Action Categorization in Soccer Videos Using String Kernels

4 authors:

Lamberto Ballan
University of Padova
89 PUBLICATIONS **3,909** CITATIONS

Marco Bertini
University of Florence
253 PUBLICATIONS **4,049** CITATIONS

Alberto Del Bimbo
University of Florence
672 PUBLICATIONS **14,277** CITATIONS

Giuseppe Serra
Università degli Studi di Modena e Reggio Emilia
125 PUBLICATIONS **4,520** CITATIONS

ACTION CATEGORIZATION IN SOCCER VIDEOS USING STRING KERNELS

Lamberto Ballan, Marco Bertini, Alberto Del Bimbo and Giuseppe Serra

Media Integration and Communication Center, University of Florence, Italy
`http://www.micc.unifi.it/vim`

ABSTRACT

Action recognition is a crucial task to provide high-level semantic description of the video content, particularly in the case of sports videos. The bag-of-words (BoW) approach has proven to be successful for the categorization of objects and scenes in images, but it's unable to model temporal information between consecutive frames for video event recognition. In this paper, we present an approach to model actions as a sequence of histograms (one for each frame) represented using a traditional bag-of-words model. Actions are so described by a string (*phrase*) of variable size, depending on the clip's length, where each frame's representation is considered as a *character*. To compare these strings we use Needlemann-Wunsch distance, a metrics defined in the information theory, that deal with strings of different length. Finally, SVMs with a string kernel that includes this distance are used to perform classification. Experimental results demonstrate the validity of the proposed approach and they show that it outperforms baseline kNN classifiers.

Index Terms— video annotation, action classification, sports videos, string kernel, edit distance

1. INTRODUCTION AND RELATED WORKS

The use of local features has recently become very popular for object detection and recognition tasks because of the robustness w.r.t. partial occlusions and clutter. Many approaches have been presented, but a common idea is to model a complex object or a scene by a collection of local salient points. Each of these local features describes a small region around the interest point and thus these features are robust against occlusion and background changes. To achieve robustness to changes of viewing conditions they should be invariant to geometrical transformations such as translation, rotation, scaling and also affine transformations [1, 2]. In particular, SIFT features by Lowe [3] have become the de facto standard because of their high performances and (relatively) low computational time. In fact, SIFT features have been frequently and successfully applied to object or scene recognition and also to many other related tasks.

Based on this kind of local features, the bag-of-words model (BoW) has achieved a great popularity in the latest
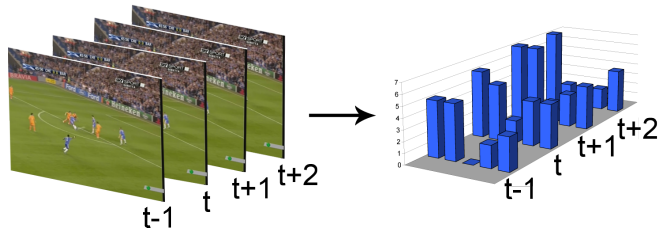
Fig. 1. Video clips are represented as a sequence of BoW histograms; actions are so described by a string (*phrase*) of variable size, depending on the clip's length.

years, especially for the task of object recognition. The BoW model was originally developed for document categorization in a text corpus, where each document is represented by its word frequency. In the visual domain, it is applied to images or video clips that are represented by the frequency of "visual words". The main reason of its success is that it provides methods that are sufficiently generic to cope with many object types simultaneously. We are thus confronted with the problem of generic visual categorization [4, 5, 6, 7], like classification of objects or scenes, instead of recognizing a specific object class.

