Proof of concept of an intelligent system for basketball understanding and advanced statistics collection

Di Feo Cristian 160327, Mozzillo Angelo 160313, Viglianisi Brandon Willy 156662
294540@studenti.unimore.it, 294526@studenti.unimore.it, 232325@studenti.unimore.it
University of Modena e Reggio Emilia, Submitted 23-07-2021

Abstract

The project aims are to analyze the feasibility of an intelligent system capable of understanding the game of basketball and collect statistics through visual data. The main purposes will be to recognise the players, understanding their team membership and localizing them in the field. To achieve these goals we will test a subset of non-deep and deep algorithms seen in our lessons in order to finally understand their strengths and limitations performing our tasks.

1. Introduction

The goal of this project is to test and analyze the pros and cons of different procedures to resolve the task of detection and identification of basketball players in videos. It is necessary to say that we have used different algorithms focusing more on the intention of getting hands-on knowledge, rather than aspiring to obtain the best possible result.

1.1. The approaches

The implemented techniques rely on two different approaches to the problem:

- A deep end-to-end implementation;
- A "classical" approach made of multiple steps.

This was seen as necessary by our team, both to respect the project requirements and also to achieve good results which are unattainable with a classical approach considering the timing constraints given and the intrinsec complex semantics of a sport like basketball.
With this being stated let's start with a more technical discussion.

2. The classical approach

We will start our study with the classical approach approach that will cover the following steps:

- Court detection using image processing and the edge based Hough lines algorithm
- Player detection adopting HOG detector
- Team identification testing BoW and SVM
- Player identification applying retrieval on a histogram feature using EMD as distance measure
- Homography from 3D to a 2D space using the geometric transformations

It's important to keep in mind that at this stage we have prioritized the analysis of the different steps, without thinking about the union of the different parts to create a complete pipeline in order to solve the entire problem. So the different steps were addressed without thinking about the link among the different parts. Sometimes it has been useful in terms of time to work on single images instead of videos.

2.1. Court Detection

The detection of the court lines is a crucial point of the approach, indeed it allows us to distinguish what is on the field and what is out of it. This will be very useful in the next step when this information will be used to discriminate between players and members of the audience. Last but not least the information of the position of the court will be used to determine the computation of the homography matrix that will transform the 3D input into a 2D output given the fact that the dimensions of the field are known.

The final implementation of this step, whose results can be seen in *fig.1*, is the child of a long and painful journey that has involved different variations of preprocessing before applying the Edge based Hough lines algorithm.

The Canny algorithm was chosen to produce the edges and right from the beginning it became clear that the success of the approach was played in the choice of the right parameters of Canny and Hough. In order to solve this dilemma, we've used what for us was our only prior knowledge: the number of lines that we want to obtain and the belief that should be the strongest lines on the image in terms of accumulation in the Hough space. So we have implemented a brute force algorithm that iterates by changing the values of the low and high threshold of Canny and the value of the threshold of Hough in the right

direction to achieve our goal of obtaining only the three strongest lines.

The pre-processing techniques used were various. We've tested different techniques including the use of mask calculated in the HSV colorspace, the use of morphology transformation such as dilation and erosion and at last also the adding of a gaussian blur to prevent spurious individuation and reduce errors induced by noise.

The last step involved the use of a k-means based algorithm to group the lines based on their angles and finally find their intersections.

fig. 1 - Court detection result

2.2. Player Detection

For Player Detection the approach uses HOG Detector combined with pre-trained SVM, provided by OpenCV, in order to recognise people.

For the detection, a standard approach was initially adopted using a sliding window 8 * 8 combined with a scale factor of 1.5. In fig. 2 the poor results obtained.

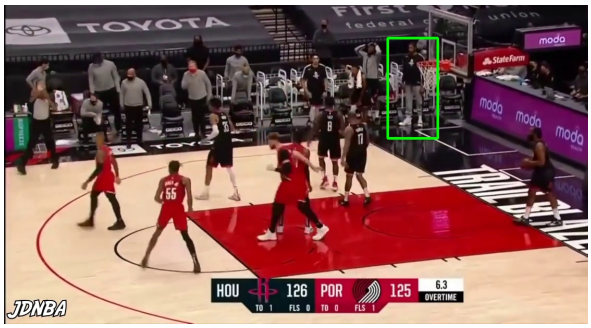

fig. 2 - Player Detection, first phase

The approach's success rate, after a series of tests, has improved by identifying the parameters:

sliding_window: 4 * 4

scales: 1.05

This increased the success rate but allowed the algorithm to recognise, in addition to the players, other people not connected to the analysis (the public, the coaches, etc.).

