

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Video Analysis in Indoor Soccer with a Quadcopter

Filipe Trocado Ferreira

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Jaime dos Santos Cardoso (PhD)

Second Supervisor: Hélder Filipe Pinto de Oliveira (PhD)

July 24, 2014

Resumo

Na indústria multimilionária do futebol profissional, bem como noutros desportos de alta competição, o planeamento estratégico levado por cada equipa é essencial na obtenção de resultados desportivos e do respetivo retorno financeiro. Os sistemas automáticos de visão para análise de eventos desportivos são, atualmente, ferramentas indispensáveis na preparação técnica e táticas nas equipas de elite dos diversos desportos coletivos. A utilização de múltiplas câmaras e a necessidade de operadores humanos facilitam o processamento de imagem necessário mas inflacionam o custo destes sistemas tornando-os inacessíveis à grande maioria das equipas de nível médio.

Nos últimos anos, os mais recentes desenvolvimentos em diversas tecnologias, tais como: sistema de sensorização, estabilidade, controlo, comunicações, armazenamento energético ou dos materiais, tornaram os veículos aéreos não tripulados, nomeadamente os Quadcopters, cada vez mais acessíveis para um vasto número de aplicações.

Neste trabalho é apresentado um sistema de visão automático para a extração de informação relevante de jogos de Futsal a partir de imagens capturadas por um AR. Drone 2.0. A utilização de um drone para adquirir imagens apresenta-se como uma solução de baixo custo, portátil e com uma grande flexibilidade de utilização. Contudo, este tipo de veículos sofre muitas vibrações e perturbações tornando a imagem pouco estática o que irá resultar num conjunto de problemas pouco usuais neste tipo de sistemas de visão. A estabilização da imagem é conseguida a partir de correspondência de características entre frames consecutivas. Para mapear a posição dos jogadores no campo é utilizado um método de calibração baseado na deteção e correspondência das linhas do campo com a de um modelo virtual. A posição dos jogadores na imagem é conseguida a partir da deteção a partir de descritores HOG e do tracking a curto prazo com o algoritmo Mean Shift.

A partir dos dados recolhidos é realizada uma análise de alto nível a algumas vertentes do jogo nomeadamente: mapas de ocupação, atitude das equipas e formação tática. Este trabalho apresentou resultados positivos quer quantitativamente quer qualitativamente contudo quase todas as tarefas apresentadas têm espaço para melhoramentos.

Abstract

In the billionaire industry of football as in many other sports the strategic planning taken by each team is essential to achieve the desired results and the respective financial return. Automatic vision systems for analysis of sports events are currently indispensable tools in technical and tactical preparation in the elite teams of the various collective sports. The use of multiple cameras and the need for human operators facilitate image processing but inflate the cost of these systems making them inaccessible to the majority of mid-level teams.

In recent years, the latest developments in various technologies, such as sensing system, stability control, communications, energy storage or materials, made unmanned aerial vehicles, in particular Quadcopters increasingly accessible to a large number applications.

In this work an automatic vision system for extraction of relevant information on indoor soccer games from images captured by an AR. Drone 2.0 is presented. The use of a drone to record images from the game is presented as a low-cost, portable and flexible solution. However, this kind of vehicles is subject to many vibrations and disturbances making image not static which will result in a set of unusual problems in this type of vision systems.

Video stabilization is achieved using features matching between two adjacent frames. To map the position of the players on the field is used, a camera calibration method based on the detection and matching of the field lines on a virtual model. The position of the players in the image is obtained with HOG detector and short term tracking with Mean Shift algorithm.

From the collected data an analysis of high-level aspects of the game, namely: occupation maps, team attitude and defensive tactic formation. This study showed positive results both quantitatively and qualitatively yet almost all the tasks presented have room for improvement.

Agradecimentos

Este trabalho é o culminar de cinco anos de muito trabalho, aprendizagem mas também de grande vivências e amizades. Tal só foi possível pelas condições que a minha família me proporcionou durante toda a minha vida.

Quero agradecer a todos os meus familiares, com especial atenção aos meus pais e avós, pela inspiração, incentivo e carinho que sempre me deram.

Aos meus orientadores neste trabalho: ao Professor Jaime Cardoso e ao Hélder Oliveira pelos conselhos, criatividade e rigor que transmitiram ao longo de todo o projeto.

A todas as pessoas do INESC-Porto por bem me receberem e, principalmente, a todos os que jogaram no XIV Torneio de Futebol do INESC e que se voluntariaram como "cobaías" para aparecer nas sequências gravadas.

A todos os professores com quem tive o prazer de me cruzar e que muito contribuíram para a minha aprendizagem e crescimento como futuro Engenheiro.

À Dianita por ser uma fonte de inspiração e distração permanente.

E por fim, mas não com menos importância, a todos os amigos com quem tive prazer de partilhar esta caminhada aprendendo e desaprendendo muito com especial carinho pelo ano de 09, uma das mais requintadas colheitas de Eletrotecnia!

Filipe Trocado Ferreira

'Chaos is merely order waiting to be deciphered'

José Saramago, *The Double*

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1 | Context | 1 |
| 1.2 | Objectives | 2 |
| 1.3 | Contributions | 2 |
| 1.4 | Structure | 3 |
| 2 | State of the Art | 5 |
| 2.1 | Computer Vision in Sports Analysis | 5 |
| 2.2 | Related Work | 7 |
| 2.2.1 | Summary | 9 |
| 2.3 | Commercial Solutions | 10 |
| 2.3.1 | Prozone | 10 |
| 2.3.2 | Amisco | 11 |
| 2.3.3 | STATS | 11 |
| 2.3.4 | OptaPro | 12 |
| 2.3.5 | ChyronHego | 12 |
| 2.3.6 | Summary | 12 |
| 2.4 | Quadcopters | 13 |
| 2.4.1 | Dynamics and Control | 13 |
| 2.4.2 | Commercial Solutions | 13 |
| 2.5 | Conclusions | 15 |
| 3 | System Overview | 17 |
| 3.1 | Image Acquisition | 17 |
| 3.2 | Image Processing | 18 |
| 3.2.1 | System Framework | 18 |
| 3.3 | Methods Evaluation | 19 |
| 3.3.1 | Quantitative Evaluation | 20 |
| 3.3.2 | Qualitative Evaluation | 21 |
| 3.3.3 | Test Sequences | 21 |
| 3.4 | Conclusions | 22 |
| 4 | Video Stabilization and Camera Calibration | 25 |
| 4.1 | Video Stabilization | 25 |
| 4.1.1 | Motion Estimation | 26 |
| 4.1.2 | Motion Compensation | 29 |
| 4.1.3 | Image Composition | 29 |
| 4.2 | Results of Video Stabilization | 29 |