The efficacy of the BoW approach is demonstrated also by the large number of systems based on this approach that participate to the TRECVID [8] challenge. Recently, part-based and BoW models have been successfully applied also to the human action classification problem [9, 10] and to event recognition in videos, typically using salient features that represent also temporal information (e.g. spatio-temporal gradients). Classification of events and actions is particularly required in video indexing and retrieval where dynamic concepts occur very frequently. Unfortunately, for this purpose the standard BoW model has shown some drawbacks with respect to the traditional image categorization task. Perhaps the most evident problem is that it does not take into account temporal relations between consecutive frames. Recently, few works have been proposed to cope with this problem. Wang *et al.* [11] have proposed to extend the BoW representation constructing relative motion histograms between visual words. In this way, they are able to describe motion of visual words obtaining better results on video event recognition. Xu and
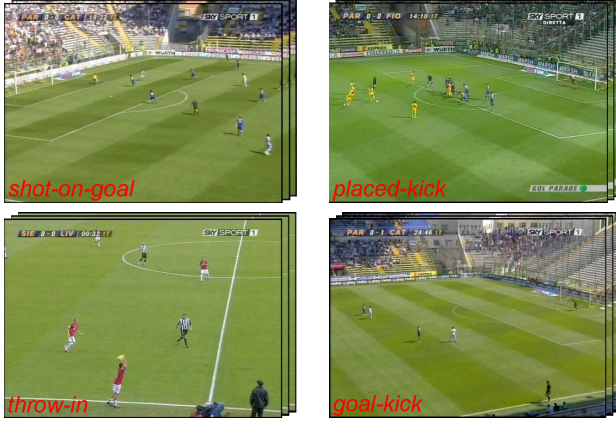
Fig. 2. Our dataset consists of four different actions: *shot-on-goal*, *placed-kick*, *throw-in* and *goal-kick*.

Chang [12] represented each frame of video clips as a bag of orderless descriptors, applying then Earth Mover's Distance (EMD) to integrate similarities among frames from two clips. They further build a multi-level temporal pyramid, observing that a clip is usually comprised of multiple sub-clips corresponding to event evolution over time. Video similarity is finally measured by fusing information at different pyramid levels.

In this paper, we present an approach to model actions as a sequence of histograms (one for each frame) represented by a traditional bag-of-words model (Fig. 1). Actions are so described by a "phrase" of variable size, depending on the clip's length, and different actions are compared using edit distance. The basic idea is to describe video clips using a global description of the video content that is able to incorporate temporal relations; in other words, each clip is described as a string (*phrase*) formed by the concatenation of the bag-of-words of consecutive frames (*characters*). Video phrases can be compared by computing edit distances between them and, in particular, we use the Needleman-Wunsch distance [13], that performs a global alignment on sequences dealing with video clips of different lenghts. Therefore, using this kind of representation we are able to perform categorization of actions or video events. Following the promising results obtained in text categorization [14] and in bioinformatics (e.g. protein classification) [15], we investigate the use of SVM based on an appropriate string kernel to perform classification using the edit distance. To test the effectiveness of the proposed approach, we present experimental results on action categorization in soccer videos. We chose the soccer domain because it shows a real-world challenging setup, and also because sports videos content analysis (particularly soccer) is a very active area, due to its wide popularity and high commercial potentials. In fact, the capability of recognizing actions and events is essential to perform automatic highlights identification and semantic annotation of sports videos [16, 17, 18].

For this purpose, we collected a new dataset consisting of 100 video clips and 4 frequent actions (see Fig. 2): *shot-on-goal*, *placed-kick*, *throw-in* and *goal-kick*. Our experiments show that classification results obtained by SVM and string kernels outperform baseline kNN classifiers and, more generally, they demonstrate the validity of the proposed method.

The rest of the paper is organized as follows: the techniques for frame and action representation are discussed in Sect. 2; the classification method, including details about the SVM string kernel, is presented in Sect. 3; experimental results are discussed in Sect. 4 and, finally, conclusions are drawn in Sect. 5.

2. ACTION REPRESENTATION

2.1. Frame Representation

Each frame is represented by a bag of features, because this representation is flexible and has been successfully applied to various image analysis tasks. First of all, a visual vocabulary is obtained by vector quantization of large sets of local feature descriptors. Typically this task is performed through k-means clustering. Once a vocabulary is obtained we compute a visual word frequency vector for each frame, counting the number of occurrences of visual words from the visual vocabulary in that frame. This frequency vector is used as frame representation. In particular, in this work we use SIFT features as local salient points.

