Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 130 (2018) 953–960

Procedia Computer Science

www.elsevier.com/locate/procedia

9th International Conference on Ambient Systems, Networks and Technologies, ANT-2018 and the 8th International Conference on Sustainable Energy Information Technology, SEIT 2018, 8-11 May, 2018, Porto, Portugal

Recognition of basketball referee signals from videos using Histogram of Oriented Gradients (HOG) and Support Vector Machine (SVM)

Julius Žemgulys[a], Vidas Raudonis[a], Rytis Maskeliūnas[b], Robertas Damaševičius[c,]*

[a]*Department of Automation, Faculty of Electrical and Electronics Engineering, Kaunas University of Technology, Kaunas 44249, Lithuania*
[b]*Department of Multimedia Engineering, Faculty of Informatics, Kaunas University of Technology, Kaunas 44249, Lithuania*
[c]*Department of Software Engineering, Faculty of Informatics, Kaunas University of Technology, Kaunas 44249, Lithuania*

Abstract

Hand gestures, either static or dynamic, for human computer interaction in real time systems is an area of active research and with many possible applications. However, vision-based hand gesture interfaces for real-time applications require fast and extremely robust hand detection, and gesture recognition. Attempting to recognize gestures performed by officials in typical sports video places tremendous computational requirements on the image segmentation techniques. Here we propose an image segmentation technique based on the Histogram of Oriented Gradients (HOG) features that allows recognizing the signals of the basketball referee from videos. We achieve an accuracy of 97.5% using Support Vector Machine (SVM) for classification.

Peer-review under responsibility of the Conference Program Chairs.

Keywords: Gesture recognition; Histogram of Oriented Gradients (HOG); Support Vector Machine (SVM); basketball referee signals, one-shot learning.

1. Introduction

Humans often can recognize new gestures after seeing just one example, but for computers, recognizing even well-defined gestures, such as sign language, is much more challenging and has traditionally required thousands of training examples to learn. Our work focuses on the gesture recognition from a single training example given for each class, a problem, which is known as one-shot learning. In the case of a single training example per class, the standard tools of statistical machine learning would be very likely to fail because they will suffer from over-fitting

* Corresponding author. Tel.: 370-609-43772.
E-mail address: robertas.damasevicius@ktu.lt

1877-0509
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2018.04.095

problem. Furthermore, hand gestures can be static or dynamic. Some gestures also have both static and dynamic elements, as in human sign languages. A static gesture is a gesture in which a single posture is held for certain duration, while a dynamic gesture consists of a sequence of postures, which may be repetitive or not, and in which the posture order and the timing of the sequence may be critical[1]. Attempting to recognize dynamic gestures performed by officials in typical sports video places huge computational requirements on the image segmentation techniques[2]. It requires performing complex analysis of images and extraction of a large number of image features for further classification and decision support[3].

In a basketball game, the basketball referees have the responsibility to enforce the game rules and communicate with the scoring table using hand signals (see Fig. 1). However, there can be manual communication between referees and the scorer sometimes that causes misunderstandings and hence delays in the progress of the game. Automatic recognition of referee hand signals can contribute towards reducing the number of misinterpretation of referee decisions in a basketball game, as well as for automatic annotation of game video recordings, providing the game spectators with real-time information, as well as making the game itself more interesting and attractive to the remote viewers. With the advances of sensor and computer technology, human-computer interaction (HCI) systems become more and more popular in our daily life while the HCI technology can be used to facilitate the interaction between referees and the players and the play officials. The research is also important in the context of Ambient Assisted Living (AAL) environments as gesture-based interfaces could be used to improve the daily life of hearing impaired people as well as to control domestic appliances such as smart TVs by employing the Kinect technology, while the advances in image segmentation are important for scene and object recognition in assistive devices for visually impaired people[4].