| | | |
|----------|---|-----------|
| 4.3 | Camera Calibration | 33 |
| 4.3.1 | Lens Distortion | 33 |
| 4.3.2 | Solving Field-to-Image Homography | 34 |
| 4.4 | Results of Camera Calibration | 36 |
| 4.4.1 | Initialization | 36 |
| 4.4.2 | Line Detection and Matching | 39 |
| 4.5 | Conclusions | 44 |
| 5 | Player Detection | 47 |
| 5.1 | Overview of Detection Methods | 47 |
| 5.1.1 | Basic Segmentation Methods | 47 |
| 5.1.2 | Sliding Window Detection Methods | 48 |
| 5.1.3 | Mean Shift and Camshift | 49 |
| 5.2 | Proposed Method | 50 |
| 5.2.1 | HOG Detector Implementation | 51 |
| 5.2.2 | Team Identification and False Positive Handling | 53 |
| 5.2.3 | Mean Shift and Short term tracking | 54 |
| 5.2.4 | Assigning detections to tracks | 56 |
| 5.2.5 | Results | 58 |
| 5.2.6 | Failure Situations | 59 |
| 5.3 | Conclusion | 60 |
| 6 | High Level Interpretation | 63 |
| 6.1 | Occupation Map | 63 |
| 6.2 | Team Attitude | 64 |
| 6.3 | Team Tactics | 66 |
| 6.4 | Conclusions | 68 |
| 7 | Conclusions and Future Work | 69 |
| 7.1 | Future Work | 70 |
| | References | 73 |

List of Figures

| | | |
|------|--|----|
| 3.1 | Representation of Quadcopter's position while recording the image sequences from indoor soccer games | 18 |
| 3.2 | Schematic of the system framework | 19 |
| 3.3 | Frame 1,100 and 500 of original sequence number 1. | 21 |
| 3.4 | Frame 1,100 and 500 of original sequence number 2. | 22 |
| 3.5 | Frame 1,100 and 500 of original sequence number 3. | 22 |
| 3.6 | Frame 1,100 and 500 of original sequence number 4. | 22 |
| 4.1 | Result of FAST corner detection in Frame 1 and Frame 2 of video sequence nr.4. | 30 |
| 4.2 | Result of Features Matching using RANSAC | 30 |
| 4.3 | Comparative analysis of the stabilization method. On a) the mean of the first 30 frames of the sequence nr.1. b) The stabilized version of the same sequence | 31 |
| 4.4 | Mean of stabilized sequence nr. 1 for the first 8 seconds. | 32 |
| 4.5 | Response of video stabilization to a strong oscillation | 32 |
| 4.6 | Transformation of line and points from image to Hough space. Points in image corresponds to sinusoids in Hough Space and Points in Hough Space to lines in image [1] | 36 |
| 4.7 | Example of the 4 pairs of corresponding points to calibration initialization | 37 |
| 4.8 | Model of the field projected on the first frame using the initial homography | 38 |
| 4.9 | Example of virtual model created manually | 39 |
| 4.10 | Example the pre-processing method to obtain a binary image to be used in line detection | 40 |
| 4.11 | Two examples of line detection on different frames. | 40 |
| 4.12 | Example of line matching using an assignment problem. Horizontal lines detected (right) are matched to the lines on the virtual model(left). Each color represents an assignment | 41 |
| 4.13 | Example of line selection for picking the best intersection points. Note that in left image there is a right line on the right of the selected as the most-right. That happens because that line is not in the model. | 42 |
| 4.14 | Example of the final result of the automatic calibration method. On the left it is possible to observe the non calibrated projection of the field model. On the right the model is correctly projected using the calibration method presented above. | 42 |
| 4.15 | Error of camera calibration method with different correction rates on sequence 4. | 43 |
| 4.16 | Error of camera calibration method with different correction rates on sequence 1. | 43 |
| 4.17 | In sequence 1 almost all the detected lines are not in the expected half field increasing the probability of bad assignment and wrong calibration. | 44 |

| | | |
|-----|---|----|
| 5.1 | Example of bad results from background subtraction using a Gaussian Mixture Model for its representation and corresponding detections. This method produced a low rate of true detections with an high percentage of false positives | 51 |
| 5.2 | Representation of usual HOG detector results. Frame 450 of sequence nr.4 | 52 |
| 5.3 | Example of the histograms extracted from one output of the HOG detector | 53 |
| 5.4 | Example of the results of team identification and false positive handling. Red boxes represents false positives deleted. Yellow boxes for players of Team A and Orange ones for player of Team B. Must notice that a Team B player is wrongly classified as player of Team A. Frame 450 of sequence nr.4 | 55 |
| 5.5 | Representation of mean shift algorithm on player position prediction on two consecutive frames. In this case only the track for one player is represented. On the left image player initial position is represented by a green bounding box. The sub-region of the box containing the shirt is represented on the next image. This region will be used by mean shift algorithm to predict the new location of the player, represented on the third image. Finally player position is predicted applying the same translation of the mean shift algorithm to the initial bounding box. | 56 |
| 5.6 | Example of final output of player detection stage. Player position are represented by its bounding box and the team by the color of the box. | 58 |
| 5.7 | Evolution of Player detection results through the different stages of the method on sequence 3. | 58 |
| 5.8 | Evolution of Player detection results through the different stages of the method on sequence 4. | 59 |
| 5.9 | Examples of usual failure situations. | 60 |
| 6.1 | Occupation map of player on sequence 3. a) and b) relates do information extracted from ground-truth data for each one of the teams, and c) and d) the proposed detection and calibration methods. | 64 |
| 6.2 | Result of the method to extract information on teams attitude and offensive trend. a) Illustrate the result for ground truth data and b) with the proposed detection and calibration methods. At blue and red are highlighted two frames which will be shown on figure 6.3 | 65 |
| 6.3 | Example of two results of player detection on sequence 3:a)is the Frame 200 of sequence 3. b)is the Frame 800 of sequence 3. | 66 |
| 6.4 | Evolution of the tactics counter for the sequence 3. a) Results extracted from ground truth data. b) Results from the proposed method. | 67 |
| 6.5 | Two examples of how player detection results will influence the results of the tactic analysis. If in the first image is possible to see that "1-2-1" is the formation used, in the second image the system cannot detect because one player is not being detected. | 67 |

