Motion Recognition in Football Game using Saptio-temporal Volumes and Video Image Processing

Hamid Esmaeili Jajarm/Msc. Student
Department of Electrical and Computer Engineering
Islamic Azad University, Science and Research branch.
Tehran, Iran
H.esmaeili@iau-shahrood.ac.ir

Alireza Behrad/ Assistant Professor
Faculty of Engineering
Shahed University
Tehran, Iran
Behrad@shahed.ac.ir

Ali Eftekhari/ Student
Department of Electrical Engineering
Islamic Azad University, South Tehran branch.
Tehran, Iran
Ali_eftekhary2003@yahoo.com

***Abstract*—This paper presents motion recognition in football game using video image processing. The proposed algorithm is based on spatio-temporal volumes for motion recognition in football game. Spatio-temporal volume unifies the analysis of spatial and temporal information by constructing a volume of data in which consecutive images are stacked to form a third, temporal dimension. In our algorithm, after applying necessary pre-processing on video frames and extracting player contour, spatio-temporal volumes are constructed for each motion. Then feature vectors are extracted from volumes. Finally a classification algorithm is applied. We examined four different classifiers for feature vector classification and motion recognition. Experimental results showed the correct motion recognition rate of 85 percent. In addition, the proposed algorithm is robust to traditional problems of motion recognition like illumination variations, player initial position in each frame and different player velocities.**

Keywords-video processingt; human motion recognition; football; spatio temporal volume; k-Nearest Neighbour; SVM;

I. INTRODUCTION

In recent years, human motion recognition and reconstruction is active and attractive subject for research. Human motion recognition is ill posed and difficult problem because of jointed and non-rigid motion of human body and occlusion problem. Considering the problems mentioned above, researchers have been studied three dimensional motion recognition from only academic viewpoint. Human motion recognition has many applications in visual surveillance such as crowd flux statistics and congestion analysis, human identification, anomaly detection and control applications like virtual reality and software games.

A 3D spatio-temporal volume (STV) is made by stacking frames over a given sequence. Accurate localization, alignment and possibly background subtraction are needed. Gorelick et al. [1] firstly stacked silhouettes over a given sequence to create an STV. Afterward, the solution of the Poisson equation is utilized to derive orientation features and local space–time saliency. To obtain global features for a given temporal range, weighted moments over these local features are calculated. Dealing with performances of different temporal durations, Achard et al. [2] employed a set of space–time volumes for each sequence. The STV surface and related local descriptors are discussed in numerous works. Yilmaz and Shah [3] utilized differential geometric properties on the STV surface, for example maximum and minimum in the space–time domain but it is sensitive to noise on the surface. Yan et al. [4] extend this idea by first constructing 3D exemplars from multiple views, for each frame in a training sequence. In [5] Jiang and Martin used 3D-shaped flows at edge points for matching in cluttered background. Ke et al in [6] used combination of STVs of silhouettes and flow.

In this paper, we used STV surface for analyzing different action in football game. To make our algorithm robust against the player cloth color, illumination variance, size and position change we proposed a new algorithm for STV construction based on player contours. The extracted features are selected to be scale and size invariant.

The rest of the paper is organized as follows: In section 2, we describe the collected data set of football video games; Section 3 explains the preprocessing step for constructing spatio-temporal volumes. Section 4 analyzes spatio-temporal volumes for feature vector extraction. In section 5, feature vector classification using four different classifiers are presented for human motion recognition. Section 6 shows the experimental results for the proposed method and conclusion appear in section 7.

II. COLLECTED DATABASE

Our database includes 40 football video games, with 10 videos for “kick” actions, 10 videos for “head” actions, 10 videos for “out” actions and 10 videos for “diving” goalkeeper actions. Fig. 1 shows 4 samples of these actions. The total frame number of videos is not constant and varies between 12 to 28 frames. The resolution of the all video frame is 477*695.

III. PREPROCESSING

A. *Player contour extraction in image sequence*

To make our algorithm robust against the player cloth color, illumination variance, size and position change, we used contours of player to construct STV.

Figure 1. Four samples of different action in our database

Because of camera motion as well as static state of players in some actions, it is not possible to extract the contours of player automatically using traditional motion segmentation algorithms. Therefore, we used a semi-automatic method to extract human contour which use edge detection algorithm followed by correction by human user. It should be mentioned when player's body contacts with the ball, both of them are used to extract the contour.

We used following algorithm to obtain player's contour:

- First we use canny edge operator to extract player internal and external edges.
- Connect gaps and small dints to obtain the player images with the smoothed edge. We used disk shaped structuring element with radius of 1 to 6.(closing operator)
- Fill player region to find the player silhouette.
- Extract the edge of the obtained silhouette and smooth the extracted edge contours to obtain player's contour.

