UNSUPERVISED ALGORITHMS FOR SEGMENTATION AND CLUSTERING APPLIED TO SOCCER PLAYERS CLASSIFICATION

P. Spagnolo, P. L. Mazzeo, M. Leo and T. D'Orazio
Institute of Intelligent Systems for Automation – National Research Council
Via Amendola 122/D-I 70126 - Bari, Italy

Keywords: Sport scene analysis, Motion Detection, Background Subtraction, Data Clustering.

Abstract: In this work we consider the problem of soccer player detection and classification. The approach we propose starts from the monocular images acquired by a still camera. Firstly, players are detected by means of background subtraction. An algorithm based on pixels energy content has been implemented in order to detect moving objects. The use of energy information, combined with a temporal sliding window procedure, allows to be substantially independent from motion hypothesis. Then players are assigned to the correspondent team by means of an unsupervised clustering algorithm that works on colour histograms in RGB space. It is composed by two distinct modules: firstly, a modified version of the BSAS clustering algorithm builds the clusters for each class of objects. Then, at runtime, each player is classified by evaluating its distance, in the features space, from the classes previously detected. Algorithms have been tested on different real soccer match of the Italian Serie A.

1 INTRODUCTION

In last years sport applications of computer vision are increasing in many contexts, such as tennis, football, golf, and so on. Many works focus on soccer applications, because of it is very popular, and is often broadcasted on television. Segmentation and team discrimination are key-tasks for such kind of applications.

The research activities in sports video have focused mainly on semantic annotation (Assfalg et al., 2003), event detection (Zhang et al., 2004), generic content structure analysis (LeXing et al., 2004) and summarization (Ekin et al., 2003). The high level applications above mentioned are based on structural low level procedures: the player segmentation, tracking and their classification.

In this work we focus our attention mostly on two aspects of sport image analysis: the segmentation of players from static cameras and their classification in the standard RGB color space.

The correct segmentation of players is fundamental for any further algorithm. The main algorithms for moving objects detection are based on background subtraction; a good review on these approaches can be found in (Kanade et al., 1998) and (Toyama et al., 1999).

Focusing the attention on sport applications, interesting works are based on mathematical morphology: in (Naemura et al., 2000) a segmentation algorithm based on the watershed transform is proposed. In (Misu et al., 2004) two distinct modules (based on background subtraction and chroma-key information) for player segmentation have been implemented as plug-ins in a more complex architecture. In (Xu, Orwell et al., 2005) spectral and topological information are integrated to model the background by means of a mixture of Gaussians. The main drawback of this approach is its dependence from several thresholds manually defined. Thresholds are also critical for the work proposed in (Xu and Shi, 2005). A different approach is proposed in (Mathes et al, 2005): here the goal of the work is the tracking of players, and it appears to work well even in presence of moving cameras. Tracking is also the main argument of works proposed in (Misu et al., 2004), (Xu and Shi, 2005) and (Xu, Orwell et al., 2005).

Another aspect of sport images analysis is the team discrimination. In (Xu, Orwell et al., 2005) color information and local position in the field are

Spagnolo P., L. Mazzeo P., Leo M. and D'Orazio T. (2007).
UNSUPERVISED ALGORITHMS FOR SEGMENTATION AND CLUSTERING APPLIED TO SOCCER PLAYERS CLASSIFICATION.
In *Proceedings of the Second International Conference on Signal Processing and Multimedia Applications*, pages 129-134
DOI: 10.5220/0002139501290134

integrated for the team discrimination. In (Vandenbroucke et al, 2003) a supervised approach based on a manual initialization of the required classes has been used for the player classification. A similar supervised approach is used in (Ekin et al, 2003) but now authors use color histograms to characterized players and referees. In (Yu et al, 2005) SVM is used to assign each segmented object to one of 5 relevant classes manually initialized. Recently, in (Beetz et al., 2006) authors have realized a system that uses the a priori knowledge to analyze broadcast images for segmentation and classification.