List of Tables

| | | |
|-----|--|----|
| 2.1 | Comparative Analysis of Commercial Solutions | 12 |
| 2.2 | Ar.Drone 2.0 Technical Specifications | 13 |
| 2.3 | Dji Phantom Technical Specifications | 14 |
| 2.4 | Firefly Technical Specifications | 14 |
| 2.5 | Quadcopters Evaluation . '++' -Quite Positive. '-' -Quite Negative | 15 |
| 4.1 | 2D transformations hierarchy [2] | 28 |

Abbreviations

| | |
|----------|--|
| 2D | Two Dimensions |
| 3D | Three Dimensions |
| AdaBoost | Adaptive Boosting |
| ASPOGAMO | Automated Sport Game Analysis |
| DLT | Direct Linear Transform |
| DoG | Difference of Gaussians |
| DPM | Deformable Part Model |
| FAST | Features from Accelerated Segment Test |
| fps | frames per second |
| GPS | Global Positioning System |
| HD | High Definition |
| HOG | Histogram of Oriented Gradients |
| HSI | Hue-Saturation-Intensity |
| HSV | Hue-Saturation-Value |
| ICP | Iterative Closest Points |
| MS | Mean Shift |
| MSAC | M-estimator SAmple Consensus |
| MSER | Maximally Stable Extremal Regions |
| OpenCV | Open Source Computer Vision Library |
| RANSAC | RANdom SAmple Consensus |
| RGB | Red-Green-Blue |
| ROS | Robotic Operating System |
| SIFT | Scale Invariant Feature Transform |
| SURF | Speeded Up Robust Features |
| SVM | Support Vector Machine |
| TI | Team Identification |
| TV | Television |
| UAV | Unmanned Air Vehicle |
| VSLAM | Visual Simultaneous Localization and Mapping |

Chapter 1

Introduction

In the billionaire industry of football as in many other sports the strategic planning taken by each team is essential to achieve the desired results and the respective financial return. On this planning information regarding the position and trajectories of the players during one or more games is often used. These data once interpreted can be used to set the strategy for a team before or during the game. However, the current systems are not accessible to the big majority of the teams so the study of low cost solutions with high reliability is required.

1.1 Context

Football is without question one of the most popular sports worldwide. All the monetary amounts related to this sport justify the millionaire budget of the teams, not only in the acquisition of players, but also technical staff. They have the responsibility to prepare the team and help the coach in order to achieve the best results. In the strategic planning for the games, information about position and movement of the player on the pitch is used by the coach and other technical staff. Initially, statistics on collective and individual performance were calculated manually with low reliability and precision. Currently, the solutions used to provide this information clearly have high acquisition and license prices because of high complex installation with multiple fixed cameras around the stadium and excessive human intervention in the video analysis.

The unmanned aerial vehicles have been gaining relevance in different areas of use: military, the recreational and also the sport. These vehicles normally equipped with high definition cameras, can be used autonomously to obtain images in a stadium or other sports hall. A UAV can be a reliable, portable and low cost solution to capture images from soccer matches. The images can be processed in order to extract useful information about individual and collective performance. However, the unique features and particularities of this system architecture for image recording bring about difficulties that have not been considered in some of the previous works, such as the camera motion, camera calibration and the need of complex methods for players' detection. This project intends to develop a system that overcomes these problems to extract and map the player positions and trajectories from the image to the world coordinates.

1.2 Objectives

This project aims to study a video analysis framework for a low cost image acquisition system of indoor soccer games using a Quadcopter. The main goal of this project is to study and implement an automatic video analysis framework in order to get complex information about the game from image sequences shot by an AR. Drone 2.0. It intended to automatically extract the position of the player from the image and map it in the world coordinates. From these low-level data high-level information can be extracted such as occupational heatmaps, offensive and defensive trends, tactics interpretation, among others.

An initial framework will be set up using common methods found in the literature. The main stages of this frame work will be:

- Video Stabilization using Features Matching
- Camera Calibration using Hough Transform and homography estimation
- Player Detection with HOG detector
- Short term Player tracking using mean shift algorithm

This framework will include an image stabilization module to handle the movement and instability of the unmanned air vehicle. Camera Calibration module is needed to map the players on the world coordinates dynamically. Different player detection methods will be studied and optimized to achieve the best results. After getting reliable low-level data on players' positions and team identification, complex information about team performance and the game will be obtained and shown to the user. The different methods will be evaluated qualitatively and also quantitatively comparing results with hand-annotated data.

This project aims to be a starting point from an automatic vision system to collecting information on sports events with an image acquisition system based on unmanned air vehicles. The main problems common to these systems will be identified and some methods will be experimented to solve them. In the end a complete framework based on common methods will be available setting a base for future works to improve the results.

1.3 Contributions

From the work developed on the scope of this dissertation resulted various contributions as:

- Development of computer vision system framework for indoor soccer analysis based on image capturing with a quadcopter.
- Simple and robust camera calibration method based on the detection and matching of the several different lines of indoor sports venues.
- Study and implementation of player detection and short tracking method based on the fusion of HOG detector with mean shift tracking.

- Sequences from indoor soccer games recorded with an Ar.Drone 2.0 and corresponding ground truth annotation data.

1.4 Structure

After this introductory chapter, follows a review of the literature and an analysis of the solutions found in the market about the subject of computer vision systems in sports analyses in chapter 2. In chapter 3 is presented an overview of the framework of the system and identified the major problems that it will face. Chapter 4 describes the first two stages of the framework related to video stabilization and camera calibration. Preliminary results are also presented at this stage. In chapter 5 players' detection methodology is presented and discussed. Quantitative results of the different stages of the method are also presented in this chapter. In chapter 6 the results of High Level interpretation are presented and discussed. Finally, in the last chapter the work developed is discussed and the main conclusions are presented.

Chapter 2

State of the Art

In this chapter some of the works found in literature about computer vision systems for extraction of sports data as well as some of the commercial solutions in the market will be analysed and discussed. The research developed on quadcopters will be shown and the choice used in this project substantiated.

2.1 Computer Vision in Sports Analysis

In the last few decades player tracking and automatic performance evaluation during soccer games as in many other sports became an interesting subject of study. The resulting data is important not only for team strategic planning but also for broadcasting information. Many of the algorithms and methods developed for these systems can be applied in many different areas of interest as: surveillance, healthcare, among others.

It is possible to divide automatic player tracking systems in two categories [3] : intrusive and not-intrusive. Currently, intrusive systems rely on Global Positioning System (GPS) or other wireless signal-based position detection and can achieve highly accurate data. However, these systems require a hard and expensive implementation. Besides they are not allowed during matches in most of the official tournaments. Non-invasive systems normally use automatic or semi-automatic video analysis. From image sequences collected from one or more camera it is possible to extract players positions and trajectories. Player's tracking is very challenging due to the competitive nature of the game: occlusion, overcrowded scenes players disappearing image plane or camera motion are undesired but usual situations found in literature that still maintain inaccurate results.

