Automatic Cameraman for Dynamic video acquisition of football match

Bikramjot Singh Hanzra
Dept. of E&EC
PEC University of Technology
160012 Chandigarh, India
Email: bikz.05@gmail.com

Romain Rossi
IRSEEM
ESIGELEC
76801 St Etienne du Rouvray, France
Email: romain.rossi@esigelec.fr

Abstract— **A system is described for dynamic video acquisition of football match. The system uses 2 cameras, a static and a dynamic camera. The raw frames from the static camera are processed to track player position in the field. The tracking data is then used to control a Pan Tilt Zoom (PTZ) Camera that focuses on the area of maximum player density. Other than this the players are also classified into their respective teams, 2D representation of player with respect to field, offside line detection is also done using the static camera frames. Use of multiple static cameras is also discussed in the paper.**

Keywords—tracking; PTZ; sports; broadcasting;

I. INTRODUCTION

Recent developments in the field of computer vision have led to a more scientific approach to use of technology in sports. In this paper, an automatic cameraman for video acquisition is described. Previous work on the topic of dynamic tracking ([1],[2],[3]and[4]) includes tracking a single target and focussing the camera to the middle of the target. [5] proposed an automatic lecture recording system using a PTZ Camera. [6] used a dynamic-static camera pair to generate the trajectory of tennis players.

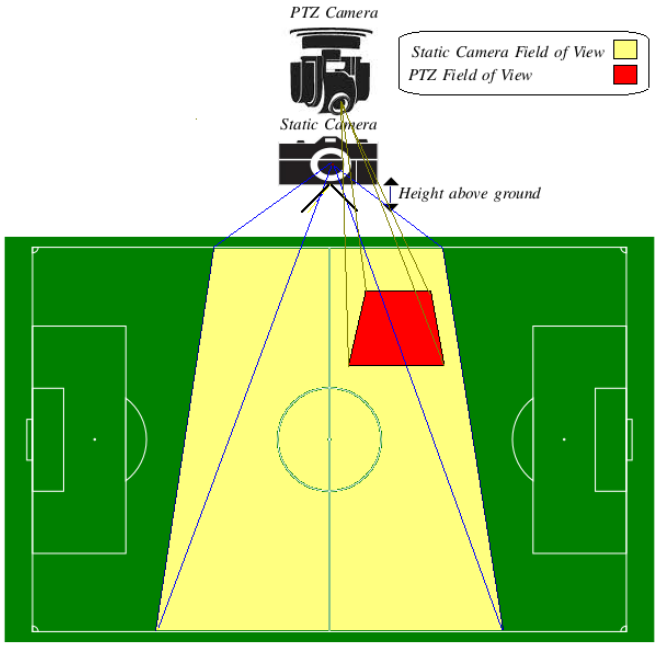


Figure 1: Position of static camera in the dataset

[7] used a system of 8 static cameras to track the positions of football players during a match using single camera processing followed by merging of data from all static cameras to estimate the player positions. Similarly [8], also uses multiple static cameras and then merges the tracking data from the individual cameras using homography to track the players. [9] showed the trajectories of players using a predictive filter. All these works discuss about the tracking of players using static cameras but do not use the data obtained from the static camera(s) to dynamically track the players using dynamic camera(s).

We propose a model in which we use 2 cameras, a static (fixed) camera and a dynamic (PTZ) camera. We track the players using the static camera and then use the dynamic camera to automatically capture the video of a specific region on the field using data produced by the static camera. Apart from being used in football, our system can also be used for video surveillance applications.

II. PROPOSED METHOD

The system includes a static camera, a dynamic camera and a computer for processing. The static camera is placed at a bird-eye view outside the field (as shown in Figure 1) and the dynamic camera is also placed along with the static camera. When the static camera starts video acquisition, the algorithm processes the frame for detection and tracking of players. The tracking data is then used to control a PTZ Camera which records a specific region based on the player tracking information.

The PTZ control is based on the association of 2D areas designated by the user on the static camera image to a specific PTZ position also set by the user. The region which has the most number of players is selected and the corresponding PTZ position is sent to the PTZ camera, which focuses onto that particular area.

III. VIDEO STREAM PROCESSING

The video processing method contains multiple steps shown in Figure 2 and detailed in this part.