In combination, to overcome the problem of overlapping bounding boxes on the same person, Non-Maximum Suppression (NMS) was used with a threshold of 0.3.

In fig. *3 left*, the approach before the NMS while in *fig. 3 right* the approach after the NMS.

fig. 3 sx - Player Detection before NMS

fig. 3 dx - Player Detection after NMS

Finally, the results of this step are shown in *fig. 4*.

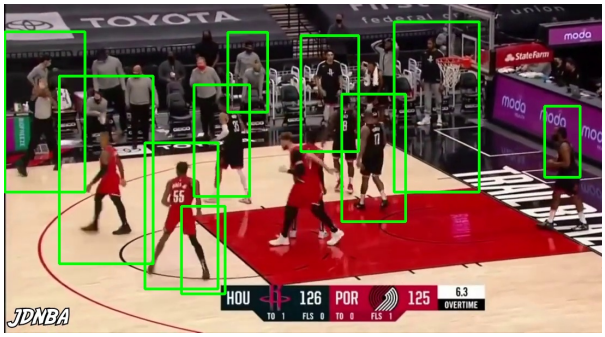

fig. 4 - Player Detection result

2.3. Team identification

For Team Identification we used Two approaches, the first uses color information while the second is based on BoW and SVM.

The first approach uses color to identify the class to which a person belongs; in particular, three colors are initialized which represent the classes they belong to: the color of the home team, the color of the opposite team and finally a high contrast color to indicate all the people belonging to a category unrelated to the players in the court.

To define the class to which a person belongs, we analyzed the bounding boxes, which returned in the previous step, and for each of them the label was assigned considering the minimum difference between the average of the bounding box color and the predefined colors. This approach, although very basic, reaches good results, for example in *fig. 5*, especially in the case in which the colors of the two teams are in high contrast; and this is precisely its weakness, since in the case of low contrast colors it does not succeed very well. Furthermore, the initialization of the main colors must be done manually, because we were unable to identify a good automatic approach.

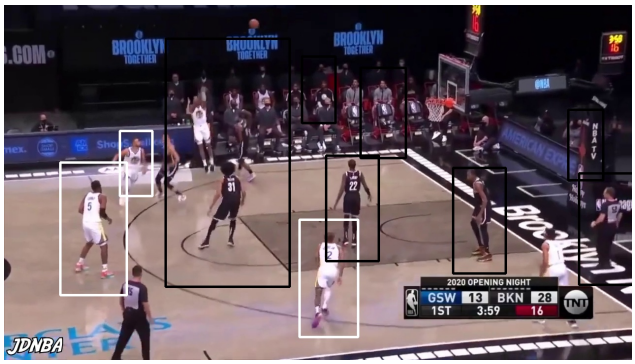

fig. 5 - Team Identification Result

Instead, regarding the approach with BoW and SVM we started creating a small dataset of roughly 80 images of bounding boxes of players labeled each with their corresponding team. The first step is then, following the steps of the bag of word algorithm, to use a keypoint detector and descriptor, in our implementation we choose SIFT [5], to find and cluster the descriptors in order to construct a vocabulary capable, with the occurrences of the visual words in an histogram, to describe each image in our dataset. Knowing the label of each image we are able to use an SVM algorithm to find the line or the curve (we tested booth a linear and a nonlinear approach) with the maximum margin that is able to divide and so to classify the images according to the team.

In figure 6 are shown some of the results:

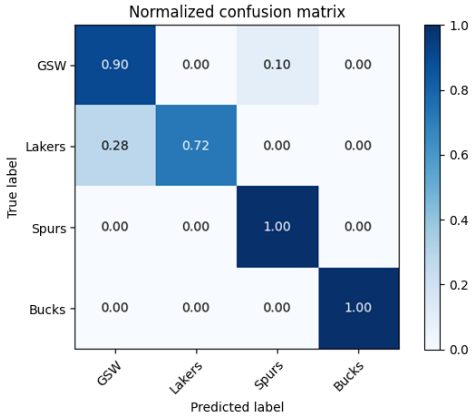

fig. 6 - Normalized confusion matrix
Test accuracy score: 0.903
SVM parameters: C=0.5; gamma = 0.1

As you can see from the confusion matrix the accuracy achieved was satisfactory. We can say that starting from the bounding boxes of the players using an algorithm that uses SIFT, BoW and SVM we are able to successfully discriminate each player team. Obviously this approach has also its cons: we have to make a dataset of each team to find our answers, and considering that in a medium size championship we deal with 25 teams and each has at least 3 jerseys of different colors, so the job is not easy. But if this can be done the results that we have seen are very promising.