2.2. Action Representation

Structurally an action is a sequence of frames, and may have different lengths depending on how the action has been carried out. We represent an action by a sequence of visual words frequency vectors, computed from the frames of the sequence; we call this sequence *phrase* (i.e. a string), where each vector is considered as a *character*. To compare these strings, and consequently actions, we can adapt metrics defined in the information theory.

The edit distance between two string of characters is the number of operations required to transform one of them into the other (substitution, insertion and deletion). In particular our approach uses the Needleman-Wunsch distance [13] because it performs a global alignment that accounts for the structure of the strings and the distance can be considered as a score of similarity. The basic idea is to build up the best alignment through optimal alignments of smaller subsequences, using dynamic programming. Considering the cost matrix C that tracks the costs of the edit operations needed to match two strings, we can then write the cost formula for the alignment of the a_i and b_j characters of two strings as:

$$C_{i,j} = min(C_{i-1,j-1} + \delta(a_i, b_j), C_{i-1,j} + \delta_I, C_{i,j-1} + \delta_D)$$

		S	E	N	D
	0	1	2	3	4
A	1	1	2	3	4
N	2	2	2	2	3
D	3	3	3	3	2

Fig. 3. Needleman-Wunsch edit distance: text example.

Fig. 4. Needleman-Wunsch edit distance: video example.

where $\delta(a_i, b_j)$ is 0 if the distance between a_i and b_j is close enough to evaluate $a_i \approx b_j$ or the cost of substitution otherwise, δ_I and δ_D are the costs of insertion and deletion, respectively. Fig. 3 and Fig. 4 show an example of the evaluation of the Needleman-Wunsch distance for the case of text and soccer action, respectively. The distance is the number in the lower-right corner of the cost matrix. The traceback that shows the sequence of edit operations leading to the best alignement between the sequences is highlighted in each cost matrix. The algorithm is $O(mn)$ in time and $O(min(m,n))$ in space, where m and n are the lengths of the two strings being compared.

A crucial point is the evaluation of the similarity among characters ($a_i \approx b_j$). In fact, when evaluating this similarity on text the number of characters is limited; this permits to define a similarity matrix between characters. Instead, in our case each frequency vectors is a different character, therefore we deal with an extremely large alphabet. This requires to define a function that evaluates the similarity of two characters. Since in our approach each character is an histogram we have evaluated several different methods to compare the frequency vectors of two frames, p and q. In particular we have considered the following distances, that are briefly described in the following: Chi-square test, Kolmogorov-Smirnov test, Bhattacharyya, Intersection, Correlation, Mahalanobis.

Chi-square test is a statistical method that permits to compare an observed frequency with a reference frequency. It is defined as:

$$d(p,q) = \sum_{k=1}^{N} \frac{(p(k) - q(k))^2}{p(k) + q(k)}. \qquad (1)$$

Low value means a better match than a high score.

Kolmogorov-Smirnov test is a statistical method that quantifies the distance between one cumulative distribution function and a reference cumulative distribution function. In our case it can be defined as:

$$d(p,q) = \sup_k |F_p(k) - F_q(k)|, \qquad (2)$$

where $F_s(k) = \sum_{j=1}^{k} s(j)$.

Bhattacharyya's distance is defined equal to:

$$d(p,q) = \left(1 - \sum_{k=1}^{N} \frac{\sqrt{p(k)q(k)}}{\sqrt{\sum_{k=1}^{N} p(k) \cdot \sum_{k=1}^{N} q(k)}}\right)^{\frac{1}{2}}. \qquad (3)$$

Using this distance a perfect match is evaluated as 0, whereas a total mismatch is 1.

Intersection distance is equal to:

$$d(p,q) = \sum_{k=1}^{N} min(p(k), q(k)). \qquad (4)$$

The intersection of two histograms is connected to the Bayes error rate, the minimum misclassification (or error) probability which is computed as the overlap between two PDF's $P(A)$ and $P(B)$. If both histograms are normalized to 1, then a perfect match is 1 and a total mismatch is 0.