978-1-4673-6101-9/13/$31.00 ©2013 IEEE

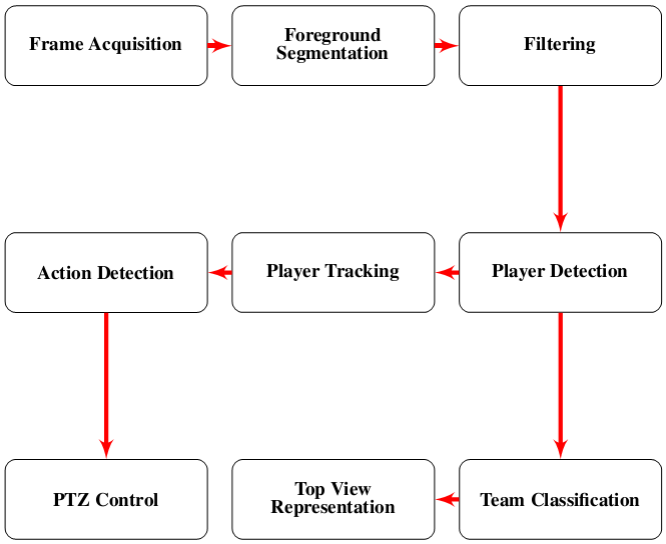


Figure 2: Flowchart of video stream processing

A. *Frame Acquisition*

The first step of the algorithm is capturing of frames from the static camera. Figure 3 shows a sample input frame.

Figure 3: Frame Acquisition: Sample Video frame (Static camera).

B. *Foreground Segmentation*

The first step of the algorithm is separation of foreground objects from the background. An adaptive Gaussian based background model is used as discussed in [10]. Each pixel in the frame is modeled by a mixture of K Gaussian distributions. We define the pixel process of a pixel X as the history of its value from frame 1 to frame t. This is represented by Equation 1.

$$\{X_1, X_2, \ldots X_3\} = \{I(x_0, y_0, i): 1 \leq i \leq t\} \quad (1)$$

where the probability of observing a X at frame t is given by Equation 2.

$$P(X_t) = \sum_{i=1}^{K} wt_{i,t} * \eta(X_t, \mu_{i,t}, \sum_{i,t}) \quad (2)$$

where –

K is the number of Gaussians.

$wt_{i,t}$ is an estimate of the weight.

$\mu_{i,t}$ is the covariance matrix of the i^{th} Gaussian in the mixture a time t.

and η is the Gaussian Probability Density function.

If $X_t \leq 2.5$ standard deviations from the mean, then the pixel is labeled as matched and is part of the background. Weight of matched Gaussians is increased and unmatched Gaussians is decreased. If all the Gaussians in the mixture for pixel X_t are unmatched, the pixel is marked as foreground pixel and the least probable Gaussian is replaced by a new Gaussian.

The algorithm learns of the background in the first few frames of the video and there is no need to prepare the background mask before the match. Figure 4 shows binary mask after background subtraction.

Figure 4: Foreground Segmentation: Foreground binary image after background subtraction.

C. *Filtering*

The foreground image contains small ball-like objects and unconnected blobs. Filtering is done using morphological operations to remove noise from the foreground image.

An eight point square neighborhood is used to identify an individual player as one region. Then, the binary image is successively closed and opened to obtain an image showing only the main objects in the scene. The closing and opening of the binary image is done by equations 3 and 4.

$$A \bullet B = (A \oplus B) \ominus B \quad (3)$$

$$A \circ B = (A \ominus B) \oplus B \quad (4)$$

where A is the Binary image and B is the Structural Element[11].

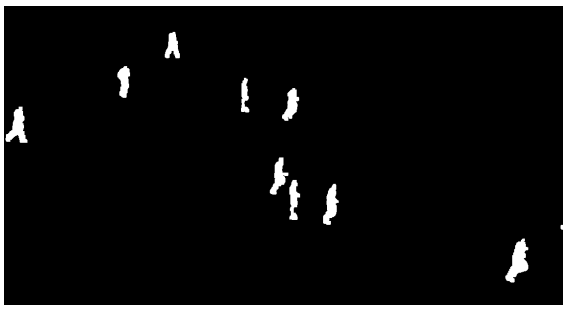


Figure 5: Filtering: Foreground binary image after filtering.