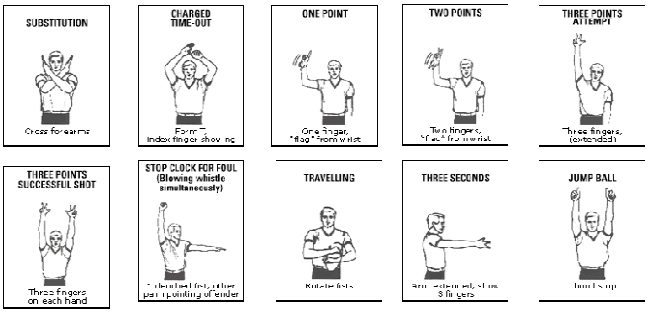

Fig. 1. Official sign language of basketball referees (adopted from[5]).

2. Related work

There are two approaches to detecting hand gestures in general and sport referee signals in particular: wearable sensors and computer vision.

The wearable sensors approach uses the sensors attached to the arms of the subjects or to his/her shirt in close proximity to the arms, or interwoven into the textile of the shirt itself. For example, Chambers et al.[2] performed recognition of 10 umpire gestures from the game of cricket using accelerometer data collected from wrist bands. Using the hidden Markov model and a variety of statistical feature sets, an accuracy of 99% has been achieved. Yeh et al.[6] use surface electromyography (sEMG) and three-axis accelerometer (ACC) sensors with Deep belief network and time-domain features to recognize the official basketball referee hand signals, reaching an accuracy of 97.9%.

The vision based approach analyses the image of the referee captured by the photo camera or extracted as a still image from the video sequence of the recorded basketball game. Then a variety of image processing techniques is used to perform image segmentation and extract image features required to perform recognition of gestures. For example, Verma[7] recognized the motion of the hand by using the Finite State Machine's (FSM) states. These states are assumed as clusters which are formed by fuzzy c-means clustering. Then the centroid of each cluster is found out mathematically, and hence the FSM states was determined and, finally, the gesture was recognized. Guyon[8] described Chalearn gesture dataset recorded using a Kinect camera including Referee Wrestling Signals and Referee Volleyball Signals. Trigueiros et al.[9] proposed a vision-based system, which is able to interpret dynamic and static

gestures of the referee. The system performs a real time hand tracking and feature extraction, and uses SVM (Support Vector Machine) for static hand posture identification, an HMM (Hidden Markov Model) for dynamic unistroke hand gesture recognition. For the hand posture recognition, an accuracy of 98.2% was achieved. Shanjia[10] used skin color information and morphological filter to generate the feature vectors for recognizing the gesture meaning, and apply the method in sports teaching.

Here we use the computer vision based approach and describe our method for recognition of basketball referee signals in the following section. The presented work is the first such attempt in the scientific literature to specifically recognize the basketball referee signals from still images.

3. Method

3.1. Outline

The algorithm of the proposed method is presented in Fig. 2 and detailed in subsections 3.2-3.5.

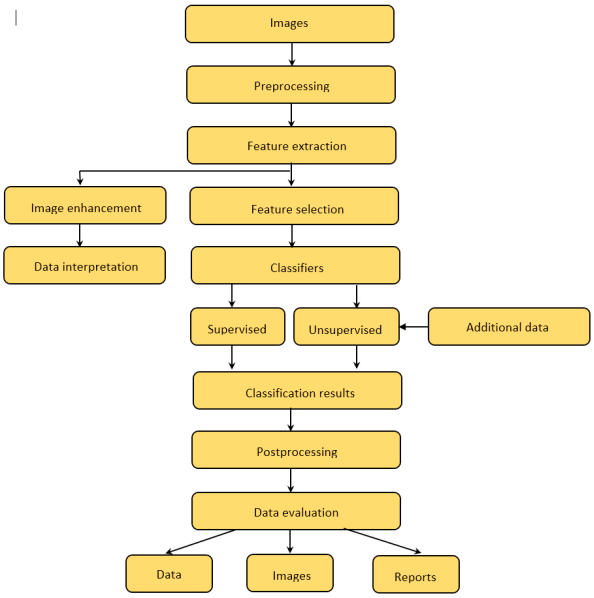

Fig. 2. Algorithm of gesture recognition

3.2. Image preprocessing

Image colors and their interrelations are usually described in color patterns or palettes. The video clips we analyze use the RGB color palette, consisting of three primary colors – red (R), green (G), and blue (B). RGB is easy to use in technology, but it is not very suitable for image processing, because the components of these colors are highly correlated, this is a problem for analyzing the image and realizing recognition algorithms. For these reasons, RGB images were converted into black and white halves (BW) using the formula (1):

$$BW = 0.333R + 0.333G + 0.333B \quad (1)$$

The points are described in 8 bits, so when looking at the intensity of the points it ranges from 0 (black) to 255 (white), for example, see an image of the referee and its histogram in Fig. 3.