2.4. Player retrieval identification

Since a retrieval implementation was required in the project our team decided to implement the identification of the player using this technique. There are a lot of alternatives, even much easier, exist for this job, but again we're not here to revolutionize this field, our aim is to test and discover the pros and cons of these implementations. So first we constructed a database of label images of famous basketball players so that more photos were available. We decided to use photos from the back of the athlete with the belief that this was the most discriminant part of our players. After this, with a similar approach done in the team identification we built, using BoW, a dataset containing for each image an histogram representation that counts the occurrences of visual words of our vocabulary and we save it offline, this was useful to do the retrieval in a fast way.

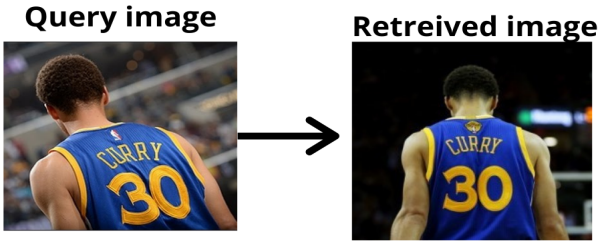

fig. 7.1 - An example of retrieval

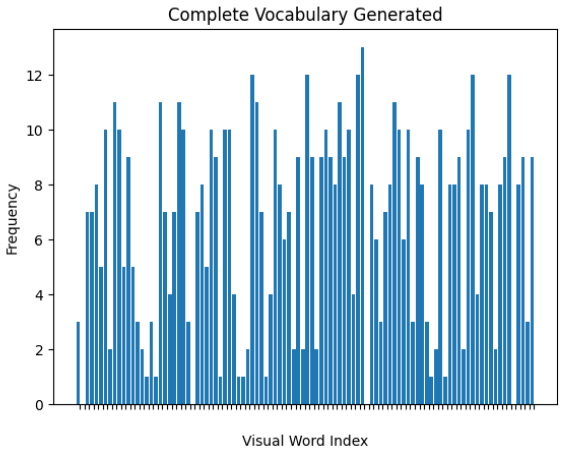

fig 7.2 - The complete visual words vocabulary

Done this the only thing left is to find the corresponding representation of our query image and choose a proper

distance measure to compare the query with our data. After some tests we decided to use the Earth Mover's Distance [6]. As a matter of fact this measure is very flexible and suited for the job and has shown in our test the best results. This is due to the fact that it measures the differences in the histograms and the amount moved for the distance moved to make the two histograms equals. This is very useful when we deal with histograms with similar but shifted distributions which would be seen differently by a classic distance that compares bin to bin. The tests performed showed an accuracy of 75%, that considesing our limited dataset is a good starting point to demonstrate the effectiveness of this approach. But it would be very hard to achieve higher performances, the problem affecting the technique is, in fact, well known. Using keypoint descriptors we are losing the spatial information that isn't taken into account during our computations. This is the true limit of this class of algorithms that will eventually limit their final performances.

2.5. Players Detection to 2D using Homography

The goal of this section is to use the baseline and sidelines found using the approach explained in the second section, and the position of players obtained thanks to the approach of the third section of this paper, for projecting the analyzed action on a 2D court using homography.

In simple terms, homography maps images of points on a plane of the world from one camera view to another. It is a projective relationship because it depends only on the intersection of planes with lines.

Homography can be calculated using relative rotation and translation between two cameras. But in our case, we don't have the relative pose between two views and we need to calculate the mapping between the first 3D match image and the second 2D court image to show the players position.

Our problem can be solved through a perspective correction of the 3D image, so we use the points correspondence (x, x') to calculate the homography and project the points to the 2D court.

First of all, the court detection approach [Cap 2.1] is used to find the first two points of intersection between baseline and sidelines, and through the parallelogram geometric formulas the remaining points are found that close the polygon, managing to obtain the four approximate points of half court analyzed *(fig. 8)*.