In fig. 2, samples of extracted player's contour are illustrated.

B. *STV construction*

When the contours of player in different frames are obtained, they are used to construct STV. To construct the STV, the extracted contours of consecutive frames for each action are stacked over each other to construct a STV

IV. FEATURTE EXTRACTION

A. *Normalizing in temporal direction*

The feature space should be insensitive to action velocity, player size and position. To deal with different action velocities, we divided the constructed STV to sectors. For this purpose we divide the total frames of each action to constant number of slices, each slice containing several consecutive frames. The number of frames in each slice is determined using the following equation:

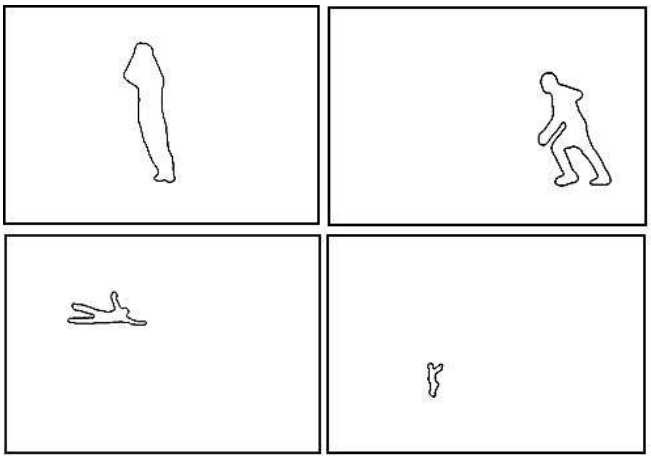

Figure 2. The player contours of images in Fig. 1

$$f = N/M \tag{1}$$

where N is the total frame's number, M is the slice number and f is the number of frames in each slice. We selected $M = 4$ experimentally in the proposed method. Therefore, by varying N from 12 to 28, f varies from 3 to 8. Each slice is further divided to several sectors as shown in fig. 3. We extract the feature for each sector. Since we have the constant number of slice and sector numbers, therefore the size of feature vector for different actions and speeds are the same. Fig. 3 shows the division of a typical frame to S = 5 and S = 7 sectors. By dividing contours to M sector and S section, we have M*S different areas in spatio-temporal volumes for feature vector extraction and matching.

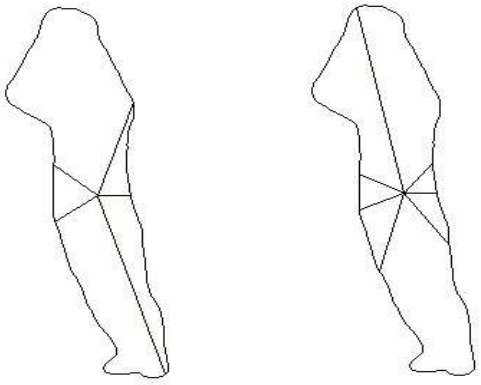

Figure 3. Division of typical frame to S=5 , S=7 sectors

B. Normal Vector

The extracted STV can be considered as 3D surface. One of the most important features for 3D surface is the normal vector to the surface in different position which is nearly insensitive to scale and size of the surface. We calculate normal vector for each sector as our first feature. To extract normal vector, we fit a plane for each sector and use normal vector of the plane as features. To fit a plane on points we use the following equation for a plane.

$$Ax + By + Cz = 1 \tag{2}$$

In (2), x , y and z denote points in spatio-temporal coordinates where z refers to the temporal frame number and A, B and C are normal vector moments of the plane. Extracted points from each sector forms a set of equations as follows:

$$\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \dots & \dots & \dots \\ \dots & \dots & \dots \\ x_n & y_n & z_n \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \dots \\ \dots \\ 1 \end{bmatrix} \Rightarrow A_m X = B_m \tag{3}$$

Where $x_1, y_1, z_1, \dots, x_n, y_n, z_n$ are sector points and A, B and C are normal vector moments of the best fitted plane that can be presented by matrix X. We can solve eq. (3) and obtain X, using Least Mean Squared (LMS) criterion as follows:

$$X = \left(A_m^T A_m\right)^{-1} A_m^T B_m \tag{4}$$

where X is the normal vector of the plane. By dividing contours to M sector and S section, we obtain a vector with size of 3 *S *M as normal vector features.