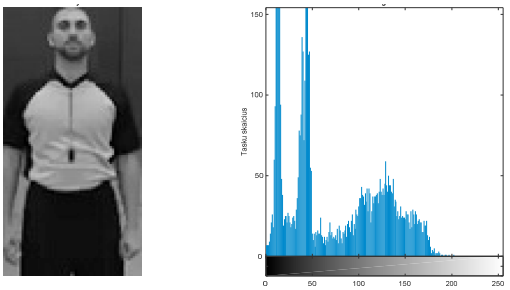

Fig. 3. Histogram of cut-out photo of basketball referee (left) and its histogram (right)

3.3. Edge detection

The next step is to find the edges to better distinguish the shape of the referee and such points would be described in two bits. Using the edge detection method can one compensate for errors due to different image illumination or quality. The methods of finding the edges differ in their ability to detect curved lines or provide finer. After experimentally testing the Kirsch[11], Sobel[12], Prewitt[13], Canny[14] and enhanced Canny[15] methods, we decided to use the Sobel method[12]. By selecting the appropriate threshold, the referee's contour is visible. The Sobel operator calculates the 2-D spatial gradient of an image, thus emphasizing the spatial frequency regions that correspond to the edges. Usually it is used to determine the approximate absolute gradient size at each point in the gray image.

The Sobel operator consists of a 3x3 point window and it is slided through the image. The kernels are designed to maximize response to the vertical and horizontal edges associated with the pixel grid. Assuming that G_x is a gradient for horizontal edges and G_y is a gradient for vertical edges, the gradient size is expressed by the formula (2):

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{2}$$

When the 3x3 window moves across the entire image, the values for each pixel are converted (ranging from 0 to 1) and a certain value is selected (the value selected for the referee contour detection was 0.7), and the contours of the desired figure are distinguished. The result of edge detection in sample image is illustrated in Fig. 4.

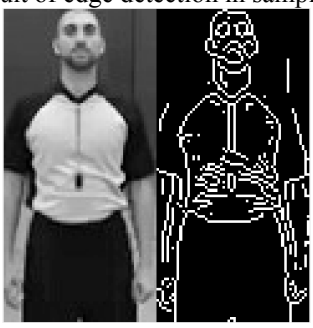

Fig. 4. Edge detection using Sobel method

3.4. Feature extraction

In Fig. 4, only one person (referee) stays in the photos, therefore, the most important is to isolate the referee from the background and then recognize his gestures. The appearance and shape of an object can be described in terms of local intensity or edge direction, even without the knowledge of these parameters. In practice, this is done by dividing the window into small spatial regions aka cells, whereas in each cell there is a local histogram of the 1-D gradient directions. By combining the image obtained with the histogram with the distinguished features, this feature extraction method is called the Oriented Gradient Histogram (HOG) method[16].

To find the gradient, one needs a grayscale image of the window I (size depends on the size of the cell). Then the gradients Ix and I_y are found as follows:

$$I_x(r,c) = I(r,c+1) - I(r,c-1) \tag{3}$$

$$I_y(r,c) = I(r-1,c) - I(r+1,c)$$

Then the gradient is transformed into polar coordinates, and their angle is limited to 0 to 180 degrees, so that the gradients, which show in different directions, have the same angle:

$$\mu = \sqrt{I_x^2 + I_y^2} \tag{4}$$

$$\theta = \frac{180}{\pi}\left(\tan_2^{-1}\left(I_y, I_x\right) \bmod \pi\right)$$

The window is split into adjacent, uncovered CxC pixel-sized creeps (C = 8). In each trail, the histogram of oriented gradients is calculated in directions B (B = 9). Since there are so few directions, a pixel whose orientation is close to the other direction may be assigned to another direction. To avoid this problem, each cell is assigned to two close-crypts, a small part of the pixel gradient size μ decreases linearly, depending on the orientation of the pixel gradation from two near-directional directions.