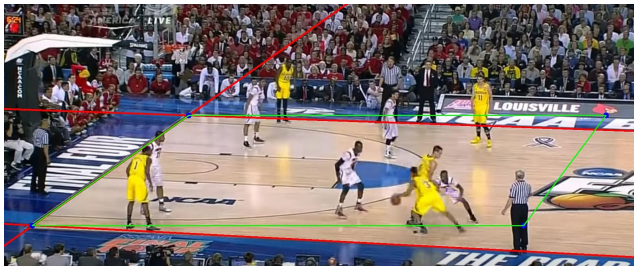

fig. 8 - Court Identification Result

Through the yolov5 model [Cap. 3] we identify the players boxes, from which the points where the player is positioned on the map are obtained.

fig. 9 - Identification of Player Coordinates

Once we identify the four points of the court in 3D, they can be used to correct the perspective to be able to project them on the 2D court, on which the points of the half court concerned have been manually fixed.

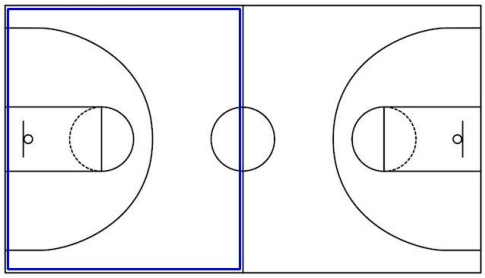

fig. 10 - Delimitation of the court

In order to do this, we used the functions made available by the OpenCV library, namely findHomography() to calculate the homography matrix that is given as input to the warpPerspective() function to correct the perspective distortion of the image.

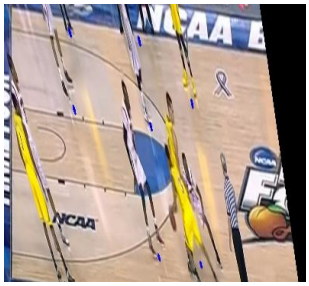

fig. 9 - Homography

The output is filtered using masks to find the positions of the blue points of the players mapped in 2D using the findContours() function in order to enter the coordinates in the image of the 2D court, obtaining the final output shown in the figure.

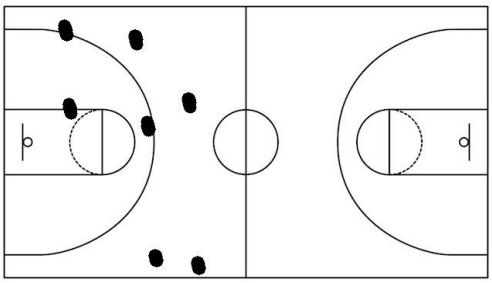

fig. 10 - Team Identification Result

Our purpose is to design the positions of the players identified in the court in 3D on the court in 2D and for this reason the homography is perfectly suited to our needs.

The approach is too supervised, the images it receives must all be in the same half court, the scene must be shot in such a way that the baseline of the court is clearly visible and the players must not cover it otherwise the filters cannot locate the two points of intersection.

2.6. Future improvements

One of the possible court detection applications, using homography, is to create a video of the complete action within the 2D court so that it can be used for any analysis of game tactics.

Certainly, among the achievable future improvements there is a pipeline that includes all the steps able to start from a video, recognize the court, identify the players and the team they belong to and finally build a 2D perspective of the action being played.

3. The deep end-to-end approach

The deep approach focuses on the use of a custom YoloV5, a neural network for object detection.

In particular, this approach was used for the player detection and team identification phases.

The first step, therefore, was to detect the players using the pre-trained weights on the COCO dataset using only the images with the “person” label. This made evident the presence of many false positives, for example the people in the stands, the coaches, the referee, etc. .. We therefore decided to train YoloV5 on another dataset containing two types of "Person" and "Human body" classes by taking 1800 images from Google Open Image and returning them in the format expected by YoloV5 using the “fiftyone” tool.

The results didn't improve, so the next step was to build a dataset closer to our needs, containing 500 images of basketball games, and label it by hand, using the LabelImg tool.

We chose the labels:

- "Player1": to indicate the players of the first team,
- "Player2": for the players of the second team,
- "Ball": for the ball,
- “Basket”: for the basket.

The label ‘ball’ was inserted to use it in order to identify the possession of each team, in accordance with the identification of the team.

Instead, the label ‘basket’ was inserted to make game statistics on the match (number of baskets made / wrong).

Once we obtained the dataset, we trained YoloV5 locally using the YoloV5s model, obtaining the weights close to our purpose.

The weights obtained from the training phase allowed us to obtain an excellent detection by eliminating a priori all the people not connected to the analysis and a fairly effective identification of the players per team.