All works above mentioned try to solve the problem of player team discrimination in a supervised way, by means of human-machine interaction for the creation of the reference classes. On the contrary, in this work we propose an unsupervised algorithm for the creation of the classes in the scene (players and referee).

Firstly, for the segmentation phase we propose an approach based on background subtraction; we present a novel background modeling algorithm able to build the background model with an unsupervised approach; no assumptions about the presence/absence of foreground objects and changes in light conditions was required. So it is particularly suitable for sport scene analysis, where the constraint of absence of actors during the initialization phase is practically unachievable. The main idea is to exploit the pixels energy information, integrated with a sliding windows procedure, in order to distinguish static points from moving ones.

After the segmentation step, an unsupervised procedure of player classification based on a modified version of the BSAS algorithm will be proposed. In the learning phase, a certain number of segmented player are added to the training set; then, all feature vectors are provided to the clustering algorithm. Its constraints are the max number of output clusters and a threshold on the adopted metric measurement. The algorithm runs and adapt itself in order to obtain the best configuration of output clusters with respect to the application domain. At runtime, after this training procedure, the feature vector is extracted for each player and is provided to the classifier, that assigns it to the correct cluster.

In the rest of the paper, firstly the background model is presented (sec. 2); then, the feature extraction (3) and classification steps (4) are explained. Finally, the experimental results obtained on real image sequences acquired during soccer matches of the Italian Serie A are reported (5).

2 PLAYER SEGMENTATION

The first step of each background-based algorithm is the background modelling. Here we focus our attention on this phase, using the popular background subtraction approach suggested in (Kanade et al., 1998), for the segmentation.

The implemented modeling algorithm evaluates the energy content for each point in a small temporal window: the statistical values relative to slow energy points are used for the background model, while they are discarded for high energy points. In the current temporal window, a point with a small amount of energy is a static point; otherwise it corresponds to a non static point, in particular it could be a foreground point belonging to a foreground object presents in the scene; or a background point corresponding to a moving background object. A coarse-to-fine approach for the background modeling is applied in a sliding window of size W (number of frames). The first image of each window is the coarse background model $B_C(x,y)$. In order to have an algorithm able to create at runtime the required model, instead of building the model at the end of a training period, the mean and standard deviation eq. (1-2) are evaluated at each frame; then, the energy content of each point is evaluated over the whole sliding window, to distinguish real background points from the other ones. Formally, for each frame the algorithm evaluates mean and standard deviation:

$$\overline{\mu^t(x,y)} = \alpha\mu^t(x,y) + (1-\alpha)\overline{\mu^{t-1}} \quad (1)$$

$$\overline{\sigma^t(x,y)} = \alpha \mid \mu^t(x,y) - \overline{\mu^t(x,y)} \mid +(1-\alpha)\overline{\sigma^{t-1}} \quad (2)$$

only if the intensity value of that point is substantially unchanged with respect to the coarse background model, that is:

$$\left| I^t(x,y) - B_C(x,y) \right| < th \quad (3)$$

where *th* is a threshold experimentally selected. In this way, at the end of first W frames, for each point the algorithm evaluates the energy content as:

$$E(x,y) = \int_{t \in W} \left| I^t(x,y) - B_C(x,y) \right|^2 \quad (4)$$

The first fine model of the background B_F is generated, as

$$B_F(x,y) = \begin{cases} (\mu(x,y),\sigma(x,y)) & if\ E(x,y) < th(W) \\ \phi & if\ E(x,y) > th(W) \end{cases} \quad (5)$$

