpenCV library contains the implementation of an optical flow method of Lucas-Kanade [2]. In our configuration

- the window size is 50×50,
- the maximal level of the image pyramid is 2,
- the iterative search process termination criteria is $\varepsilon 0.03$ and
- the maximal count of iteration steps is 10.

The resulted motion vectors can be represented on the domain of the frames as scaled vectors from the predefined grid points (Figure 1, Motion vectors frame).

The implementation details of the optical flow algorithm and the tuning of proper parametrization is out of the scope of our current research.

The estimated motion vectors describe the drawed symbol between two sequential frames. For higher level view we have to aggregate the vectors. A straightforward method of the aggregation is the usage of matrix which resembles to a heatmap. (We can found similar approaches in the literature [6, 8].)

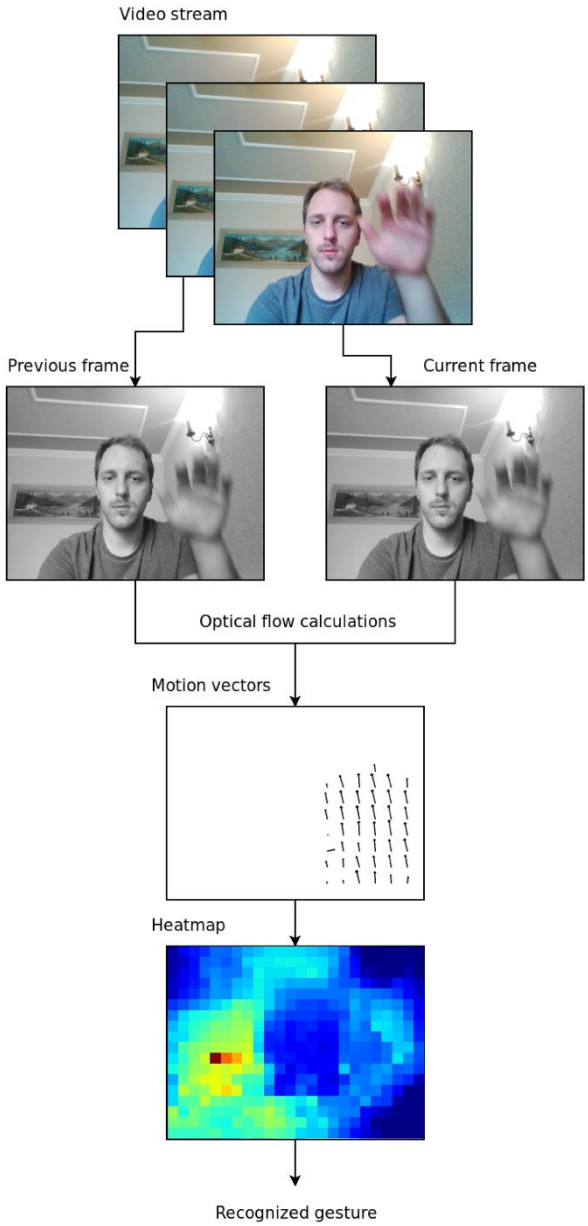

Fig. 1: The main steps of the presented image processing method

Let consider only the measure of the motions which is available as the lengths of motion vectors. The aggregation of these values in a specific time interval results a matrix. It is suitable as the features of the analyzed gesture.

The length of the drawn symbols (in time) can be differ significantly. We can apply temporal filters for solving this issue. In its simplest form it is able to recognize the time points where a gesture has finished and a new has started. In this way, after a successful

time segment detection the symbol recognition task becomes the classical pattern recognition problem.

We define the ideal patterns in the resolution of the grid. We can estimate them from samples and use as references for classification. For instance, by calculating the mean of element wise squared distances, the symbol of the closest reference matrix will provide the recognized gesture.