C. Curvature based features

Curvature of curve or surface is another proper feature for describing surface. Curvature is not sensitive to the player's size; therefore we don't need additional stage to normalize the player's size. To calculate the curvature we represent each contour as 2D curve as follows:

$$r(u) = \left(x(u), y(u)\right), \tag{5}$$

To calculate the curvature it is necessary to smooth the curve using the Gaussian Function of width σ as follows:

$$X(u,\sigma) = x(u) * g(u,\sigma)$$
$$Y(u,\sigma) = y(u) * g(u,\sigma)$$
$$R(u,\sigma) = \left(X(u,\sigma), Y(u,\sigma)\right)$$

where $g(u,\sigma)$ is Gaussian function. The curvature is calculated using the following equation:

$$k = \frac{X'Y'' - X''Y'}{\left(\left(X'\right)^2 + \left(Y'\right)^2\right)^{3/2}} \tag{7}$$
$$X' = \frac{dX}{du}, X'' = \frac{d^2X}{du^2}$$

The curvature value in eq. (7) is obtained for all points of surface. We then calculate the maximum of curvature, minimum of curvature, average of curvature and zero crossing of the curvature graph in each sector as feature of that sector.

D. Feature based on player bounding box

Another feature that is used in the proposed algorithm to differentiate different action is the difference between player's contour and its bounding box. Because of player's motion, bounding box area and the contour circumference change and we can use their differences as features. We normalize rectangle's area and circumference because they are sensitive to the player's size.

E. Feature based on periodicity of motion

The periodicity of the motion shows that how much the position and shape of the player contour is similar to its position and shape in the first frame. To measure periodicity, we calculate the similarity between the silhouettes of the player in the different frame of the STV and the silhouettes of the first frame. To calculate similarity, we used normalized cross correlation.

V. CLASSIFICATION

We used four different classifiers to classify 4 classes of football game actions. The classifiers are:

1. k-Nearest Neighbor (kNN)
2. Gaussian radial basis function Support Vector Machine (rbf SVM)
    a. One versus One Support Vector Machine
    b. One versus All Support Vector Machine
    c. Directed Acyclic Graph Support Vector Machines (DAGSVM)

A. k-Nearest Neighbor

k-Nearest Neighbor is a supervised learning algorithm based on minimum distance of input sample from the training samples which determine k-nearest neighbors to classify the action. Each of these k neighbors is related to a class of action. We select a class of action with maximum

number of occurrence. We used Euclidean distance to determine nearest neighbors.

B. *one versus one*

In this approach, an SVM is constructed for every pair of classes by training it to discriminate the two classes. Thus, the number of SVMs used in this approach is $w(w-1)/2$ that *w* is the number of output classes. An SVM for a pair of classes is constructed using training samples belonging to the two classes only. In this stage, a majority voting scheme is used. The outputs of SVMs are used to determine the number of votes won by each class. The class with maximum number of votes is assigned to the test sample.

C. *one versus All*

In this approach, an SVM is constructed for each class by discriminating that class against the remaining $(w-1)$ classes. The number of SVMs used in this approach is *w*. A test sample x is classified by using the winner-takes-all decision strategy, i.e., the class with the maximum value of the discriminant function is assigned to it. All the training samples are used in constructing an SVM for a class.

D. *DAGSVM*

In the training phase, it works as the one versus one method solving $w(w-1)/2$ binary SVMs. However, in the testing phase, it uses a rooted binary DAG which has $w(w-1)/2$ internal nodes and w leaves. Fig. 4 shows the DAG scheme. Given a test sample x, starting at the root node, a pairwise SVM decision is made and either class is rejected. Then it moves to either left or right depending on the result, and continues until reaching to one of leaves which indicates the predicted class. So the DAG requires $(w-1)$ comparisons and hence is more efficient than the one versus one method.

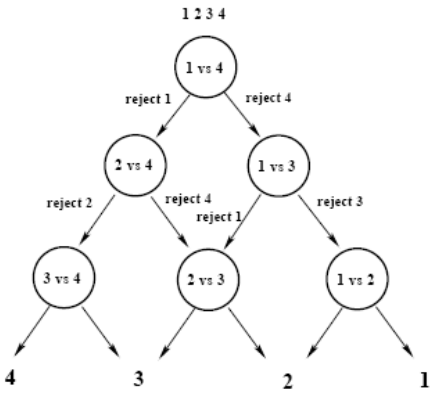

Figure 4. A DAG for four classes

VI. EXPRIMENTAL RESULT

The proposed algorithm implemented using a MATLAB program. As mentioned before we divided the contour of each frame to S sectors. In this article, we selected different value of S as S=5, S=7, S=10, S=12 and S=15 for classification to show its effect. For the validation we used leave one out method and the algorithm tested with four classifiers mentioned in previous section. Figure 5 shows the results of the action recognition using different classification algorithms and different sector numbers.

To increase the efficiency algorithm we changed the effect of different features by assigning weight for each feature. We tested different weight for the features and selected the best weights. Figure 6 shows the results of recognition for the best weights.

