Augmented Reality Applied to Images of Football Games

Diogo Filipe Dores
MIEIC
FEUP
Porto, Portugal
up201504614@fe.up.pt

José Pedro Borges
MIEIC
FEUP
Porto, Portugal
up201503603@fe.up.pt

Miguel Mano Fernandes
MIEIC
FEUP
Porto, Portugal
up201503538@fe.up.pt

***Abstract*—This report describes the development strategies behind the Augmented Reality Applied to Images of Football Games project. It was developed using OpenCV and Python. The algorithm used is described thoroughly in the sections presented. The results of the execution of the developed program are shown in the end of the report.**

***Keywords*—Augmented Reality, OpenCV, Python, Homography, Football, Soccer, Players, Python**

I. Introduction

The main goal of the project is to develop a program that inserts into static images of football games visual marks, such as an offline line or a circle that denotes the minimum distance that players should be from a free kick as well as the distance from one point of the field to another. As such, our project is divided into three modes:

The first mode is to insert an offline line in a location selected by the user. The second mode is to insert a circle with a radius of 9.15m around a selected point. And the third is to determine the distance between a selected point and the middle of the goal, but any two points may be selected.

The implementation of these modes will be detailed in the next section.

The technology used for the development of this project was OpenCV with the Python programming language.

II. Proposed Solution

The final solution implemented for this project was to create a single program that could run each of the three modes mentioned in the section above. For that, the user initially has to manually select 4 out of 12 available points in order to create a valid homography for the resolution of the presented problem.

For the sake of letting the user know the order of the selection of the points for the homography, an image, designed by the group, representing a top view of the goal and penalty area of one half of the field, is displayed. There are twelve points detailed in the image, that correspond to ten vertices of the areas and two corner locations. Next to each one of these points is displayed an index, which represents the order by which the user must select the points in the real image in order to obtain obtain the homography.

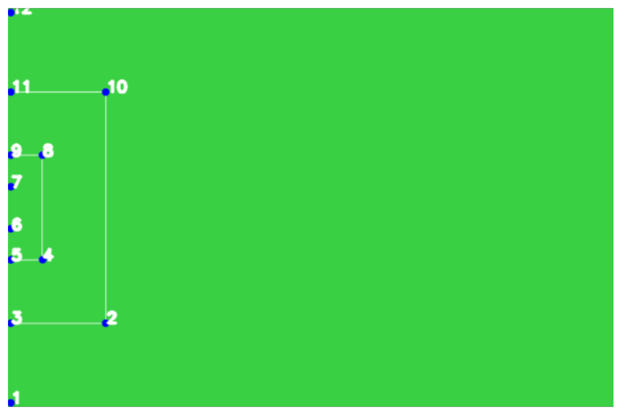

Fig. 1. Image created by the group

The user should select all visible points by clicking the left mouse button. If, by any chance, a point is not represented in the field or is covered by a player, the user can click the right mouse button to ignore that point and proceed to the next one.

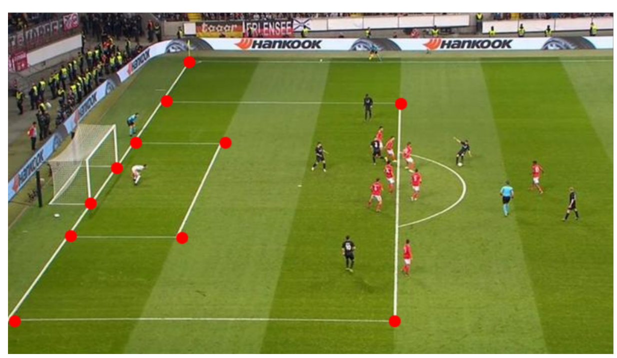

Fig. 2. Example of an image input

The procedure detailed above needs to be executed for each mode. Using the points selected by the user, a homography matrix is created. Regarding the necessary steps in each mode:

Fig. 3. Applied homography

A. Offside

For this mode, the user is prompted to select the player that is offside. After selecting it, its coordinates are transformed into the equivalent coordinates in the newly obtained homography. An array is created with the coordinates of the top-most and bottom-most points in a vertical, straight line that intersects with the coordinates of the player. The inverse homography matrix is then applied to these coordinates, obtaining the final coordinates. A line is drawn, connecting them. The final image is shown with an offside line applied.

