Soccer players detection and team classification using histograms and SVM

Mattia Rigon

[a]Department of Information Engineering and Computer Science, University of Trento, Trento, Italy

Abstract

In the realm of sports analytics and performance assessment, the application of advanced technologies has become increasingly indispensable. This project addresses the growing need for a comprehensive system that leverages image and video processing techniques to analyze soccer matches efficiently. More in detail has been developed a software which is able to perform player detection, team classification and calculate the position of each player detected in order to compute some soccer analysis. An illustrative example of analysis will be presented, showcasing the creation of a heat map depicting the positions of each team.

1. Introduction

This project has been developed in order to analyze videos that have some characteristics:

- Fixed camera.
- Far point of view of the camera.
- One camera for each midfield, that produce a 4K resolution video.

First of all, in order to compute the detection of a soccer player it's necessary to reduce the area where apply the detection only inside the soccer field. Once the field has been detected, it's crucial find the position of the lines and the stationary items/person inside the field. Then it's time for the soccer player detection, and classification. Thereafter, the soccer player's posion is mapped into a two dimension area in order to be visualized inside the soccer field and compute the heatmaps of the teams during the match. Those heatmaps describe for each team where their players stay most of the time during the match; at the end, another general map will show the sum of the two teams' heatmaps.

Figure 1: Frame from one camera point of view

2. Soccer Field Detection

Several techniques have been tried in order to automate the detection of the soccer field, but at the end it has been decided to manually select the point that bounds the field in order to avoid any problem. In fact, an error in this step would compromise all the mapping in the two dimension space and all the future stages. In support of this, since the camera are fixed, once you have done the configuration for one stadium you can be sure that this step will work out in the right way.

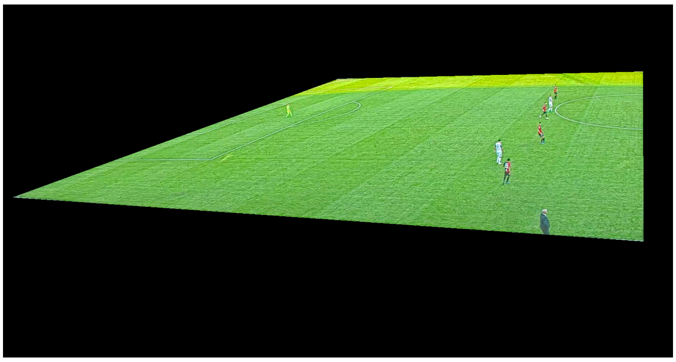

Figure 2: Isolation of soccer field

3. Soccer Player Detection

3.1. Filter on colors

In order to detect all players, the first step consists of applying a filter to each frame. Each frame is filtered on hsv space, then it is converted into a gray scale image and subsequently into a binary image: white pixels correspond to non green pixels and black pixels to green ones. From each frame the green pixels are then removed, but still it is necessary to properly manage the lines, some possible rumors or staonary items (like cameras place at sidelines of the playground) as well.

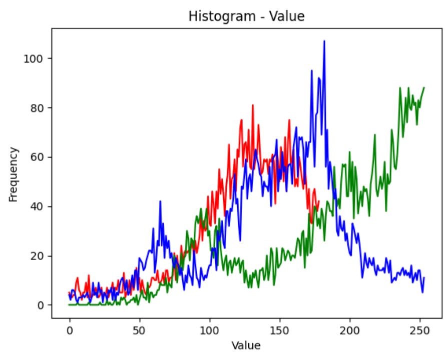

Figure 3: Hsv histogram.

Figure 5: Binary image after have removed lines

3.2. *Playground lines and stationary objects*

Therefore when a video is analyzed for the first time are picked N (tested with N=50) frames randomly. These frames will be used to determinate the stationary non green points and generate the dataset on which all the classification steps will be applied. Once this set of frames is generated, the pre-process that has just been explained is applied to each frame. After that, a binary image is created using the following policy: for each pixel a ratio which says how many times a pixel was white in the binary-randomly-generated set of frames is calculated. Taking into consideration the pixels of the just-generated image, it is then possible to determine whether a pixel is white or black: if its ratio is greater than 80% it is white, otherwise it is black. In this way, we obtain a good estimate of the videos' stationary points which generally represent the lines of the playground.