Correlation is defined as:

$$d(p,q) = \frac{\sum_{k=1}^{N} p'(k)q'(k)}{\sqrt{\sum_{k=1}^{N} p'^2(k)q'^2(k)}}, \qquad (5)$$

where $s'(k) = s(k) - (1/N)(\sum_{j=1}^{N} s(j))$ and N equals the number of bins in the histogram. For correlation, a high score represents a better match than a low score.

Mahalanobis is a distance between an unknown sample and a set of samples which has known mean vector and covariance matrix. Formally given a sample x and a group of samples Y with mean μ and covariance matrix Σ the Mahalanobis distance is:

$$d(x,Y) = (x - \mu)' \Sigma^{-1} (x - \mu). \qquad (6)$$

In our case this distance can be exploited to find the similarity between a frequency vector of a frame p and a set of frames $q_{-n}.., q_{-1}, q, q_1, ..q_n$, where q_{-n} is n^{th} frame before q. In particular n is empirically set to ten.

3. ACTION CATEGORIZATION

Support Vector Machines (SVMs) are a class of supervised learning algorithms, introduced by Vapnik *et al.* [19], that have become extremely popular in the latest years for solving classification problems. In their simplest version, given

a set of labeled training vectors of two classes, SVMs learn a linear decision boundary to discriminate between the two classes that maximize the margin, which is defined to be the smallest distance between the decision boundary and any of the input samples. The result is a linear classification that can be used to classify new input data. An important property is that the determination of the model parameters corresponds to a convex optimization problem, so any solution is a global optimum. In the two classes classification problem suppose to have a training data set that comprises N input vectors $x_1, ..., x_N$, with corresponding target values $t_1, ..., t_N$ where $t_n \in \{-1, 1\}$. The SVMs approach finds the linear decision boundary $y(x)$ as:

$$y(x) = w^T \phi(x) + b \qquad (7)$$

where ϕ denotes a fixed feature-space transformation, b is a bias parameter, so that, if the training data set is linearly separable, $y(x_n) > 0$ for points having $t_n = +1$ and $y(x_n) < 0$ for points having $t_n = -1$. In this case the maximum marginal solution is found by solving for the optimal weights $\mathbf{a} = (a_1, \ldots, a_N)$ in the dual problem in which we maximize:

$$\widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m t_n t_m k(x_n, x_m) \qquad (8)$$

with respect to $\mathbf{a}$, that is subject to the constraints:

$$a_n \geq 0 \quad n = 1, ..., N, \qquad (9)$$

$$\sum_{n=1}^{N} a_n t_n = 0. \qquad (10)$$

where $k(x_n, x_m)$, called kernel function, is defined by $k(x, x') = \phi(x)^T \phi(x')$. The parameters w and b are then derived from the optimal $\mathbf{a}$. The dual problem takes the form of a quadratic programming problem, which can be efficiently solved. Moreover, the SVM approach permits to use kernel techniques, so that the maximum margin classifier can be applied efficiently to a feature space whose dimensionality exceeds the number of data points. Many approaches in image categorization use different kernels such as linear, radial and chi-square basis functions; in particular the latter often gives the best results [7]. These kernels need that all the input vectors have the same length; this fact is a problem when classifying actions contained in a video clip, since the clips have usually different lengths depending on how an action is performed. Unlike other approaches that solve this problem simply by representing the clips with a fixed number of samples [20], we introduce a kernel that deals with input vectors with different dimensionality, in order to account for the temporal progression of the actions. Starting from a Gaussian Kernel that takes the form:

$$k(x, x') = exp(- \|x - x'\|^2 / 2\sigma^2). \qquad (11)$$

we replace the Euclidean with the Needlmann-Wunsch distance. The proposed kernel is:

$$k(x, x') = exp(-d(x, x')). \qquad (12)$$

where $d(x, x')$ is the Needlmann-Wunsch distance between x, x' input vectors. In this approach the structure of the string is evaluated by the edit distance and not by the kernel, that uses only the value of this distance. It has been demonstrated [21] that this type of kernels is suitable for classification of shapes, handwritten digits and chromosome images, despite the fact that the general edit distance has not been proved to be a valid kernel; this is confirmed in our experiments, where all the pre-computed string kernels were checked to confirm their validity. This approach has been compared to a baseline k-nearest neighbors (kNN) classifier, with weighted neighbors, with the Needleman-Wunsch distance instead of the standard Euclidean distance.