B. Free Kick

For this mode, the user is prompted to select the ball so that a free kick area can be established. After selecting it, its coordinates are transformed into the equivalent coordinates in the newly obtained homography. An array is created with the four coordinates of a box with 9.5 * 2 meters of length. The inverse homography matrix is then applied to these coordinates, obtaining the final coordinates. An ellipse is drawn inside the box created. The final image is shown with an 9.5m wide ellipse surrounding the ball.

C. Distance to goal

For this mode, the user is prompted to select a player and the middle of the goal. After selecting them, its coordinates are transformed into the equivalent coordinates in the newly obtained homography. Using these coordinates, the distance between the two points is calculated. The inverse homography matrix is then applied to these coordinates, obtaining the final coordinates. An arrowed line is drawn connecting the two points. The final image is shown with the mentioned line as the distance to the goal, in meters.

III. Results Obtained

After the execution of the program is completed, the necessary geometry is always drawn on the image of the field. This geometry is always displayed in the correct angle relative to the field in the image, regardless of the mode selected. It is also drawn in a way that does not overwrite most of the players on the field. Regarding the second mode in particular, the circle is supposed to have a radius of 9.15 meters and this measurement is accurately portrayed by the program. All circles that are drawn appear to have that radius, when compared to the relative size of the field. In the third mode, all distances also appear to be accurately calculated.

A. Problems and solutions

Regarding the problems during the development of this project, the most notable were:

Difficulties obtaining the green color range of the field in order to isolate the players from the field, since these could vary in many different ways. Eventually, a satisfactory color range was found, although some unnatural color artifacts appear sometimes over a player, when a line is drawn over it.

Finally, and mainly, the automatic detection of the white field line intersections and its adaptation to the homography proved to be too troublesome to solve. The Hough Transform technique was used in order to find a possible solution, however this approach was abandoned since the results obtained from it were nor satisfactory.

As far as analysing the intersections in the image, the group thought about marking them in a noticeable color and sweeping the image horizontally, counting the number of occurrences. If in a certain swept line, only an occurrence was detected, then it would be a field corner. If two were found, then they would correspond to two corners of the great area, and so on.

IV. Conclusion and Results

Overall, the group is satisfied with the results obtained in this project. All aspects that were expected have been accomplished. Although the group did not manage to automatically obtain the points for the homography it believes it still made a great effort in trying to solve this problem.

A. Results

This section illustrates the results of different executions of the developed program

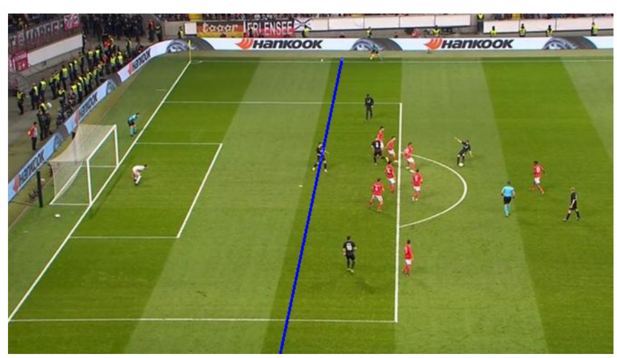

Fig. 4. Offside example 1

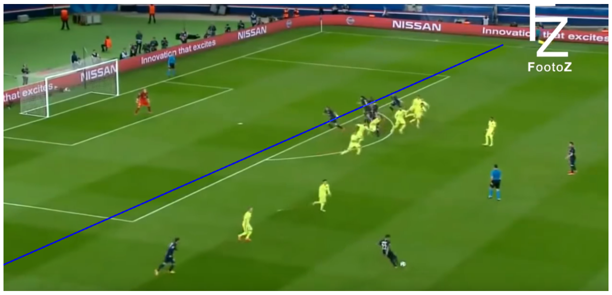

Fig. 5. Offside example 2

Fig. 6. Free Kick example 1

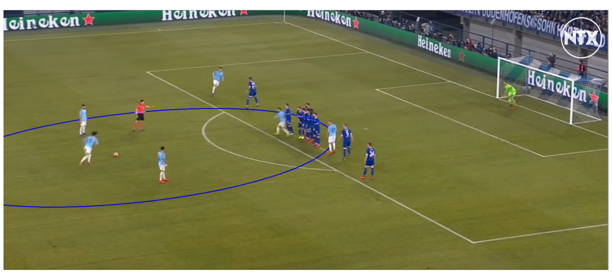

Fig. 7. Free Kick example 2

Fig. 8. Distance to Goal example 1

Fig. 9. Distance to Goal example 2