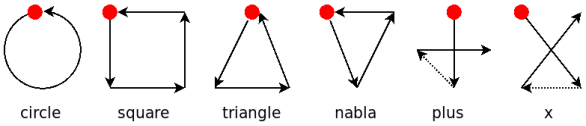

Fig. 2: Set of the selected symbols for gesture recognition

3. Results

For checking the effectiveness of the proposed method, we have defined six symbols (see Figure 2). The red dots note the starting point of the symbols. We have recorded six videos with 9-9 occurrences of them. (These samples created by two presenters. There are 4 samples from the first and 5 samples from the second presenter for each symbols.)

We have applied an equidistant orthogonal grid with 18 rows and 24 columns. Therefore, the matrices of the motion vector lengths and the heatmaps are 18×24 sized.

In Figure 3 we can see the measure of the average motion between the frames of the samples. The average motion estimated between any two sequential frames of the video as the mean of the lengths of the motion vectors at the grid points. (The red lines in Figure 3 separates the samples of the first and the second presenter.)

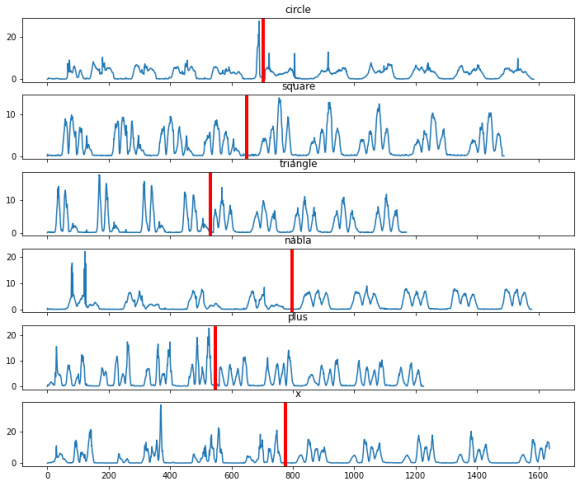

Fig. 3: Mean of the motion vector lengths in the six video samples

As we can see, the shape of the measures of the recorded gestures are reasonably similar. However, in particular cases (for example the first occurrence of the nabla symbol) there are some peaks. Instead of filtering this kind of noise, our method aggregates the vector lengths the specific time intervals.

The gesture recognition algorithm has to find the start and finish time of the gestures. We used a simple, but fairly robust heuristic. We have defined a threshold value for the minimal amount of average motion (denoted by μ) and a threshold value for separating the gestures in time (denoted by τ). The algorithm classifies the heatmap when the mean of the motion vector lengths are lower than μ at least on τ number of sequential frames. (Naturally, we can use seconds instead of frame count.) We use μ=2 and τ=25 as experimental values. After, the heatmaps are obtained as the sum of motion vector lengths matrices on the segmented time intervals.

In the followings, we show two-two examples of the heatmaps at some video segments.

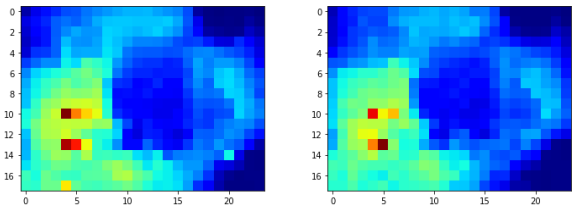

Fig. 4: Heatmaps of circle gestures

In Figure 4 we can see sample heatmaps of the circle gestures.

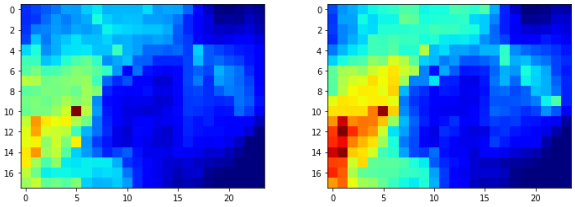

Fig. 5: Heatmaps of square gestures

These heatmaps do not provide the drawn symbols directly, because there are many incidental motions near the virtual drawing point.

Two heatmaps of the square are in Figure 5. They are slightly different than the heatmaps of circle gestures. We can expect it, because the circle and square symbols show similarities.