A computer vision system can be modelled by the different stages of the process that differ from one another in terms of input, function and output. The system model can be divided in the following main stages: Image Acquisition, Pre-Processing, Image Segmentation, Object Recognition, Tracking.

Image Acquisition

Image Acquisition is, as expected, the first stage of a computer vision system but also a critical one. The methods used to collect the images from the game will strongly influence the following stages of the system. Image acquisition architectures normally differ on the number of cameras and in how they are located on the sports venue. Multiple fixed cameras allow to cover all the field and ease the segmentation methods to apply afterwards [4, 5, 6, 7]. Simpler image acquisition architectures such as with a single camera [8, 9] or using TV broadcasting sequences [10, 11, 12] will require more complex processing on the following stages, mainly on player detection and camera calibration.

Pre-Processing

Pre-Processing is a required step after collecting the images. It is used to filter, correct and enhance some aspects of the image. On sports analysis the main functions of pre-processing stage are associated with the relation between the camera and the world.

Video Stabilization is often performed to compensate undesired camera motion which is important when spatial image coherence is required.

Camera Calibration is an indispensable step of sports analysis since it is necessary to find the relation between image coordinates and world coordinates. For instance, finding the camera parameters it is possible to relate the position of the player in the image with their actual position on the field [4, 13]. When fixed cameras are used, this stage is trivial and can be performed manually, otherwise, when image camera moves, dynamic automatic methods are required.

Image Segmentation

Image segmentation is the process to divide the image in different regions with common features. In sports analysis these regions are usually associated to players, ball or other interesting regions as field lines while simultaneously other undesired regions have to ignore: spectators, strange objects, bad illumination among others.

The most usual techniques are based on background subtraction since using a background model created from initial frames [7] to more complex dynamic model using its representation on a specific colorspace taking advantage of a dominant and homogeneous color field [4, 5, 10].

However, when neither background is static nor there is a dominant field color as in indoor sports, the basic methods presented above are not suitable for players' segmentation.

Object Detection

More complex methods can be used to perform player detection. Some of these methods lie on the extraction of features and posterior classification [8, 14].

Features can be extracted from points or regions on the image considering their distinctiveness and invariance to image transformation and illumination changes. These can be related to pixel intensities or other feature space as shape, orientation, textures among others.

Classifiers compare the features extracted from the image with those used to train it. A classifier can be trained from a set of positive and negative samples and corresponding extracted features. Different methods can then use the training data to categorize a features vector into one of the trained classes and updating them.

Sliding window detectors extracts features from the window region on the image and categorize it using the classifier previously trained as detection and non detection.

These group of methods are widely used on sport analysis when basic segmentation methods are not suitable.

Tracking

From the position on the image of the detection of the player is possible to extract their trajectories, however the detections along the frames contains noise and are not always available. Tracking an object through image sequences can be performed using different approaches. Methods based on the apparent motion of objects on the image (optical flow [15]). Similarity measurement can be used to predict the position of a object on following frames [16]. Using dynamic models to describe players movement and corresponding measurements, it is possible to predict the player's position and smooth measured data. Assuming linear model and Gaussian error, Kalman Filter [17] is a very popular solution to object tracking. Particle Filter [18] can assume non linear dynamic model and non Gaussian error approximating it to reality but also expanding computational requirements.

2.2 Related Work

Ekin *et. al* [10] research was one of the first relevant works on the subject. From images captured from TV broadcasting, the proposed system processed an adaptive and robust field extraction. Different events were automatically detected in the presented framework such as: goals, referee appearances and penalty box. The results presented showed an accuracy around 80% on shot classification, and around 90% to referee and penalty box detection.

Saito *et. al* [7] proposed a multi camera system for player tracking. In each one of the fixed and hand-calibrated camera players were detected using background removal. All detections were projected and clustered on ground coordinates. A Kalman Filter was used to estimate players

positions and smooth results. Results were then backprojected to image coordinate for user visualization. The clustering on ground coordinates allowed to solve cases of occlusion and get a perfect track of 16 players over 100 frames.

Ren *et al.* [5] presented a work for ball tracking using multiple fixed cameras. Detection is based on an adaptive background model using a pixel Gaussian Mixture Model. Tracking is made on image plan using a Kalman Filter and different observation states for each object. Tracking correction is applied to deal with merged objects by a matching process and forward filtering to unify different tracked objects to get the real trajectory.

Dearden *et al.* [9] work is based on particle filter for player tracking using images from a single camera of TV broadcasting. Players regions segmentation was carried out by a background histogram backprojection that allows dynamic upgrading of field model. The observation model of the filter includes the blob's positions and sizes as well color information. A particle filter is processed for each player being tracked on image plan allowing deleting and creating new tracks when players leave and get in image. This approach can also solve solve cases of occlusions especially when players are from different teams.

A different approach to solve occlusions and player identification was presented by Figueroa *et al* [4]. Once more, using multiple fixed cameras segmentation was performed using background subtraction resulting in blobs of one or more players. Authors used a graph to represent the blobs and minimal path searching to track the players. Each node stores information about the geometry of the blob, color, number of players, velocity of the object and the distance to the linked nodes. To identify cases of occlusions, this algorithm estimates the number of objects in a blob by grouping the nodes with common edges and the relation between blob area and position of nodes of the same group allows the estimation of the number of players segmented on the same region. The process of splitting the blobs works on the way as the occlusion identification. The results showed the accuracy of the process with 82% of the occlusions being solved automatically and only 6% of the frames needed manual tracking.

ASPOGAMO system [12] presented highly accurate results for extraction of player's trajectories from images of TV broadcasting of FIFA World Cup 2006. The system is capable to work with one or more uncalibrated moving cameras. Automatic dynamic calibration is achieved with a model based localization for estimating the missing extrinsic camera parameters on each frame. In order to overtake most of the difficulties of player detection the system used a set of probabilistic clues to calculate likelihood-maps for player locations. Candidate regions containing players are extracted using an usual background extraction with a grass color model. Then these regions are analysed in terms of color, compactness and height constraints. This allows to split players merged in the same blob, identify player's team and delete false positives. In the end, the observations are compared with a multiple hypothesis tracker's estimations. ASPOGAMO reaches 90% detection rate on not crowded scenes and 70% for overcrowded scenes. With just one camera, occlusions are difficult to solve and left on a post posterior module for data fusing with other camera source.