A low energy content means that the considered point is a static one and the corresponding statistics are included in the background model, whereas high energy points, corresponding to foreground or moving background objects cannot contribute to the model. The whole procedure is iterated on another sequence of W frames, starting from the frame W+1. The coarse model of the background is now the frame W+1, and the new statistical values (1) and (2) are evaluated for each point, like as the new energy content (4). The relevant difference with (5) is that now the new statistical parameters are averaged with the previous values, if they are present; so, the new formulation of (5) becomes:

$$B_F(x,y)=\begin{cases}(\mu(x,y),\sigma(x,y)) \;\; if\; E(x,y)<th(W) \\ \qquad\qquad \wedge B_F(x,y)=\phi \\ \beta * B_F(x,y)+(1-\beta)*(\mu(x,y),\sigma(x,y)) \\ \qquad\qquad if\; E(x,y)<th(W) \wedge B_F(x,y)\neq\phi \\ \phi \qquad\qquad if\;\; E(x,y)>th(W)\end{cases} \quad (6)$$

The parameter β is the classic updating parameter introduced in several works on background subtraction (Kanade et al., 1998, and Toyama et al., 1999). It allows to update the existent background model values to the new light conditions in the scene. A dynamic termination criteria is introduced: the modeling procedure stops when a great number of background points have meaningful values:

$$\#(B_F(x,y)=\phi)\cong 0 \quad (7)$$

This approach has been improved in order to distinguish movements of the background objects from foreground objects. Because of in sport scene analysis probably this generalization is redundant, this improvement has not been used in our experiments, so it is not reported here. However this upgrade makes the segmentation procedure more general, and suitable for being applied in each kind of motion detection problem.

3 FEATURES EXTRACTION AND DISTANCE SELECTION

After the segmentation of actors, it is necessary to correctly assign them to the relative classes; to make it, a feature vector is extracted from each actor, and provided to a classification phase. These features must be able to discriminate the different classes of interest. In this way a class is represented by a point in the multidimensional feature-space: the performance of the system is obviously strictly dependent on the selected features. A wrong choice of the multidimensional space can make the data not separable, preventing any useful classification. The selected features need to offer a wide discrimination capability at the minimum computational cost. Histograms in the RGB color space are a good trade-off between these requisites. Color histograms present a lot of advantages, first of all the independence from the posture of players and their scale factor, as remarked in the next sections. In particular, to obtain more stable results with respect to scaling, we have chosen to effect an area normalization of histograms: In this way the histogram shapes remain substantially unchanged; but it is possible to obtain a smoothing of the histograms that can reduce interclass variations. The advantages obtained by normalizing are compensated by the greater proximity of feature vectors in the multidimensional space that makes harder the correct clustering. In fig. 1 normalized histograms of three different actors in a soccer matches are plotted.

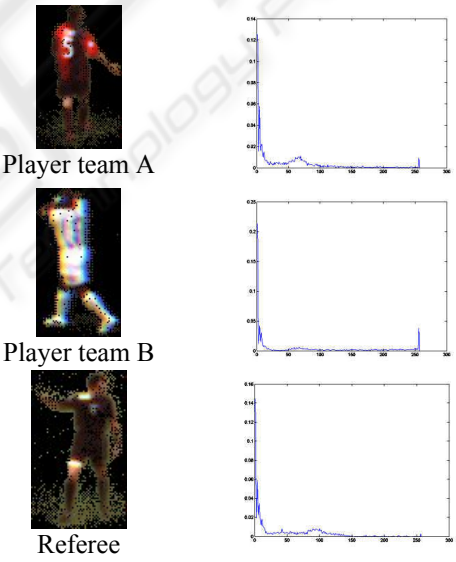


Figure 1: Three patches of different actors, and the relative normalized histograms.

Using the selected features, images become points in a feature space and the class recognition can be seen as a clustering problem. In a generic clustering algorithm it is necessary to define a metric as a way for estimating the "proximity" of two items, according to the selected features. The selected measurement is the Manhattan distance:

$$d_1(\mathbf{x},\mathbf{y})=\sum_{i=1}^{1} w_i|x_i-y_i| \quad (8)$$

where **x** and **y** are the histograms (with values x_i , y_i) and w_i are the weight coefficients. In this context there is no difference in significance of vector elements so all weights have been set to 1. This proximity measurement has the advantage of being simple and fast with respect to other more complex distances (such as the Euclidean one).