D. *Player Detection*

After filtering, we obtain a binary image that contains

the objects of interest i.e. players. On the binary image, Canny edge detector [12] is used to find the edge pixels that separate different segments in an image. But it fails to give any information from the edges as an object. Next step is to collect edges pixels into contours. We apply a contour extraction algorithm based on [13] to the binary image using an OpenCV [14] implementation. This generates a collection of external contours. Contours are represented in OpenCV by sequences in which every entry in the sequence encodes information about the location of the next point on the curve. Figure 6 represents the segmented player contours. Each individual player is represented by a contour, whereby the edge pixel co-ordinates of each player are used to represent the shape of the player.

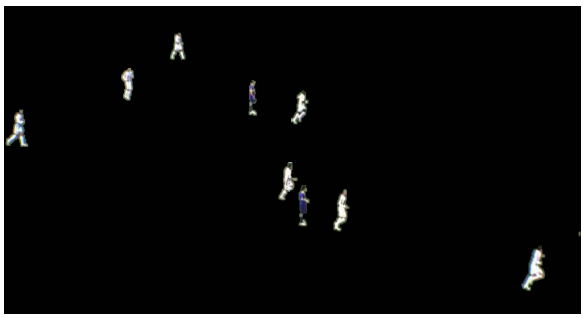


Figure 6: Player Detection: Players are represented as contours in the image

Figure 7 shows the contour of an individual player.

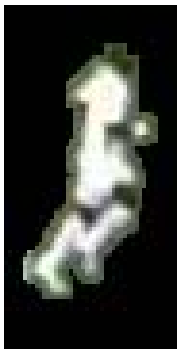


Figure 7: Contour of a single player

E. Player Tracking

The contours obtained above are further processed to get the centroid and bounding box for each player. These 2 parameters are used to track the players throughout the game-play.

The aim of player tracking is to be able to track player positions during occlusion correctly. We calculate the area of each player contour and the mean of all the areas for a frame and if –

$$Area_{Player_i} > Th * Area_{Mean} \quad (5)$$

where –

$Area_{Player_i}$ is the area of the i^{th} player.

Th is the threshold.

and $Area_{Mean}$ is the mean of all player areas.

is TRUE, then we consider the area to have multiple players in the same contour. Figure 8 shows the trajectories of tracked players.

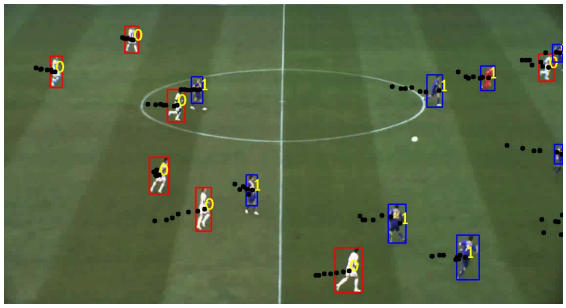


Figure 8: Player Tracking: Player trajectories are represented in the image.

F. Action Detection and PTZ Control

The first step of action tracking and PTZ control is marking of regions on the static camera and dynamic camera. Regions are linearly mapped from static camera to dynamic camera. The user interactively sets regions on the static camera image to particular values of Pan, Tilt and Zoom of the PTZ camera. These regions could be defined before running the algorithm or changed while the algorithm is running.

The second step after marking the regions is to move the PTZ camera to the region selected by the algorithm. The algorithm calculates the number of players in each region using the contour centroid values and selects the regions that has most number of player. The PTZ camera then focuses onto that region by changing to the Pan, tilt and zoom value corresponding to that particular region.

G. Team Classification

There will be a maximum of 5 different kits (uniforms i.e. jersey plus shorts) on the pitch, 2 team kits, 2 goalkeeper kits and the referees kit. [15] used a model to learn the 5 kit colors before the game. For each player iteratively pixel matching is done with the 5 kit colors and player is assigned to the kit with the highest number of votes. [7] used histograms to represent the five colors.

Figure 9: Player Classification: Red and Blue Bounding Boxes represent the 2 teams

In our implementation, we calculate the 3 point vector mean of player area pixel values and then cluster these values

with the help of K-Means clustering [16]. Figure 9 shows the result of our approach where the players of 2 teams are represented by different color bounding boxes.