4. EXPERIMENTAL RESULTS

We have carried out experiments on our soccer-actions dataset, that is available on request at our webpage [1]. This dataset consists of 100 video clips in MPEG2 format, at full PAL resolution (720×576 pixels, 25 fps), resulting from shot segmentation. It contains 4 different actions: *shot-on-goal*, *placed-kick*, *throw-in* and *goal-kick*. The sequences were taken from 5 different matches of the Italian *"Serie A"* league (season 2007/08) between 7 different teams. For each class there are 25 clips of variable lengths, from a minimum of ~ 4 sec (corresponding to ~ 100 frames) to a maximum of ~ 10 sec (~ 2500 frames). The collection represents a really challenging dataset because actions are featured in a wide range of scenarios. Sequences are taken with different lighting conditions: in particular one match was played in artificial light while four were played in natural light; all the matches were played in different stadiums. Moreover, action classes show an high intra-class variability because even instances of the same action may have very different progression. For each class 20 videos were used as training and 5 as testing set, using a 3-fold cross validation. In the first two experiments we have evaluated the effect of the size of the codebook and of the metrics presented in Sect. 2, on the classification accuracy; the kNN classifier has been used for both the experiments. Finally, the third experiment shows the improvement obtained using SVM string kernels respect to the baseline kNN.

Experiment 1: In the first experiment has been evaluated the performance of the classification while varying the codebook size, using the Chi-square metric; for each codebook size the threshold was varied, recording the accuracy obtained. The best results for each codebook size are reported

---
[1] http://www.micc.unifi.it/vim

Metric	Codebook Size	Threshold	Accuracy
Chi-square	**30**	**0.13**	**0.52**
Chi-square	150	0.4	0.52
Chi-square	300	0.45	0.50

Table 1. Comparison of different codebook sizes.

in Tab. 1. It can be observed that unlike the case of object classification the increase of the codebook size does not improve the performance and, instead, the effect may become negative. This can be explained by analysing the type of views of the sport domain: actions are shown using the main camera that provides an overview of the playfield and of the ongoing action, and thus the SIFT points are mostly detected in correspondence of playfield lines, crowd and players' jerseys and trousers as shown in Fig. 5, and thus the whole scene can be thoroughly represented using an histogram with a limited number of bins for the interest points. Increasing the number of bins may risk to amplify the intra-class variability and then reduce the accuracy of classification, resulting finally also in higher computational costs.

Fig. 5. Example of SIFT points detected in a video frame.

Experiment 2: The second experiment evaluates what is the best distance that has to be used to evaluate the Needleman-Wunsch distance. Following the results of the previous experiment the number of visual words used to build the codebook was set to 30, then the similarity threshold was changed and the accuracy obtained was recorded. Tab. 2 reports the best results obtained for each distance, along with the corresponding threshold. The best distance is the Chi-square, since it has a more uniform performance for the various classes of actions that is not achieved by the correlation metric.

Experiment 3: Finally we have compared the results of the baseline kNN classifier with the results of the SVM classifier with the proposed kernel. The dictionary size used

Metric	Threshold	Accuracy
Bhattacharyya	0.5	0.45
Chi-square	0.13	**0.52**
Correlation	0.7	0.52
Intersection	0.1	0.51
Kolmogorov - Smirnov	0.5	0.49
Mahalanobis	7	0.34

Table 2. Comparison of different metrics used to compare the *characters* (frequency vectors) of the strings that represent the clips.