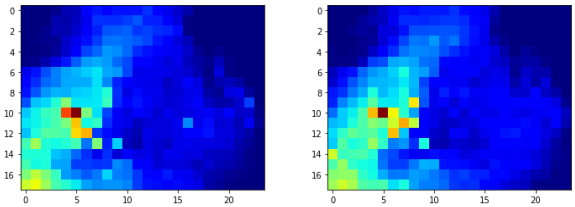

Fig. 6: Heatmaps of triangle gestures

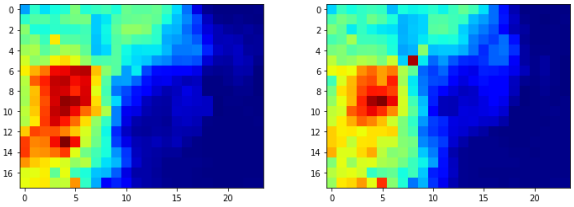

Fig. 7: Heatmaps of nabla gestures

The Figure 6 and Figure 7 present samples for the two types of triangular symbols. These results suggest that, the triangle and nabla symbols can be distinguished with high confidency regardless of the similarity of symbols.

The samples of plus and x gestures (in Figure 8 and Figure 9) reinforce that the selection of the symbol set was proper for the mentioned feature extraction method. However, the heatmaps do not resemble to the symbol of the gesture, but the resulted features can be distinguished visually and by machine also.

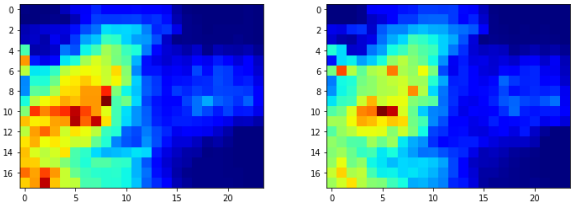

Fig. 8: Heatmaps of plus gestures

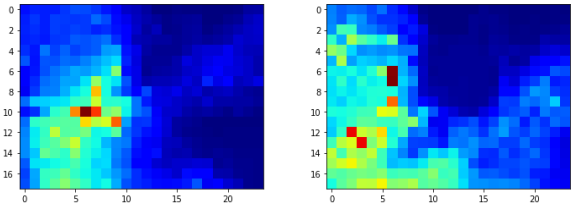

Fig. 9: Heatmaps of x gestures

For checking the reliability of the heatmaps for gesture recognition, we have chosen a simple classification method which predicts the corresponding symbol of the given heatmap. It calculates the euclidean distance of the normalized heatmaps and finds the closest from the available samples. After, the algorithm uses its symbol as the prediction.

The first sample set has 24 heatmaps (4-4 samples for the 6 symbols). In 21 cases the predictions were correct. In a case, it has predicted a triangle instead of a circle and in two other cases, it has predicted also triangles instead of x symbols.

The second sample set contains 30 heatmaps (5-5 samples for the symbols). In 28 cases the predictions were correct. In one case, the algorithm has predicted a nabla symbol instead of a square, and in an other failed case, it has predicted plus symbol instead of x symbol.

4. Discussion

The proposed method applies motion vectors as the primary features for gesture recognition. It makes unnecessary the classical image filtering and noise reduction steps. The parameters of the algorithms are intuitive and verifiable. In the presented samples we have obtained acceptable results by using experimental values.

The accuracy of the gesture recognition is highly depends on the selection of the symbols. We have chosen six symbols for experimentation. For larger number of symbols we have to measure the distance of the symbols in the feature space. (The study of proper symbol set selection is also can be a further research direction.) The separation of gestures has resolved by a measure and a time threshold. One of our assumption about the recognition problem is that the gestures are separated well. In other cases it worth to apply more complex models (for instance the dynamic time warping method [13]).