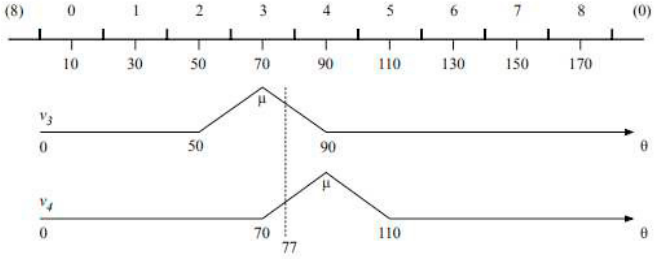

Fig. 5. Finding the gradient when B = 9

In Fig. 5 we can see how gradients are assigned to the centers of the adjacent 70 and 90 degrees. The gradient orientation at this lobe is 77 degrees and the gradient is assigned to 0.65 μ for the third direction and 0.35 μ for the fourth direction. The sum of two assignments is always equal to μ.

The cages are grouped into overlapping 2x2 blocks, each block having a size of 2Cx2C pixels. Two vertically or horizontally successive blocks are covered by two paths, which means that the block's step is C pixels. As a result, each cell is covered by four blocks. The four-cell histograms are combined and one characteristic value b is obtained in each block, and it is normalized using its Euclidean form:

$$b \leftarrow \frac{b}{\sqrt{\|b\|^2 + \varepsilon}} \tag{5}$$

here ε is a small positive constant to avoid division by zero in blocks without gradients.

Finally, the HOG feature is calculated by combining the features of the normalized blocks into one vector (normalization is performed twice before and after the minimum search):

$$h \leftarrow \frac{h}{\sqrt{\|h\|^2 + \varepsilon}}, \; h_n \leftarrow \min\left(h_n, \tau\right) \tag{6}$$

here h_n is the n-th input of h, and τ is a positive threshold.

The cropping of h inputs so that they do not exceed τ (after the first normalization) ensures that very large gradients will not have much impact, otherwise the details of the other picture will be thrown off. The final

normalization makes the HOG feature independent of the overall video contrast[16]. The resulting HOG feature consists of a number of histograms, which are four times larger than blocks. In this paper, all training photos were cropped to 128x64 pixels. If you use a 4x4 pixel path, then the photo will fit 16 cells horizontally and 32 streams vertically, resulting in 31 blocks vertically and 15 blocks horizontally, since the block consists of four tracks and each of the histograms has nine rows, and then the resulting length of the vector *h* is calculated as follows:

$$h = 31 \times 15 \times 4 \times 9 = 16740 \tag{7}$$

Visualization of the HOG features using different cell sizes are presented in Fig. 6.