One of the most impressive and complete work is by Okuma *et al.* [8]. The area of research is the tracking and action recognition of ice hockey players from video images of a single not

fixed camera. For the tracking the authors used a boosted particle filter whose observation model included Histogram of Oriented Gradients (HOG) for shape classification and Hue-Saturation-Value (HSV) color histogram for color classification. Since the shape of the players changes over time, a template updating is executed to increase the accuracy of the method. The particle filter beside the prediction model based on the players dynamics includes also the detections of an Adaboost Cascaded algorithm [19] trained with a large amount of samples (around 6000). Results showed that the combination of the methods (HOG+HSV+Adaboost) performed better with an average tracking error of less than 0.2 meters.

One of the most relevant works on the area of indoor soccer is from Morais et al. [20]. Using multiple fixed cameras around the pavilion this system fuses the detections in each camera to locate the player position in the field using a particle filter. The detection is realized with a Viola and Jones [19] detector trained with 16.000 samples of Futsal players and 19.000 negative samples. Then, using the homography of each camera, each detection is projected into world coordinates. The last step of the observation model of the filter is to fuse the data from different cameras. An appearance model using HSV and Gradients histograms is used to reinforce the detection and dismiss false positives. The systems failed in cases of occlusions and overcrowded scenes of players of the same team. Quantitative results showed an average error of 0.6 meters.

In Santiago et al. [21] handball players' tracking is achieved using an acquisition system composed by two static cameras on the top of the sports hall. Player detection is performed by background subtraction and an *a posteriori* color analysis for team identification and false positive handling. The authors presented a dynamic color calibration process based on region growing and a set of fuzzy rules to categorize the teams colors subspace. The detections were used as measurements in a Kalman Filter for each player after projected into ground coordinates. The tracking rate achieved was around 95% showing the reliability of this system.

One of the last and most promising works in the area of players tracking in indoor sports was carried by Lu et al. [13]. The system uses only one moving camera. Basketball player detection is realized by a Deformable Part Model (DPM) with a RGB color classification for team identification and delete false positives. Tracking is performed by Kalman Filter and results are combined in a Conditional Random Field with weak visual cues as Maximally stable extremal regions (MSER), Scale Invariant Feature Transform (SIFT) and other features to improve player identification. Automatic camera to court homography is achieved using frame-to-frame homography combined with an optimization of the model fitting into a filtered edge map. The results presented were outstanding comparing to the rest of the work and considering the acquisition system: Tracking had a precision of 92% and a recall of 80%. Player Identification method reached an accuracy of 80% and the automatic homography had an error of less than 14cm on the basketball court.

2.2.1 Summary

The first systems used multiple fixed cameras around the stadium or sports hall covering all the playfield to overtake some of the segmentation and tracking problems. In these cases segmentation can be easily obtained by background removal. The background model shall be updated

due to external conditions and lighting changes. Camera calibration is necessary to map players in the world coordinates. When cameras are fixed, calibration can be easily estimated picking correspondent points between image and the field model.

Advanced methods on image processing and tracking allowed to decrease the number of cameras used in image acquisition and in some cases only one camera is used, most of the times the TV Broadcast one.

Tracking process must consider the noise on detection process and that measurements are not always available. Usually the dynamics of the players are modeled and complex observation models are taken into account. Kalman Filter and Particle Filters are the most common implementations but more works are using Optimization and Linear Programming to perform player tracking.

Most of the works found in literature focus mainly on players position and trajectories and barely include ball detection. Also, high level and collective performance information is extracted as well action as goals, passes or set pieces are excluded from the goals of most researches. From the technological point of view, relevant aspects are left out. Computation time and real time constraints are barely considered. Finally, all the image acquisition architectures use one or more fixed cameras and there's no relevant work using portable or moving systems for image capture.

2.3 Commercial Solutions

In this section some of the products found in the market of evaluation of sport events will be analysed. It is intended to evaluate these solutions in terms of functionalities, complexity of the system and the cost of implementation and license.

2.3.1 Prozone

Prozone¹ is a British company and one of the pioneers in the market of individual and collective performance evaluation in many sport events with a main focus on soccer. Among their clients list one can find some of the coaches with highest reputation worldwide. The variety of the provided services include real-time game analysis, performance evaluation and also scouting and opponent advising.

Prozone3 is the most reliable and expensive product of Prozone. A set of 8 to 12 cameras placed around the stadium captures the game action in the entire field and from different perspectives. The players, ball and referees are detected in a semi-automatic procedure. Despite using automatic image analysis in player detection, human intervention is required to correct the tracking and also for game events annotation such as: goals, passes, tackles, set pieces, etc.

The product provides relevant data about players position and trajectories from the entire game, individual and collective performance statistics and also trends and usual behaviours associated to players and teams.

¹ www.prozonesports.com

The whole image acquisition system installation has a cost of 120,000 euros and each game analysis has an additional cost.

2.3.2 Amisco

Amisco² a multinational company founded in 1995 that claim to be the leaders and main pioneers in player tracking technology. They work next to biggest professional teams worldwide and provide services of player recruitment, deep game analysis and live game advising.

This service provides real-time data about movement of all players and their interaction with the ball. More than 4 million data elements are collected from each game and are inserted in a global database. This information is obtained through image processing of sequences collected from an acquisition architecture composed by 6 to 8 HD cameras. TV broadcast records can be used to process a less set of data. All the process is supervised by a human operator to ensure the accuracy of the results.

2.3.3 STATS

STATS³ is one of the biggest companies providing information about sports events. The main clients are Media and Broadcasting companies as well some basketball and soccer professional teams.

2.3.3.1 STATS SportVU Tracking Technology-Football/Soccer

This system acquires in real-time the position of players and ball on the playfield. It provides a big variety of relevant information, as per example, distance ran, average velocity, ball possession time, etc.

The service has two distinct configurations. SportVU SV uses three HD cameras placed on a single place of the stadium. The other configuration is SportVU MV and includes 6 HD cameras placed on 2 places of the stadium. The first configuration detects only players and the ball while the second one provides a more robust localization and tracking as well catching 3D animations.

Data are processed in real-time for TV Broadcasting and showed by graphics and 2D/3D animations.