4 CLASSIFICATION

The classification procedure is composed by two steps: firstly, during a training phase, the classes are created by means of a clustering algorithm based on a modified version of the BSAS algorithm; it is an unsupervised approach, so it is substantially independent from human interaction. Then, at runtime, each object detected by the segmentation procedure is assigned to one of the classes previously extracted. So, it is clear that the most important step is the clusters creation procedure: an erroneous clusterization of the feature space will provide a wrong classification of actors.

For the clusters building phase, a training set is created by adding a certain number of objects randomly selected. Each object is represented by its normalized histogram, as explained in the previous section; after the creation of the training set, it is examined by the BSAS algorithm with the goal of detecting the interest classes.

BSAS is a very basic and simple clustering algorithm; vectors are presented only once and the number of clusters is not known a priori. What is needed is the similarity measure d(**x**,C), a threshold *th* on this measure, and the number of maximum clusters allowed *q*. The idea is to assign every newly presented vector to an existing cluster or create a new cluster for this sample, depending on the distance to the already defined clusters. More details about this algorithm can be found in [7].

Different choices for the distance function lead to different results and unfortunately the order in which the samples are presented can also have a great effect to the final result. What's also very important is the value of *th*. If *th* is too small, unnecessary clusters are created, and if too large a value is chosen less than required number of clusters are formed.

In our implementation we have lightly modified the classic version of the algorithm. In particular, we have chosen to fix the threshold *th* to a small value, that is increased if the number of detected cluster exceeds the predefined value *q*. In this way the algorithm converges to the correct cluster configuration with the best (smallest) value of *th*. Moreover, in order to smooth the dependence from the order in which the samples are presented, a merge procedure is carried out on the output clusters, using *th* as a merge threshold: if the distance between two clusters is less than *th*, then they are merged, and the clustering procedure is started again. If the algorithm is not able to detect a consistent number of clusters, a new training set is built, and the whole training procedure is repeated. It happens, for example, if the training set in composed only by actors belonging to the same class (players of the same team). At runtime, for each actor selected by the segmentation procedure, the relative feature vector is compared with the representatives of each clusters (also called *prototypes*); the distance (7) is used again to select the winner class. After this, the winner prototype is updated in order to adapt itself to the variations in light conditions; this updating is carried out by means of a weighed mean between the actual value of the prototype for the (winner) class *k* and the feature vector of the classified object, as follow:

$$C_k = \frac{1}{w_k + 1}(w_k C_k + V) \tag{9}$$

where C_k has to be intended as the prototype of the cluster *k*, *V* is the feature vector of the examined objects, and w_k is the weigh of the cluster *k*, i.e. the number of objects classified as belonging to the cluster *k* before then.

This procedure for the classification of players has a great number of advantages with respect to the other proposed in literature. First of all, it is very fast and can be implemented in real time systems. Moreover, it works well in RGB color space, so it is not necessary to convert images in another color space, as HSI. Finally, the proposed approach is totally unsupervised, and no human interaction is required for the creation of the cluster and the classification.

5 EXPERIMENTAL RESULTS AND CONCLUSIONS

We have tested the proposed algorithms on different sequences acquired during real soccer matches of the Italian Serie A, in different conditions, in presence of both natural and artificial lights. The images used for our experiments are provided by a static camera pointed at the middle of the field. So only a max of three classes of objects will be present (player of the two teams, and referee); however, this is a limit strictly related with the experimental setup adopted, and it doesn't depend from the implementation of

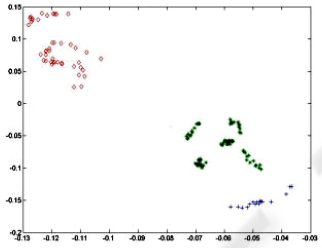
the algorithm, that can work reliability even in presence of a greater number of classes.