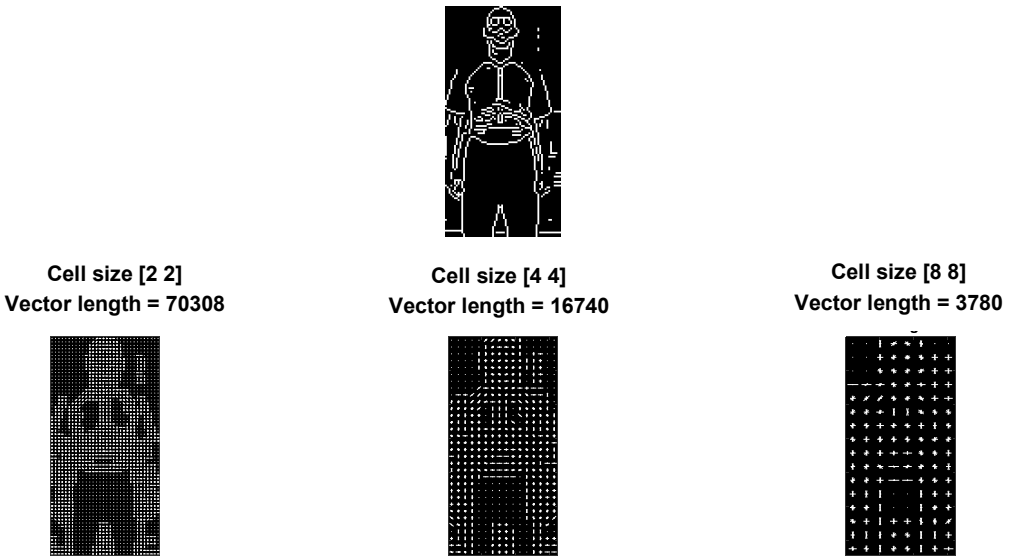

Fig. 6. HOG features using different cell sizes

As it is visually difficult to determine, which cell size is best suited for using the referee and his signs to distinguish it from the background; therefore, the difference between summed vectors with different cell sizes was calculated (the bigger the difference, the easier it will be for a referee gesture to be classified) as follows:

$$h_{diff} = \sum \left(h_1 - h_2 \right) \tag{8}$$

3.5. Classification

Support Vector Machine (SVM) is a type of classifier with a supervisor, whose operation is based on finding optimal separation between points representing different classes. SVM defines the hyperplane depending on the training data. The hyperplane forms the boundaries of the decisions that make the classification. The hyperplane is designed to divide input data into two classes, based on the kernel function. Most SVM classifiers can classify objects into two classes, but if required, a multi-level classification scheme can also be adopted. The hyperplane can be described as a line described by a certain function. This line is selected at the maximum distance from all data points, thus reducing the influence of noise in data. The goal of the SVM algorithm is to find the optimum margin hyperplane finds the maximum data margin between classes.

4. Experiments and results

4.1. Dataset

Video materials, downloaded from Youtube[17], were used to identify the basketball referee signals, in which the referee stands in front of the camera and all the gestures are clearly visible. In total, 20 images were cut out with a referee for four classes of data: a standing referee (no gesture), the three points gesture, the stop clock gesture and the player substitution gesture. A sample image cut out from the video clip is shown in Fig. 7.

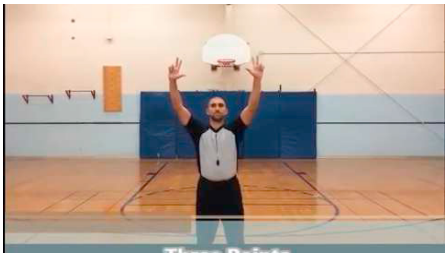

Fig. 7. Sample image from dataset[17]

4.2. Procedure

Three gesture signs (three points, substitution, and stop clock) and a standing referee (with no gesture sign shown), i.e. four classes of gestures in total, were attempted to be recognized. For each class, 20 different images were used, with different gesture signs showed. Typically, the SVM classification is used for two classes (positive and negative) only, however, the multi-level classification was used in this case. The features of the HOG method were distinguished with 4x4 paths, so that 20x16740 vectors were assigned to one class. During the testing stage, 20 images were used with referees showing different gestures. We have performed the classification with linear-kernel SVM using the MATLAB (Mathworks Inc.) software package, and the results are presented in the next subsection.

4.3. Results

We evaluated the classification results using standard accuracy and F-score metrics. We have obtained an accuracy of 0.9750, and F-score of 0.9495. The confusion matrix of classification results is presented in Fig. 8. The stop clock and three-point signals were easily distinguished, but due to the small amount of training data, a single standing referee and substitution signals were mixed, and the differences between them were very small.

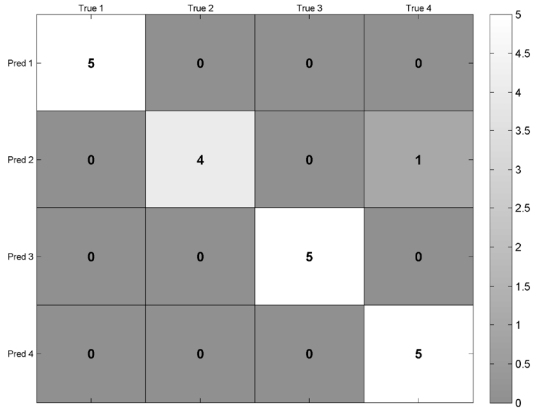

Fig. 8. Confusion matrix for gesture sign classes (classes: 1- standing referee, 2- substitution sign, 3 - stop clock sign, and 4 – three points sign)