For the training of YoloV5, the images were resized to 640x640 pixels, using a batch size of 32, for 100 epochs.

3.1. Results

In this section we have to take into account that we used 500 labeled images for training our model, and that it takes at least a thousand of images on average to get good object detection.

We reported the results obtained by testing the network on 100 images with the respective labels to evaluate its correct working.

First of all, considering the player detection, between "player1" and "player2" on 854 test instances the model identifies as many as 512, thus obtaining a 60% success.

As for the detection of the ball, we obtained very poor results due to the absence of the ball within the 500 training images. We were already in the worst-case having few training images, so the results reflect what we expected.

On the contrary, the basket, present in most of the

training images and which is not affected by the context structure, is identified with an 89% success.

Finally, the mean Average Precision of 55% suggests that the model predictions reflect the limitations given by the training dataset.

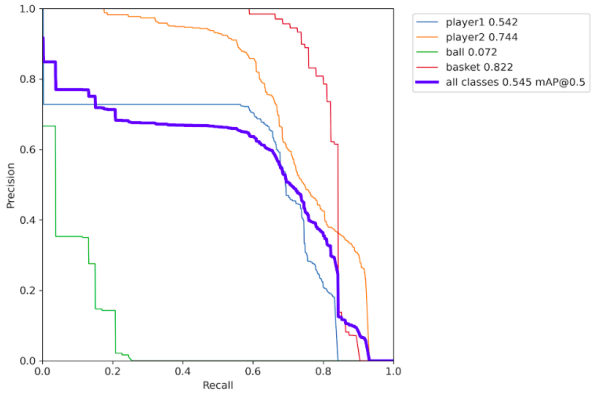

fig.11 - Precision and Recall Curve

Detection Performance					
	Player 1	Player 2	Ball	Basket	Overall
TP	236	276	18	84	610
FP	157	186	34	10	392
Precision	0.60	0.60	0.35	0.89	0.61
Recall	0.67	0.70	0.13	0.76	0.57
mAP@.5	0.54	0.74	0.07	0.82	0.55

Table 1 - Detection Performance

3.2. Future improvements

- Test a 2D + 1 neural network for action recognition
- Use a deep network with triplet loss for player detection
- Improve the quality and size of datasets

4. Conclusions

This work has given us the opportunity to understand the difference between the approaches that use classical techniques and the deep approaches. Despite the low quantity of the available data, we obtained good results according to the deep approach and acceptable results using classical techniques.

For the classic approach, it was interesting to understand how to change the parameters to obtain the best possible results; however some tasks are still difficult to solve, first of all a good court detection even in conditions of poor visibility of the court lines, or the still imperfect team identification and finally the 2D player detection. This is because the tuning and execution of algorithms of this type is extremely complex in the face of a chaotic environment such as that of a basketball court.

Instead for the deep approach, we are able to observe the power of these networks and despite the poor quantity and quality of data they proved to be more than valid for the detection of players and teams.

On the other hand, the computational power required for the training of these networks is to be taken into consideration, just think that in our case the training took 28 hours. However, the intrinsic nature of these means is clear in us, capable of elegantly and compactly translating complex semantics such as sports.

The conviction remains predominant in us that having the opportunity to understand the strengths and weaknesses of handcrafted and deep algorithms is certainly the greatest result of this work, a result that we will bring as a team and as individuals for the rest of our student and working career.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, pp. 886-893 vol. 1, doi: 10.1109/CVPR.2005.177.

[2] [Online] Available: https://github.com/ultralytics/yolov5.

[3] Tzutalin. LabelImg. Git code (2015). https://github.com/tzutalin/labelImg

[4] Moore, B. E. and Corso, J. J., GitHub. Note: https://github.com/voxel51/fiftyone

[5] Lowe, D G 2004 Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2): 91--110. DOI: https://doi.org/10.1023/B:VISI.0000029664.99615.94

[6] Rubner, Y., Tomasi, C. & Guibas, L.J. The Earth Mover's Distance as a Metric for Image Retrieval. International Journal of Computer Vision 40, 99–121 (2000). https://doi.org/10.1023/A:1026543900054

[7] Stephan Janssen. The journey towards creating a Basketball mini-map(2019). Linkedin: The journey towards creating a Basketball mini-map (linkedin.com)

[8] Basic concepts of the homography explained with code. OpenCV:docs.opencv.org/master/d9/dab/tutorial_homography.html