2.3.3.2 STATS SportVU Tracking Technology-Basketball

The product of STATS targeted to basket uses a 6 HD cameras configuration placed around the sports hall. It detects the position of the players and ball in at a rate of 25Hz. Human intervention in this process is residual and data about ball possession, dribbles, shots is processed in every 60 seconds in an automatic procedure. This system costs around 75,000 euros per year and it was acquired almost exclusively by NBA teams.⁴

²www.sport-universal.com

³www.stats.com

⁴National League of Basketball-www.nba.com

Table 2.1: Comparative Analysis of Commercial Solutions

| Products | Review Criteria | | | |
|-----------------------|--------------------------|--------------------|------------|---------------|
| | Image Acquisition | Human Intervention | Real-Time | Price (euros) |
| Prozone Playback 3 | 8-12 Cameras | Needed | No | >100,000 |
| Amisco Match Analysis | 6-8 Cameras | Needed | No | >100,000 |
| SportVU - Soccer | 3/6 cameras | Almost unnecessary | Yes(1-60s) | 70-100,000 |
| SportVU- Basketball | 4-6 cameras | Unnecessary | Yes(1-60s) | 70-100,000 |
| OptaPro DataScout | TV Broadcast | Exclusive | No | ? |
| Tracab | 2 sets of stereo cameras | Unnecessary | Yes | ? |

2.3.4 OptaPro

OptaPro⁵ sees itself as world leader on sports data providing. They include services of game analysis, opponent scouting and player recruitment.

The service of data acquisition and processing from OptaPro uses images from TV broadcasting to annotate all the relevant data during a football match. Each game is analysed by 3 people, each person for each team and the third one supervising all the process. All the processed data is inserted into a database that currently counts with more than 30,000 matches each one with more than 2000 events, including position, trajectories and ball interactions.

2.3.5 ChyronHego

ChyronHego⁶ is a Swedish company for multimedia contents providing. Its services include tv broadcasting support such as collecting and displaying data of sport events.

Tracab from ChyronHego is currently the only solution on the market capable of automatic and real-time detection of all players during an entire soccer match. The acquisition system is composed by two sets of stereo HD cameras placed on the top of the stadium in order to capture all the pitch. The system can extract the 3D position of all players and ball in real-time under different environmental conditions. Information about prices is not publicly available and recently all the stadiums of English Premier League were equipped with this product.

2.3.6 Summary

Existing solutions in the market have presented a high innovated technology and high reliability in their results, however, there's lack of robustness on their automatic procedures. Human intervention is still necessary in some of the procedures of detection and tracking. Almost all the solutions covered use multiple fixed HD cameras (table 2.1). The complexity of implementation and the cost associated with human operators make these systems expensive and not reachable to the majority of teams.

⁵www.optasportpro.com

⁶www.chyronhego.com

2.4 Quadcopters

The latest developments in several technologies such as sensorization, stability, control, communication, energy storage or materials, as well as its mechanic simplicity and easy manoeuvrability, led to unmanned aerial vehicles as per example Quadcopters have won great renown in several applications. In this section the comparative analysis of different Quadcopters and other UAV's will be presented in order to justify which one is suitable to the scope of the project.

2.4.1 Dynamics and Control

Quadcopters are systems naturally unstable [22] therefore, they are impossible to be handled without any auxiliary control system. Several sensors located in the vehicle or outside it provide information on pose and position to a processing unit. This unit will process a control algorithm so that each propeller supplies the necessary torque to achieve a certain position or trajectory.

The most usual configuration and the one will be used on this research is the 4 rotors mounted on a cross structure. Each pair of non-adjacent rotors spin in the same direction and the speed of each one will determine the direction and velocity of the Quadcopter.

The vehicle has 4 rotors so the control has 4 degrees of freedom, namely: Vertical acceleration, Pitch, Roll and Yaw. The structure allows to decouple the control of each of these variables.

2.4.2 Commercial Solutions

Currently, there is a big variety of UAV in the market. For this research a relatively cheap, small Quadcopter equipped with a camera capable of live streaming is intended. The criteria in this selection will be: price, robustness,

2.4.2.1 Ar.Drone 2.0

This Quadcopter from Parrot⁷ is one of the best sellers on the market due to its price, ease of use and also easy repairing of all the components.

Table 2.2: Ar.Drone 2.0 Technical Specifications

| | |
|----------------|---|
| Structure | Carbon Fiber and Outdoor Hull |
| Weight | 400 g |
| Autonomy | 18 min |
| Sensors | Gyroscope, Accelerometer, Magnetometer, Pressure Sensor, Ultrasonics, Vertical Camera for ground velocity |
| Communications | Wi-Fi |
| Camera | 720p 30fps. Low latency Wi-Fi Transmission |
| Observations | Fully Repairable. Open and documented communications protocol |
| Price | 300 euros |

⁷ardrone2.parrot.com

This is a widely popular solution on the market because it can be controlled easily through mobile devices with Wi-Fi. This is an important feature as it will allow the usage of its communications protocol fully documented from any computer.

2.4.2.2 **Dji Phantom**

This Quadcopter⁸ is widely used in cultural and sports events because of its easy control and great stability with GPS localization. Despite not having a camera, it allows a stabilization mount for a GoPro⁹.

Table 2.3: Dji Phantom Technical Specifications

| | |
|----------------|--|
| Structure | Carbon Fiber |
| Weight | 600g |
| Autonomy | 10 min |
| Sensors | GPS, Gyroscope, Accelerometer, Magnetometer, Pressure Sensor |
| Communications | Radio |
| Camera | - |
| Price | 500 euros |
| Observations | Possibility to equip with a gopro |

One of the big advantages of this vehicle is based on the use of GPS localization. However, in indoor places that's not possible and Dji Phantom will not use most of its features.

2.4.2.3 **AscTec Firefly**

Firefly is the last product of AscTec¹⁰ and it's considered one of the most advanced on the market. This Hexacopter (patented system with 6 rotors instead of the usual 4) is a product focused on autonomous flights using its HD camera. But its greatest obstacle is undoubtedly the price. This system is modular and allows to add and remove different components according to the goal of the flight.

Table 2.4: Firefly Technical Specifications

| | |
|----------------|--|
| Structure | Modular Structure |
| Weight | 500g |
| Autonomy | 10 min |
| Sensors | GPS, Gyroscope, Accelerometer, Magnetometer, Pressure Sensor |
| Communications | Xbee, 2.4GHz |
| Camera | 752x480-90fps |
| Price | 6.000 euros |
| Observation | Matlab/Simulink programmed |

⁸www.dji.com

⁹gopro.com

¹⁰<http://www.asctec.de/uav-applications/research/products/asctec-firefly/>

2.4.2.4 Choice

The products presented were evaluated taking as criteria: Price, Contrability, Robustness/Modularity and Camera Table 2.5 shows briefly the evaluation performed.