The process of lines detection is performed the first time that we analyse a video, the results are then saved in order to improve its performance the following times that we analyse that video; in this way we perform the detection of lines or the upload just at the startup of the program. Once the lines and the stationary points have been detected, the lines' binary image will be subtracted. In order to detect the lines, several approaches have been tested during the development of this project. The approach that mostly deserves our attention is the combination of canny edge detection and Hough transform. This approach has been rejected mainly because the detection of the curved lines was very hard to perform, while the one using stationary points has been proved to be more effective instead. However, if the analysed video has no more fixed cameras, it is necessary to look for another approach.

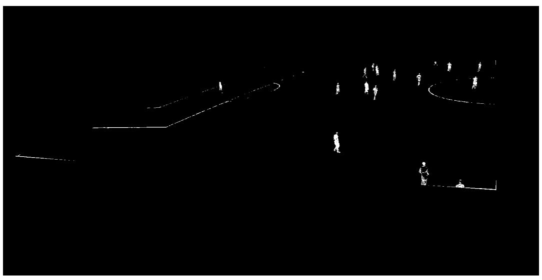

Figure 4: Binary image with lines

3.3. *Player detection*

Once the binary image that identifies the players through white pixels has been obtained, it is time to identify the players of the game. In order to do that, the contours technique (Contours can be simply defined as a curve joining all the continuous points along the boundary which have the same colour or intensity) has been employed. After the detection of contours, a rectangular bounding box of the item in question is generated for each contour. Then, a filter is applied to the dimension of this bounding box and all smaller objects are eliminated. At this point, the rectangular bound generated on the binary image is reported in the original one and it is therefore possible to classify it.

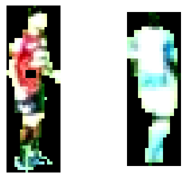

Figure 6: Players

4. Team Classification

In order to classify properly each bounding box of a player, it necessary to define a metric for compare images. Once the metric has been defined, we can compute distance in some space and this allows as to apply a machine learning technique in order to classify each player. Since colour is the most discriminant feature when classifying team membership, the histograms in the hsv color space has been chosen as the value to describe each player image. A very important step to compute in order to obtain a good result is the normalization of the histograms, since the images that it classifies will have a different dimension.

4.1. *Dataset generation and model training*

As is known machine learning is divided in two main branches with regard to classification: supervised and unsupervised. The purpose of this project is use a SVM (Support vector machines) in order to classify players, because it performs a fast classification and has some flexibility adaption in the soft margin case. SVM, as all supervised machine learning technique, needs labeled data. Since we do not want to do manually all labels for each player image and the project's goal is not to use pre-trained models or some advanced machine learning techniques developed in other previous projects, a solution that uses an unsupervised technique has been developed in order to automatically generate a dataset. On top of this automatically generated dataset will be trained the SVM. This solution can be used without any human intervention, but, if the unsupervised method did a bad classification, the human intervention could be easily introduced just moving some wrong labelled images from one folder to another. In this way, we have created the labelled dataset for our SVM model.

Dataset generation and the unsupervised classification are performed only the first time that we analyze a specific video. As for the detection of the lines, are used the same random generated frames on which will be applied player detection, these set of players detected between this frames will be subject to unsupervised classification and the creation of the dataset. If the program finds the dataset associated with the video that it is processing, it will skip the dataset generation and will only perform SVM training.

In the case of SVM, a SVM with soft margin for large dataset with $C = \frac{1}{\text{len(dataset)}}$ has been chosen. With regard to the unsupervised technique, Kmeans and Gaussian Mixture Model have been tested and among them the Kmeans method has been proved to be the most effective one.