H. Top View Representation

The objective is to find the external parameter i.e. position co-ordinates relatively to a world co-ordinate system. One of the most used techniques is the one proposed by Tsai Algorithm [17]. Its implementation needs corresponding 3D point co-ordinates and 2D pixels in the image. It uses a two-stage technique to first the external parameters and then the internal parameters. [18] and [19] proposed methods just needs to recover the parameters relative to the position and orientation of the camera.

Since in our application, we only want transformation from one 2D plane to another, we use a perspective transformation [20]. This approach only needs four point (set by the user) in the image plane for the transformation.

$$\begin{bmatrix} XW \\ YW \\ W \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (6)$$

where-

XW and YW are the world co-ordinates.
x and y are image co-ordinates.
a, b, c, d, e, f, g, h and i represent the transfer matrix parameters.
and $W = gx + hy + 1$

The transfer matrix is calculated as under –

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -X_1x_1 & -X_1y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -Y_1x_1 & -Y_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -X_2x_2 & -X_2y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -Y_2x_2 & -Y_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -X_3x_3 & -X_3y_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -Y_3x_3 & -Y_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -X_4x_4 & -X_4y_4 \\ 0 & 0 & 0 & x_4 & y_4 & 0 & -Y_4x_4 & -Y_4y_4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ X_3 \\ Y_3 \\ X_4 \\ Y_4 \end{bmatrix} \quad (7)$$

where-

X_k and Y_k $(1 \leq k \leq 4)$ are the 4 world co-ordinates.
x_k and $y_k (1 \leq k \leq 4)$ are the 4 image co-ordinates.

Figure 10 shows the result of perspective transformation and the 2D top view.

(a) Transformed Image (b) 2D Top View

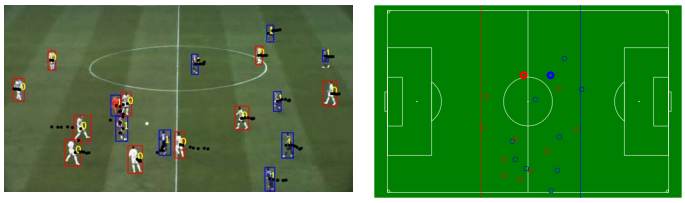
Figure 10: Transformed Image and 2D top view

I. Offside Line

Offside is a defensive tactic that is used by the defensive team to prevent the attacking team from scoring.

In our implementation, we draw 2 offside lines, one for each team. This is done by:

- First, calculating the mean x co-ordinate of both the teams, minimum and maximum player x co-ordinate of both the teams.
- Based on the value of means, we select one line that is the minimum player x co-ordinate of one team and the maximum player co-ordinate of the other team and vice-versa.



(a) Field View (b) Offside View

Figure 11: Offside Lines

J. Image Stitching

Image Stitching is the process of stitching or combining images taken from different reference points to produce a new image called panoramic image. Since a single camera is not enough to cover the whole field we use multiple cameras to capture the image of the entire field. An OpenCV implementation has been used.

The image stitching process is divided into 3 main components:

1) Image Registration

The first step is extracting features from image and then matching images based on these features. Image rotations are also calculated. For the following image:

Features in Image 12a: 295
Features in Image 12b: 564
Features in Image 12c: 619
Features in Image 12d: 335

2) Calibration

In this camera matrix and distortion matrix are calculated and wave correction is done.

Initial intrinsic parameters 12a:
= [570.75, 0, 320; 0, 570.75, 240; 0, 0, 1]
Initial intrinsic parameters 12b:
= [570.75, 0, 320; 0, 570.75, 240; 0, 0, 1]
Initial intrinsic parameters 12c:
= [570.75, 0, 320; 0, 570.75, 240; 0, 0, 1]
Initial intrinsic parameters 12d:
= [570.75, 0, 320; 0, 570.75, 240; 0, 0, 1]
Camera 12a:
= [697.18, 0, 320; 0, 697.18, 240; 0, 0, 1]
Camera 12b:
= [684.19, 0, 320; 0, 684.19, 240; 0, 0, 1]
Camera 12c:
= [666.87, 0, 320; 0, 666.87, 240; 0, 0, 1]
Camera 12d:
= [648.90, 0, 320; 0, 648.90, 240; 0, 0, 1]