Table 1: Evaluation of the capability of the clustering procedure to correctly detect the required clusters.

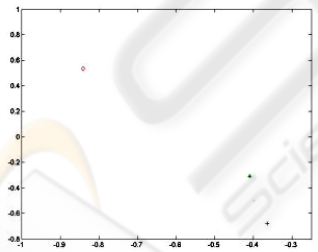
Test Sequence	Training Set Ground Truth	Detected clusters
A	3	3
B	2	2
C	3	3
D	3	2
E	2	2

a b c

Figure 2: examples of actors from the sequence D: player of team A (a), player of team B (b) and referee (c); clusters (a) and (c) are similar, and the algorithm is not able to correctly distinguish them.



a – spatial distribution of feature vectors



b – position of representatives of clusters

Figure 3: The spatial distribution of feature vectors for the test sequence C (a) and the position of the representatives of the three clusters founded by the algorithm

The first experiment regards the reliability of the class detection phase, i.e. the capability of the clustering algorithm to find the correct classes of objects belonging to the training set. In table 1 we propose the output of experiments carried out on 5 different matches (from A to E). As it can be noted, in 4 experiments the algorithm has correctly detected the clusters: the number of detected clusters matches with the number of cluster that composed the training set (ground truth). Only for the sequence D the algorithm finds a number of clusters less than the ground truth. In fig. 2 some patches extracted from the training set for that experiment are represented : due to the light conditions, images are saturated and patches related to the team A and patches related to the referee are very similar, and the clustering algorithm has not distinguished them.

In order to give a visual representation of such experiment, in fig. 3-a we have plotted the spatial distribution of the training set for one of the test sequence above mentioned (in particular the sequence C). It appears evident that feature vectors are well separated, and the clustering algorithm here proposed have found three clusters, as reported in table 1; the positions of representatives of the three clusters in the feature space are plotted in fig 3-b.

The main aim of the second experiment is to give a quantitative evaluation of the performance of the whole classification procedure. To do it, we have manually built the ground truth by labelling actors in the scene on the basis of their class (player team A, team B, and referee). The results proposed in table 2 depending on both the clusters building phase and the effective classification phase (minimum distance criteria). For each test sequence the performances of the algorithms are presented in term of the detection rate (DR) and false alarm rate (FAR), as proposed in several works:

$$DR = \frac{TP}{TP + FN} \quad FAR = \frac{FP}{TP + FP} \tag{10}$$

where TP (true positive) are the actors correctly classified; FP (false positive) are the objects classified as belonging to the examined class while in the ground truth they correspond another class; and FN (false negative) are objects of a certain class not correctly classified. In table 2 we can see the summarization of the results obtained on the test sequences after a manual segmentation of the ground truth. It can be noted that the FAR parameter is always under the 6%, while the detection rate is always over 90%. The worst results have been obtained in the sequence D: as reported in the previous experiment (see table 1 and fig. 2) this sequence is very hard to handle because of the similarity between the shirts of the referee and players of one team, as remarked previously (see fig. 2).

Table 2: Rates to measure the confidence.

Test sequence	Player Team A		Player Team B		Referee	
	DR	FAR	DR	FAR	DR	FAR
A	94.55	2.45	98.33	3.11	93.72	4.33
C	93.34	3.55	97.61	2.98	94.16	3.07
D	82.21	8.83	91.22	3.09	78.26	9.03

In order to give a more detailed analysis of the results, in table 3 the confusion matrix relative to the test sequence D is represented. As evident by examining this table, players of the team B are substantially correctly classified. Otherwise, the players of team A and referee are often misclassified (in the table we have bolded values relative to these misclassification).

Table 3: Confusion Matrix for the sequence D-1 (in %).