Figure 7: Detection results

5. Two dimension space mapping

Once that the detection and the classification steps have been applied inside the video in input, it is time to calculate the position that each player has inside the soccer field. In order to perform this space transformation, it is necessary to identify the 4 points that bound the soccer field and the 4 corners that will determinate our field in two dimensions. Once we have the starting and ending points, we can use the Open CV getPerspectiveTransform in order to calculate the transformation matrix which will be used inside the warpPerspective that will then fit our field inside a rectangular. After having idenfied the position of each player inside this new image, we can easily transform it and draw it inside a soccer field image.

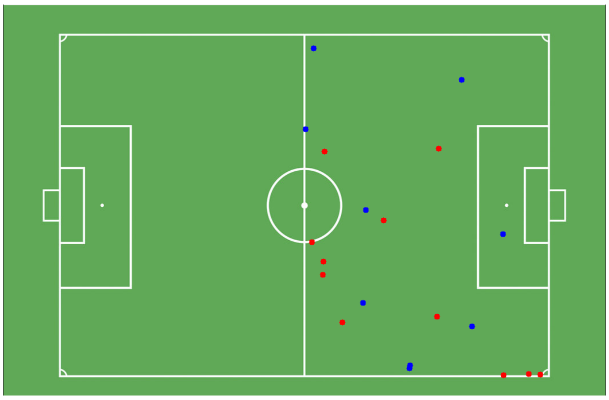

Figure 8: Two dimensional mapping

6. Heatmaps generations

Given the results and process that it's been applied, it is possible to perform a lot of soccer analysis. In order to give a concrete example of an analysis that makes sense also with the errors and accuracy of this project, the so-called team heatmaps have been chosen. A soccer heatmap is a graphical representation that visually displays the distribution and intensity of player movement or ball possession across the soccer field during a match. It uses colour gradients to indicate areas where players or the ball spend more time, providing insights into team strategies, player positioning, and the flow of the game. Heatmaps are valuable tools for coaches, analysts, and enthusiasts to analyze and understand the spatial dynamics and patterns within a soccer match.

In order to generate these heatmaps it has been created a matrix of the same dimension of the field and each players in a position (x,y) update this matrix with an increment, at the end of the video the heatmap is generated according to this matrix. It's manage a matrix for each team in order to be able to generate 3 heatmaps; one for each team and another one which is the sum of the 2 teams heatmaps.

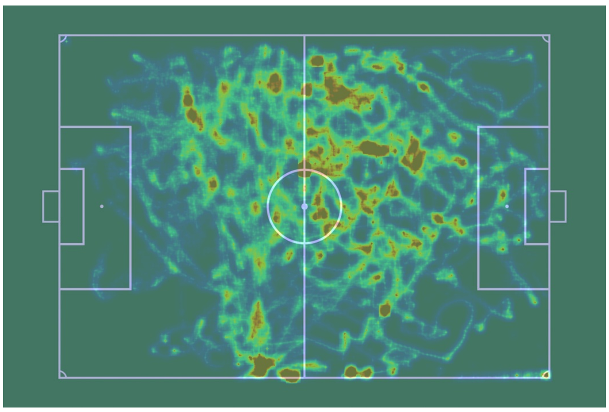

Figure 9: Team heatmap.

7. Problems and possible improvement

The solution which has been applied in this project might have some problems in some specific situations, and those problems could be fixed in the future. One of the main problem is the detection of the manager or players who are warming up on the line of the field. The techniques which have been adopted are not able to distinguish the manager or the referee from the players. A solution that might solve this problem is labelling new classes and training the SVM also with these new classes.

A more accurate calculation of the position of the players could also be necessary if we want to perform another type of analysis that requires precision, such as offside detection.

At the moment object occlusion and object tracking are not implemented; however, these two implementations will be an important improvement to this project since, when two players are overlapping each other, one of the them is not detected. Moreover, the object tracking could also be useful in order to perform personal soccer player analysis, such as personal distance made or similar things, and could be an important improvement in the classification process, avoiding rapid changing of predicted labels.

8. Summary and conclusions

In conclusion, it is possible to state that the technique that has been tested inside this project has been proved to be effective in order to compute soccer player detection and classification. Moreover, this project can be easily improved and extended in order to obtain results that could generate even more sophisticated analyses.