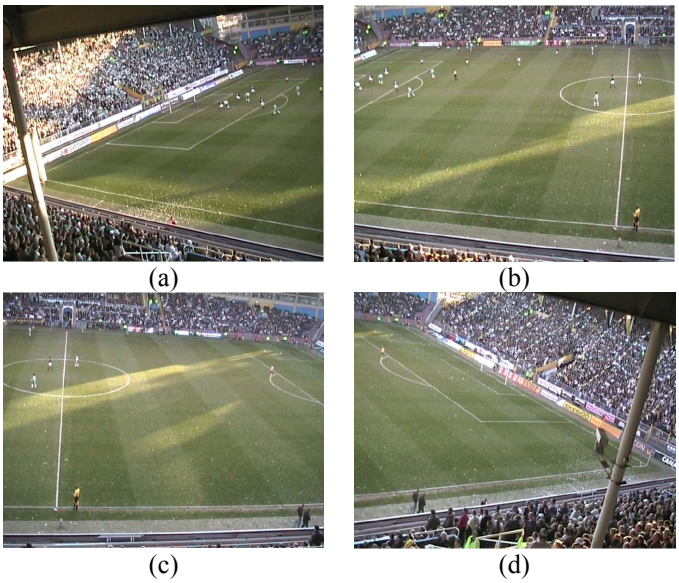


Figure 12: Images from 4 different reference frames

3) Blending

In this wave correction, warping and exposure compensation is done. Figure 13 shows the stitched image obtained from 12a, 12b, 12c and 12d.

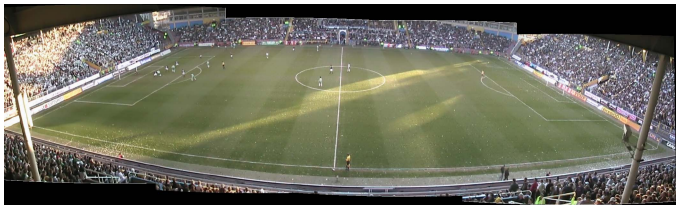


Figure 13: Stitched Image

IV. Experimentation and Results

To evaluate the performance of the proposed video tracking algorithm, experiments were carried out on video sequences from the CNR soccer dataset [21].The video sequences were of duration 1 minute and 59 seconds shot at a frame rate of 25 frames per second.

For foreground segmentation, we represent each pixel value by a pair of 3 Gaussians and the last 100 frames are used as 'Pixel Process'. 3 Gaussians are adequate to represent a pixel as there are not many variations on the field and the background is indeed uni-modular. The Background Subtraction algorithm is robust to

- Gradual illumination changes and changing weather conditions.
- Slow moving player or even stationary players for a few frames.
- Figure 14 shows the results of the background subtraction algorithm. Here, the player and the ball are clearly separable and distinct.