5. Conclusions

We have presented a gesture recognition method which uses motion flow calculations and heatmap-like aggregation for feature extraction. The method does not rely on the shapes and colors of the drawing object. By the proper selection of temporal filter the recognition can be invariant to the speed of symbol drawing.

The method calculates the heatmaps in the same way for any symbol. The feature extraction method is flexible, but the proper selection of symbol set remains the responsibility of the presenter.

The reliability of the gesture recognition depends on the homogeneity of the samples. The change of the scene or the presenter object can result significantly different heatmaps.

This kind of gesture recognition is suitable for human-machine interaction, for instance as the visual input processor of Augmented Reality applications and mobile robot control. The usage of the proposed method is beneficial, where the changing of light condition, the low video image resolution or the limited processing power cause difficulties for the widely-used approaches.

Acknowledgement

The described article/presentation/study was carried out as part of the EFOP-3.6.1-16-2016-00011 “Younger and Renewing University – Innovative Knowledge City – institutional development of the University of Miskolc aiming at intelligent specialisation” project implemented in the framework of the Szechenyi 2020 program. The realization of this project is supported by the European Union, co-financed by the European Social Fund.

References

[1] Arcangel, R., et al. (2017), *Robust Mouse Control based on Dynamic Template Matching of Hand Gestures*, The

International Congress for global Science and Technology.

[2] Baker, Simon, and Iain Matthews. (2004), *Lucas-kanade 20 years on: A unifying framework*, International journal of computer vision 56.3, pp. 221-255.

[3] Biswas, K. K., & Basu, S. K. (2011, December). *Gesture recognition using microsoft kinect®*. In The 5th international conference on automation, robotics and applications, pp. 100-103.

[4] Joseph, V., Talpade, A., Suvarna, N., & Mendonca, Z. (2018, June), *Visual gesture recognition for text writing in air*. In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 23-26).

[5] Kumar, S., Tanwar, M., Kumar, A., Rani, G., & Chamoli, P. (2019). *Hand Gesture Recognition Based Calculator*.

[6] Lazaridis, L., Dimou, A., & Daras, P. (2018, September), *Abnormal Behavior Detection in Crowded Scenes Using Density Heatmaps and Optical Flow*, 2018 26th European Signal Processing Conference (EUSIPCO) pp. 2060-2064.

[7] Lin, Zhe, Zhuolin Jiang, and Larry S. Davis. (2009*), Recognizing actions by shape-motion prototype trees*, 2009 IEEE 12th international conference on computer vision. IEEE, 2009.

[8] Pfister, T., Charles, J., & Zisserman, A. (2015), *Flowing convnets for human pose estimation in videos,* In Proceedings of the IEEE International Conference on Computer Vision, pp. 1913-1921.

[9] Purohit, Ayush, and Shardul Singh Chauhan. *A Precise Technique for Hand Gesture Recognition*, Graphics, Vision and Image Processing Journal, ISSN 1687-398X, Volume 17, Issue 2, ICGST LLC, Delaware, USA, Nov. 2017.

[10] Sánchez-Nielsen, E., Antón-Canalís, L., & Hernández-Tejera, M. (2004). *Hand gesture recognition for human-machine interaction*.

[11] Shamim, Fazvina Mohammed, and Sarvesh Vishwakarma (2016), *Exploiting the Motion Learning Paradigm for Recognizing Human Actions*, Bonfring International Journal of Advances in Image Processing 6.3 pp. 11-16.

[12] Suarez, J., & Murphy, R. R. (2012, September), *Hand gesture recognition with depth images: A review*, In 2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication pp. 411-417.

[13] Tang, Jingren, et al. (2018), *Structured dynamic time warping for continuous hand trajectory gesture recognition*, Pattern Recognition 80, pp. 21-31.

[14] Yaseen, Ali Fadhil. (2018), *A Survey on the Tracking of Object in Sequence Image*.

[15] Zhang, Xin, et al. (2013), *A new writing experience: Finger writing in the air using a kinect sensor*, IEEE MultiMedia 20.4 pp. 85-93.