5. Conclusions and future work

A classifier of the hand gesture signals of the basketball referee was implemented. First, the images from video stream were converted to black and white halves, their size changed to 128x64, and edges were identified using the Sobel edge detection method. Using the HOG feature extraction method, we used the size of a 4x4 cell, obtaining images described by a vector, whose length $h = 16740$. After applying the SVM classification, we have obtained a recognition accuracy of 0.9750, and F-score of 0.9495.

In future work, we will evaluate our method on a larger dataset of basketball referee hand signal images, and will test the method on live video feed rather than still images, aiming for an application on real-world basketball games.

References

1. Liang B, Zheng L. Gesture recognition from one example using depth images. *Lecture Notes on Software Engineering* 2013, 1 (4), 339.
2. Chambers GS, Venkatesh S, West GAW, Bui HH. Segmentation of intentional human gestures for sports video annotation. 10th *International Multimedia Modelling Conference,* Brisbane, Queensland, Australia, 2004, 124-129.
3. Gabryel M, Damasevicius R. The Image Classification with Different Types of Image Features. In *Artificial Intelligence and Soft Computing. ICAISC 2017.* Lecture Notes in Computer Science, vol. 10245, 497-506, 2017.
4. Petraitis T, Maskeliunas R, Damasevicius R, Polap D, Wozniak W, Gabryel M. Environment Recognition based on Images using Bag-of-Words. *In 9th International Joint Conference on Computational Intelligence,* IJCCI 2017, Funchal, Madeira, Portugal, November 1-3, 2017. SciTePress 2017, 166-176.
5. Referee's Hand Signals. Available at: http://websites.sportstg.com/
6. Yeh CW, Pan TY, Hu MC. A Sensor-Based Official Basketball Referee Signals Recognition System Using Deep Belief Networks. In *23rd International Conference on MultiMedia Modeling - MMM 2017,* Reykjavik, Iceland, 2017, Part I. Lecture Notes in Computer Science 10132, Springer 2017, I, 565-575.
7. Verma R, Dev A. Vision based Hand Gesture Recognition Using Finite State Machines and Fuzzy Logic. In *2009 International Conference on Ultra Modern Telecommunications & Workshops,* St. Petersburg, 2009, 1-6.
8. Guyon I, Athitsos V, Jangyodsuk P, Hamner B, Escalante HJ. ChaLearn gesture challenge: Design and first results. In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012, 1-6.
9. Trigueiros P, Ribeiro F, Reis LP. Vision Based Referee Sign Language Recognition System for the RoboCup MSL League. In: RoboCup 2013: Robot World Cup XVII. Lecture Notes in Computer Science, vol. 8371, 2014, 360-372.
10. Shanjia, Z. Environment model construction of hand gesture recognition for sports teaching. *Agro Food Industry Hi-Tech* 2017, 28(1), 2388-2391.
11. Kirsch R. Computer determination of the constituent structure of biological images. *Comput. Biomed. Res.* 1971, 4, 315–328.
12. Sobel I. Camera Models and Perception. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 1970.
13. Prewitt JMS. Object Enhancement and Extraction; Lipkin, B.S., Rosenfeld, A., Eds.; Picture Analysis and Psychopictorics; Academic Press: New York, NY, USA, 1970, 5–149.
14. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 1986, 8, 679–698.
15. Zhao JJ, Wang J, Wei W, Chang XM, Pei B. Enhanced Boundary Detection Method Based on Canny Theory. Information Technology Journal 2013, 12: 6723-6728. DOI: 10.3923/itj.2013.6723.6728
16. Dalal N, Triggs B. Histograms of Oriented Gradients for Human Detection. In 2005 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05),* Vol. 1, 2005, 886-893.
17. FIBA signals - basketball referee education. Available at: https://www.youtube.com/watch?v=k1yNcWsvu84&t=297s, 2016.