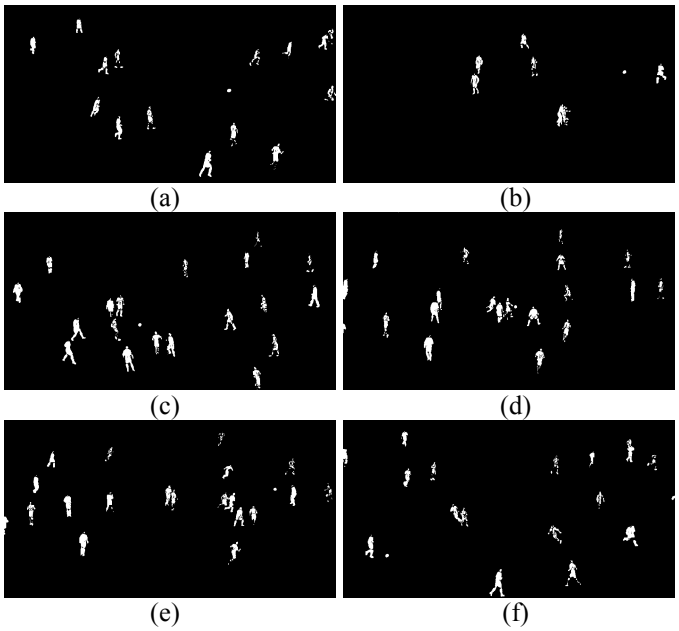
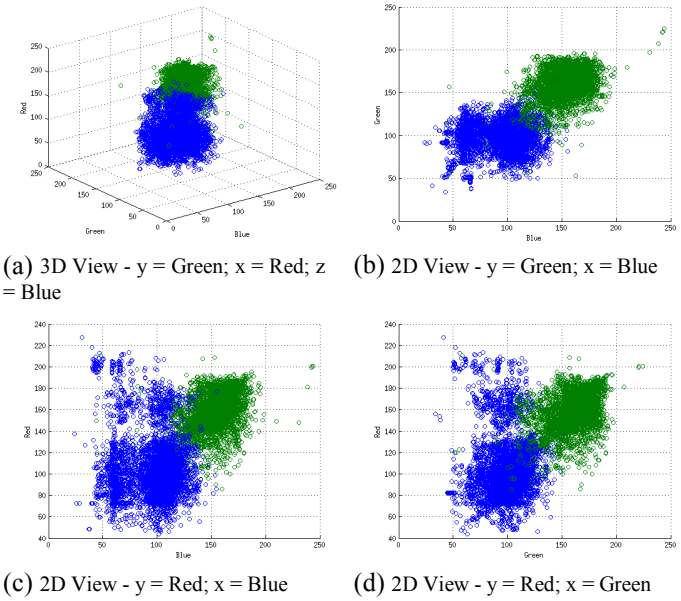


Figure 14: Background Subtraction: The figures show results of the background subtraction algorithm (The ball and the players are clearly visible)

Figure 15 shows the result of player classification into 2 clusters. Each data point represents a 3 point vector mean of player contour pixel values. The 2 clusters can be distinguished easily and the classification has a very high accuracy. Once the background mask is prepared in the first few frames, there is no false detection.



(a) 3D View - y = Green; x = Red; z = Blue (b) 2D View - y = Green; x = Blue

(c) 2D View - y = Red; x = Blue (d) 2D View - y = Red; x = Green

Figure 15: shows the result of player classification into 2 clusters.

Figure 16 shows the result of region marked in black as possible occlusion regions.

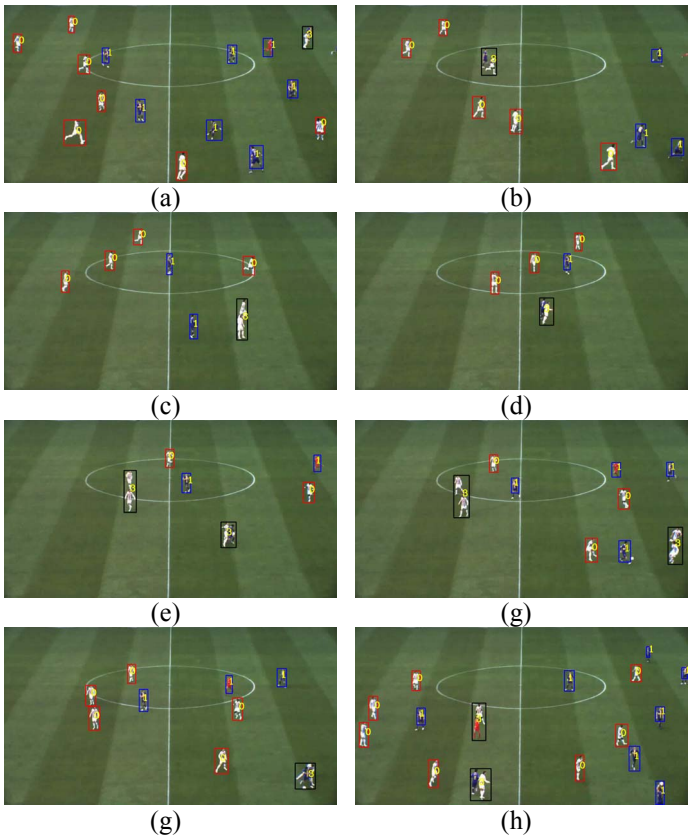


Figure 16: Occlusions: The figures show the detected occlusions (Black rectangular boxes)

V. CONCLUSION

We proposed a method and setup to automatically track action in a football match using a static camera for real-time player tracking and a PTZ camera for video acquisition. The PTZ control is based on user-defined areas and associated PTZ camera position so it can be easily adapted to other sports.