Table 2.5: Quadcopters Evaluation . '++' -Quite Positive. '-' -Quite Negative

| | Criteria | | | |
|-----------------|----------|----------|------------------|--------|
| Product | Price | Control. | Robust./Modular. | Camera |
| Ar.Drone 2.0 | ++ | + | - | + |
| Phantom(+GoPro) | + | - | - | + |
| Firefly | - - | ++ | + | + |

Having regard the the criteria presented above, the chosen solution was the Parrot's Ar.Drone 2.0 due to its low price, open communications protocol that lets you explore numerous possibilities as autonomous flights and even the HD camera with high quality and transmission rate.

Phantom was excluded because of its control system that difficults future implementations of autonomous flights on indoor environments.

Despite of its reduced robustness, Ar.Drone is fully repairable. This can not be neglected since this will be a testing platform and falls and impacts are expected.

2.5 Conclusions

Computational vision systems for sports analysis are currently indispensable tools for tactical and technical preparation of elite teams in many collective and individual sports. The usage of multiple cameras for image acquisition and the necessity of human intervention in some of the procedure inflate the prices of these system making them unreachable for the majority of the teams.

At scientific level, the optimization of segmentation, detection and tracking methods allowed the achievement of positive results. Automatic systems are capable to solve cases of occlusions and players leaving and entering the image plane with high accuracy. However, all the systems reviewed are based on expensive and complex image acquisition architectures. There is lack of studies based on low cost image acquisition systems turning this study an important contribution on this research area.

In the last few years, unmanned air vehicles, namely Quadcopters have become increasingly available to a wide range of applications. Usually equipped with high definition onboard cameras, these vehicles are suitable for capturing images of multiple events including sports. Their portability, easy manoeuvring and also a possibility for automatic flights make this low cost solution tempting for capturing images for automatic vision systems for performance extraction on sports.

Chapter 3

System Overview

In this chapter is presented an overview of the system. From the technological perspective is explained how images sequences from the games are acquired and processed. Conceptually, the framework with the different stages to extract the information of the game automatically will be described briefly.

3.1 Image Acquisition

The images from indoor soccer games used in this project are shot by the Ar.Drone's frontal camera. The Drone is controlled using Parrot's commercial application for mobile devices¹. For the purpose of this research the drone is hovering on a static position 5 to 7 meters above the floor, close to the side line of the pitch (figure 3.1). After a small modification on Ar.Drone's structure, its frontal camera is pointing 30° down in order to capture the game action and simultaneously avoid cases of occlusions. The sequences used in system development and testing cover only one half field. This approach would require other image sources to shoot the entire field. On the other hand this solution will facilitate some hard tasks, namely: automatic camera calibration and motion compensation. Other approaches like multiple drones or automatic game action tracking were not included taking into account the aim of this project.

The videos recorded by Ar.Drone's camera have 1280x720 resolution and a rate of 30 frames per second (fps). The sequences were shot in Pavilhão Luis Falcão² and Pavilhão Desportivo Politécnico do Porto³ during official amateur tournaments. In both tournaments, different shirt colors were used to identify the teams. Referees as well as spectators can appear on some these sequences.

¹[https://play.google.com/store/apps/details?id=com.parrot.free flight](https://play.google.com/store/apps/details?id=com.parrot.free+flight)

²<https://sigarra.up.pt/cdup/pt/>

³<http://www.ipp.pt/cde/index.php?id=60>

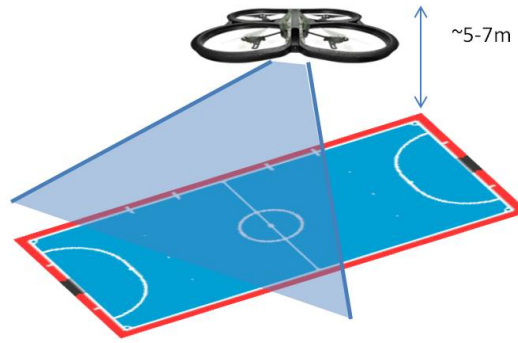


Figure 3.1: Representation of Quadcopter's position while recording the image sequences from indoor soccer games

3.2 Image Processing

Image analysis was performed offline using MATLAB⁴ including Computer Vision⁵ and Image Processing⁶ toolboxes.

Sequences of 30 seconds to 1 minute were selected to avoid situations with sudden movements of the air vehicle but including some other undesired but usual situations in this kind of systems, namely: players moving in and out of the image, occlusions, camera motion among others.

Different methods are used in the various stages of the process. The following framework represents conceptually each one of these stages in terms of the input, output and the main reason for utilization.

3.2.1 System Framework

In order to extract relevant information about the game from the image sequences captured by Ar.Drone's frontal camera a series of steps are carried out. Inevitably, the quadcopter used to collect images will suffer oscillations inherent to its own dynamic and external conditions. The first stage of the framework is motion compensation and image stabilization in which the frame-to-frame transformation will be estimated in order to compensate the motion caused by the drone instability.

The next step is camera calibration, this means how to relate image to world coordinates. This is a necessary step since all the information extracted from the image is only meaningful in world coordinates. Despite the image stabilization step, image-to-world homography will vary over time and there's a need for a dynamic automatic re-calibration procedure. This step will provide the homography matrix that relates players position in image to the world coordinates in each frame.

⁴<http://www.mathworks.com/products/matlab/>

⁵<http://www.mathworks.com/products/computer-vision/>

⁶<http://www.mathworks.com/products/image/>

Player detection is undoubtedly a critical step for the whole system. Basic segmentation methods commonly used are not suitable for this research due to camera motion and non existence of a dominant background color. Instead, detection based on histogram of oriented gradients is used to detect people in upright position. Detections will not be always available and there will be also some false positives. Consequently, the next stage of the process is to filter the results of the detection and estimate the position of the players when detections are not available. Different information can be used to predict that, as color, movement or player's dynamics.

Since there are reliable results of player's positions and trajectories, high level information can be interpreted from the data as for example: heatmaps, offensive/defensive trends, among others.

This system framework is illustrated in the figure 3.2 representing the main processing stages and also the input and output of each one.