Fig. 6. Confusion matrix of the baseline kNN classifier.

is 30, the metric used to compare two characters is the Chi-square, following the results of the previous two experiments. The global accuracy obtained by the baseline kNN and by the SVM with string kernel is reported in Tab. 3; the SVM largely outperforms the kNN classifier. Fig. 6 and Fig. 7 report the confusion matrices for kNN and SVM classifiers, respectively. Large part of the improvement in terms of accuracy is due to the fact that the SVM has a better performance on the two most critical actions: *shot-on-goal* and *throw-in*. This latter class has the worst classification results, due to the fact that it has an extremely large variability in the part of the action that follows immediately the throw of the ball (e.g. the player may chose several different directions and strengths for the throw, the defending team may steal the ball, etc.).

5. CONCLUSIONS

In this paper we have presented a method for action recognition based on the BoW approach. The proposed system uses

	kNN	SVM
Mean Accuracy	0.52	**0.73**

Table 3. Global accuracy of action recognition using kNN and SVM string classifiers.

	Placed-kick	Shot-on-goal	Throw-in	Goal-kick
Placed-kick	0.8	00	00	0.2
Shot-on-goal	00	0.8	00	0.2
Throw-in	0.25	0.06	0.63	0.06
Goal-kick	00	00	0.3	0.7

Fig. 7. Confusion matrix of the SVM String classifier.

generic visual features (SIFT points) that represent the static visual appearance of the scene, while the dynamic progression of the action is modelled as a *phrase* composed by the temporal sequence of the bag-of-words histograms. Phrases are compared using the Needleman-Wunsch edit distance and SVMs with string kernels have been used to deal with these feature vectors of variable length. Our experiments show that classification results obtained using SVM and string kernels outperform baseline kNN classifiers and, more generally, they exhibit the validity of the proposed method. Our future work will deal with the application of the proposed approach to a broader video domain, e.g. news videos using the LSCOM events/activities list, and the use of other string kernels.

6. REFERENCES

[1] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, 2005.

[2] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, 2005.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. of ICCV*, 2003.

[5] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proc. of CVPR*, 2003.

[6] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proc. of ACM MIR*, 2007.

[7] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *International Journal of Computer Vision*, vol. 73, no. 2, pp. 213–238, 2007.

[8] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *Proc. of ACM MIR*, 2006.

[9] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. of VSPETS*, 2005.

[10] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," *International Journal of Computer Vision*, vol. 79, no. 3, pp. 299–318, 2008.

[11] F. Wang, Y.-G. Jiang, and C.-W. Ngo, "Video event detection using motion relativity and visual relatedness," in *Proc. of ACM Multimedia*, 2008.

[12] D. Xu and S.-F. Chang, "Video event recognition using kernel methods with multilevel temporal alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, 2008.

[13] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins.," *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443–453, 1970.

[14] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text classification using string kernels," *Journal of Machine Learning Research*, 2002.

[15] C. Leslie, E. Eskin, J. Weston, and W. S. Noble, "Mismatch string kernels for SVM protein classification," in *Proc. of NIPS*, 2003.

[16] J. Assfalg, M. Bertini, C. Colombo, A. Del Bimbo, and W. Nunziati, "Semantic annotation of soccer videos: automatic highlights identification," *Computer Vision and Image Understanding*, vol. 92, no. 2-3, pp. 285–305, 2003.

[17] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 796–807, 2003.

[18] G. Zhu, Q. Huang, C. Xu, L. Xing, W. Gao, and H. Yao, "Human behavior analysis for highlight ranking in broadcast racket sports video," *IEEE Transactions on Multimedia*, vol. 9, no. 6, pp. 1167–1182, 2007.

[19] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. of ACM Workshop on Computational Learning Theory*, 1992.

[20] D. A. Sadlier and N. E. O'Connor, "Event detection in field sports video using audio-visual features and a support vector machine," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 10, pp. 1225–1233, 2005.

[21] M. Neuhaus and H. Bunke, "Edit distance-based kernel functions for structural pattern classification," *Pattern Recognition*, vol. 39, no. 10, pp. 1852–1863, October 2006.