The player tracking method is based on an adaptive Gaussian-based approach for foreground segmentation, followed by filtering and contour detection. Our algorithm also performs occlusion detection and player classification into teams.

Our approach allows fast and easy system setup since it doesn't need calibration. Moreover its real-time performances make it suitable for broadcasting applications. In future developments, other useful information (player number and name, achieved goals, ball position and trajectory) can be extracted using either the static or dynamic camera images.

VI. REFRERENCES

[1] D. C. Woo and D. W. Capson, "3d visual tracking using a network of low-cost pan/tilt cameras," in Electrical and Computer Engineering, 2000 Canadian Conference on, vol. 2, pp. 884–889, IEEE, 2000.

[2] R. Canals, A. Roussel, J.-L. Famechon, and S. Treuillet, "A biprocessor- oriented vision-based target tracking system," Industrial Electronics, IEEE Transactions on, vol. 49, no. 2, pp. 500–506, 2002.

[3] L. Jordao, M. Perrone, J. P. Costeira, and J. Santos-Victor, "Active face and feature tracking," in Image Analysis and Processing, 1999. Proceedings. International Conference on, pp. 572–576, IEEE, 1999.

[4] S. Reddi and G. Loizou, "Analysis of camera behavior during tracking," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 17, no. 8, pp. 765–778, 1995.

[5] H.-P. Chou, J.-M. Wang, C.-S. Fuh, S.-C. Lin, and S.-W. Chen, "Automated lecture recording system," in System Science and Engineering (ICSSE), 2010 International Conference on, pp. 167–172, IEEE, 2010.

[6] J. G. Girldez, J. L. Lamas, Y. L. D. Meneses, and J. Jacot, "Automatic cameraman with trajectory extraction,"

[7] M. Xu, J. Orwell, and G. Jones, "Tracking football players with multiple cameras," in Image Processing, 2004. ICIP'04. 2004 International Conference on, vol. 5, pp. 2909–2912, IEEE, 2004.

[8] Iwase, Sachiko, and Hideo Saito. "Parallel tracking of all soccer players by integrating detected positions in multiple view images." Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on. Vol. 4. IEEE, 2004.

[9] Morais, Erikson, et al. "Automatic tracking of indoor soccer players using videos from multiple cameras." Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on. IEEE, 2012.

[10] .C. Stauffer and W. E. L. Grimson, "Adaptive background mixture mod- els for real-time tracking," in Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on., vol. 2, IEEE, 1999.

[11] R. C. Gonzalez and R. Woods, Digital Signal Processing. Pearson Publications, 2008.

[12] J. Canny, "A computational approach to edge detection," Pattern Analy- sis and Machine Intelligence, IEEE Transactions on, no. 6, pp. 679–698, 1986.

[13] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following.," CVGIP 30 1, pp 32-46, 1985.

[14] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.

[15] W. C. Naidoo and J. R. Tapamo, "Soccer video analysis by ball, player and referee tracking," in Proceedings of the 2006 annual research conference of the South African institute of computer scientists and in- formation technologists on IT research in developing countries, pp. 51–60, South African Institute for Computer Scientists and Information Technologists, 2006.

[16] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., Vol. 1 (Univ. of Calif. Press, 1967), 281-297.

[17] R. Y. Tsai, "An efficient and accurate camera calibration technique for 3d machine vision," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 1986, 1986.

[18] Z. Zhang, "A flexible new technique for camera calibration," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 22, no. 11, pp. 1330–1334, 2000.

[19] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in Computer Vision and Pattern Recog- nition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, pp. 1106–1112, IEEE, 1997.

[20] A. Criminisi, I. Reid, and A. Zisserman, "A plane measuring device," Image and Vision Computing, vol. 17, no. 8, pp. 625–634, 1999.

[21] T. Dorazio, M. Leo, N. Mosca, P. Spagnolo, and P. Mazzeo, "A semi-automatic system for ground truth generation of soccer video se- quences," (Genoa, Italy), In the Proceeding of the 6th IEEE International Conference on Advanced Video and Signal Surveillance, September 2-4 2009.