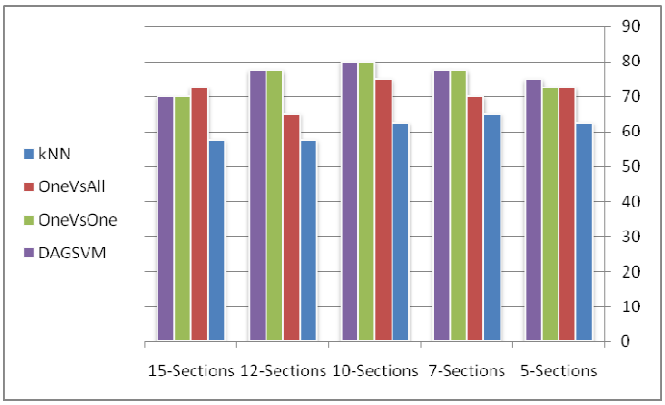

Figure 5. Action recognition rates using different classifiers and different sector values.

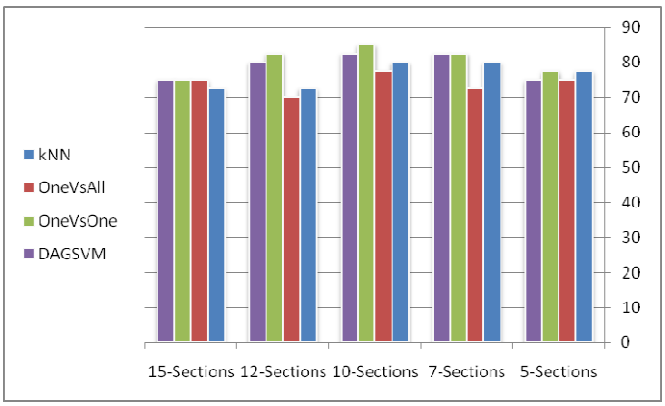

Figure 6. Action recognition rates using different classifiers and different sector values by assigning weight to different features.

The experimental results showed that the one versus one method for 10 sectors outperformed the other methods. For this method, we obtained 85% of correct classification rate. Table 1 shows the action confusion matrix for the entire database for the best result. The element in row i and column j of the matrix indicates the number of action i which are classified as action j. The sum of all elements in every row is 10.

TABLE I. ACTION CONFUSION MATRIX FOR THE PROPOSED METHOD

	head	out	kick	goalkeeper diving
head	9	1	0	0

out	0	10	0	0
kick	2	0	7	1
goalkeeper diving	1	1	0	8

To compare the results of the paper with those of other methods we implemented the method of [7] as well. Table 2 shows the action confusion matrix for the method proposed in [7] applied to the videos in our database database. The overall correct classification rate attained by this method is 57.5%.

TABLE II. ACTION CONFUSION MATRIX FOR METHOD FROM [7]

	head	out	kick	goalkeeper diving
head	5	0	4	1
out	0	5	5	0
kick	1	0	9	0
goalkeeper diving	3	0	3	4

VII. CONCLUSION

A new algorithm based on spatio-temporal volumes for human action recognition in football game is presented. In this algorithm, after applying preprocessing steps on video frames, spatio-temporal volumes are constructed for each action. These volumes are then analyzed and feature vectors are extracted. Four different classifiers including k-Nearest Neighbor, one versus one SVM, one versus All SVM and DAGSVM are utilized for feature vector classification and action recognition. We tested the proposed algorithm with the collected data set and experimental results showed the reliability of our algorithm. In addition, the proposed algorithm is robust to traditional problems of human motion recognition like illumination variations, human initial position in each frame, different action velocities.

REFERENCES

[1] L. Gorelick, M. Blank, E. Shechtman, M. Irani and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, pp. 2247-2253, 2007.

[2] C. Achard, X. Qu, A. Mokhber and M. Milgram, "A novel approach for recognition of human actions with semi-global features," *Machine Vision and Applications*, Vol. 19, pp. 27-34, 2008.

[3] A. Yilmaz and M. Shah, "A differential geometric approach to representing the human actions" *Computer Vision and Image Understanding*, Vol. 119, pp. 335-351, 2008.

[4] P. Yan, S. M. Khan and M. Shah, "Learning 4D action feature models for arbitary view action recognition," *Proc. Int. Conf. on Computer Vision and Pattern Recognition*, pp. 1-7, June 2008.

[5] H. Jiang and D. Martin, "Finding actions using shape flows" Proc. Int. Conf. on Computer Vision, pp. 278-292, October 2008.

[6] Y. Ke, R. Sukthankar and M. Hebert, "Spatio-temporal shape and folw correlation for action recognition," *Proc. Int. Conf. on Computer Vision and Pattern Recogniton*, pp. 1-8, June 2007.

[7] K. Guo, P. Ishwar and J. Konrad, "Action Recognition in Video Covariance Matching of Silhouette Tunnels" *Proc. Int. Conf. on Computer Graphics and Image Processing*. Brazil, pp. 299-306, 2009.