Ground truth / Output results	Player team A	Player team B	Referee
Player team A	82.21	5.12	**17.92**
Player team B	4.33	91.22	3.82
Referee	**13.46**	3.66	78.26

This situation is due to the similarity between these two classes, as evident seeing fig. 2: even if for the human observer they are different, for the classifier they are similar; probably it is due to the feature vectors: another feature extraction procedure, more refined, probably could improve the results. In fig. 4 some images from the test sequences of our experiments can be seen. We have used bounding box of different color for each different class.

As a future work, we are testing different kind of features that could provide information about the *spatial* distribution of color instead of the simple *spectral* distribution information typical of histograms. Moreover, we are testing different clustering algorithms in order to select the most reliable for this applicative context.

REFERENCES

Assfalg, J., Bestini, M., Colombo, C., Del Bimbo, A., Nunziati, W. , 2003. Semantic annotation of soccer videos: automatic highlights identification, in *CVIU* 92(2) pp. 285-305.

Zhang, Di, Chang, S.F., 2004. Real-time view recognition and event detection for sports video, in *J. Vis. Commun. Image r.* 15, pp. 330-347.

LeXing Xie, Peng Xu, Shih-Fu Chang, Divakaran, A., 2004. Structure analysis of soccer video with domain knowledge and Hidden Markov Models, in *Patt. Rec. Lett.* 25 pp. 767-75.

Ekin, A., Tekalp, A.M., Mehrotra, R., 2003. Automatic soccer video analysis and summarization, in *IEEE Trans. on Image Processing*, 12(7), pp 796-807

Kanade, T., Collins, T., Lipton, A., 1998. Advances in Cooperative Multi-Sensor Video Surveillance, in *Darpa Image Und. Work.*, Morgan Kaufmann,pp.3-24.

Toyama, K., Krumm, J., Brumitt, B., Meyers, B., 1999. Wallflower: Principles and practice of background maintenance, in *ICCV*, pp. 255-261

Theodoridis, S., Koutroumbas, K., 2003. *Pattern Recognition*, Academic Press, San Diego, ISBN 0-12-686140-4.

Naemura, M., Fukuda, A., Mizutani, Y., Izumi, Y., Tanaka, Y., Enami, K., 2000. Morphological Segmentation of Sport Scenes using Color Information, in *IEEE Tr. on Br.*,46(3) pp.181-8.

Misu, T., Gohshi, S., Izumi, Y., Fujita, Y., Naemura, M., 2004. Robust tracking of athletes using multiple features of multiple views, *Journ. of WSCG*, vol.12, 1-3

Xu, M., Orwell, J., Lowey, L., Thirde, D., 2005. Architecture and algorithms for tracking football players with multiple cameras, in *IEE Proc. Vis. Im. and Sign. Proc*,152 (2) pp.232-41.

Xu, Z., Shi, P., 2005. Segmentation of players and team discrimination in soccer videos, in *IEEE int. Work. VLSI Design & Video Tech.*, May 28-30, Suzhou, China.

Mathes, T., Piater, J., 2005. Robust Non-rigid Object tracking using Point Distribution Models, in *BMVC*

Vandenbroucke, N., Macaire, L., Postaire, J.G., 2003. Color image segmentation by pixel classification in an adapted hybrid color space. Application to soccer image analysis, *CVIU*(90), 2, pp. 190-216.

Ekin, A., Tekalp, A.M., 2003. Robust dominant color region detection and color-based applications for sports video, in *ICIP* (1), pp. 21-24.

Yu, X., Sen Hay, T., Yan, X., Chng, E., 2005. A Player-Possession Acquisition System for Broadcast Soccer Video, in *ICME* July, 6-8, Singapore, pp. 522-525.

Beetz, M., Bandouch, J., Gedikli, S., 2006. Camera-based Observation of Football Games for Analyzing Multi-agent Activities, in *Proc. of AAMAS*, pp. 42-49.

Figure 4: Output images after the classification phase: boxes of the same colours refer to players classified as belonging to the same team.