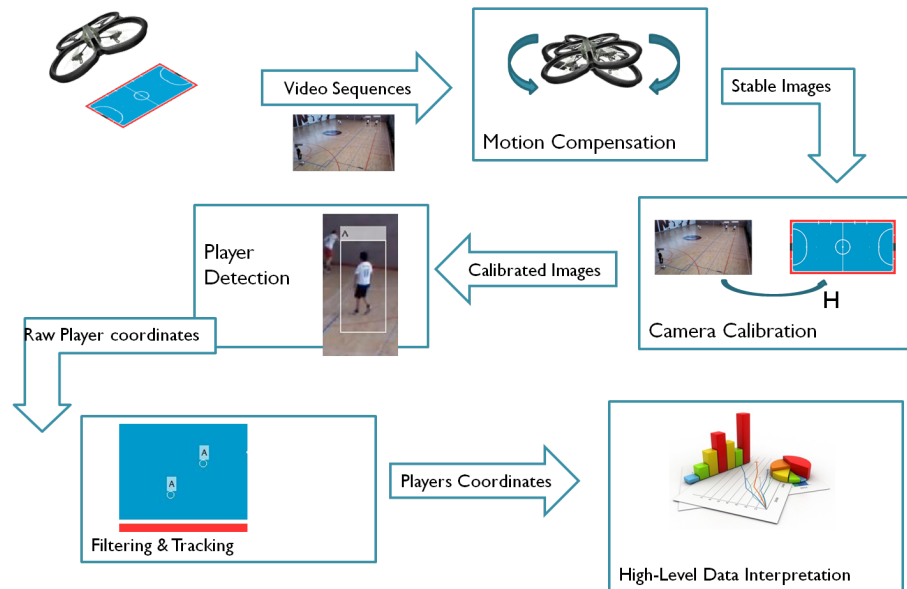


Figure 3.2: Schematic of the system framework

3.3 Methods Evaluation

The different methods and the different stages of each method will be evaluated using quantitative and qualitative criteria regarding the nature of the problem. Taking in account the criteria used on the evaluation, the following division can be presented:

- Quantitative Evaluation:
 - Camera Calibration
 - Player detection
- Qualitative Evaluation:

- Video Stabilization
- High level data

3.3.1 Quantitative Evaluation

"Camera Calibration" and "Player detection" are stages of the framework in which a metric evaluation can be used. The results of different methods will be compared to manual annotated ground truth data in both cases. Ground truth annotation was carried out using a MATLAB script that allows to annotate at a pre-determined frame rate the position and team of the players and also the four points needed to define the homography between the field and the image plane. Then the data are interpolated in all the frames. Manual annotation in each 10th frame performed visual acceptable results. At this stage it is important to refer that manual data annotation will have noise that will appear on the final results.

3.3.1.1 Camera Calibration Evaluation

The metric used to evaluate the calibration method will be the distance in pixels of the four corner points of the half field rectangle. These four points are used since they usually appear in all sequences and are enough to define the world-to-image homography. Let us define $X_i = (x_i, y_i)$ with $i \in [1, 2, 3, 4]$ as the four points manually annotated in a frame t and $\tilde{X}_i = (\tilde{x}_i, \tilde{y}_i)$ as the four points resulting of the camera calibration method on the same frame. The error of the calibration method can then be expressed as:

$$error_t = \sum_i \sqrt{(x_i - \tilde{x}_i)^2 + (y_i - \tilde{y}_i)^2} \quad (3.1)$$

3.3.1.2 Player Detection Evaluation

In computer vision bounding boxes are commonly used to represent the location of a certain object on the image. In this project the same representation is used. The different methods for player detection will be evaluated in terms of precision and recall, defined next:

$$precision = \frac{detections}{detections + falsepositives} \quad (3.2)$$

$$recall = \frac{detections}{detections + misseddetections} \quad (3.3)$$

Since in this research there is no focus on player identification there will not be a direct link between each detection and the ground truth equivalent. To solve this problem Munkres algorithm is applied [23] to assign detections to ground truth data. The cost calculation is based on the distance and size between bounding boxes.

To be assigned as a correct detections different rules can be used. In this study one rule generally used in computer vision problems will be applied: Let us define A as a detected bounding box and B the assigned ground truth annotation, then the detection is classified as correct if:

$$\frac{Area(A \cap B)}{Area(A \cup B)} > 0.5 \quad (3.4)$$

Team classification will be evaluated comparing the teams between the detections assigned and the corresponding annotated players.

In this process it is possible for a player to be misclassified due to the ground annotation method, since the interpolation and assignment are automatic and not controlled by the user. But decreasing by ten the number of ground annotated data, visual interpolated results seemed reliable.

3.3.2 Qualitative Evaluation

Video stabilization in computer vision is hard to evaluate quantitatively so in this project the results of this stage were evaluated qualitatively and with subjective criteria such as motion compensation, long term accuracy and efficiency.

High level data interpretation is a very subjective topic and requires highly expert knowledge about the different aspects of the game to have an accurate evaluation. In this project evaluation is kept basic and based in common knowledge about indoor soccer tactical and technical aspects.

3.3.3 Test Sequences

To test the methods and their robustness sequences on indoor, soccer venues during games or warm ups of official amateur tournaments were recorded. It was intended to cover different circumstances and deal with usual difficulties on this kind of systems. Four different video sequences to test the different stages of the methods will be used , which are described next:

- **Sequence number 1⁷**: Shot in Pavilhão Luis Falcão during team warm up. In the field there are eight players from the white team and three from the black team. At the 5th second the drone suffers a strong oscillation.



Figure 3.3: Frame 1,100 and 500 of original sequence number 1.

⁷<https://www.youtube.com/watch?v=3VDAR10wqDM>

- **Sequence number 2⁸**: Shot in Pavilhão Luis Falcão on a non game situation. Only two players on the field without wearing a team kit.



Figure 3.4: Frame 1,100 and 500 of original sequence number 2.

- **Sequence number 3⁹**: Shot in Pavilhão Desportivo Politécnico do Porto during an official game. Two different teams of four field players wearing black and white equipments. This sequence suffers from bad illumination reflection of the floor.



Figure 3.5: Frame 1,100 and 500 of original sequence number 3.

- **Sequence number 4¹⁰**: Shot in Pavilhão Desportivo Politécnico do Porto during an official game. Two different teams of four field players wearing yellow and orange equipments. Some players from different teams wear similar shirt colors.



Figure 3.6: Frame 1,100 and 500 of original sequence number 4.

3.4 Conclusions

For the purpose of this research a simple approach is chosen to capture images from indoor soccer games. The drone will be hovering a static location capturing the game action in only half of the field. To get the action of the entire field more complex approaches would be required, namely the use of multiple drones or an automatic ball tracking system where the drone would rotate to

⁸<https://www.youtube.com/watch?v=v90UmhYzCAo>

⁹<https://www.youtube.com/watch?v=tUpY8VHWWaw>

¹⁰<https://www.youtube.com/watch?v=TNCKiUTyqIk>

follow the ball. A Drone will be hovering on a high location to simultaneously increase the visible area and avoid cases of occlusions, otherwise, some players will be too distant from the camera making their detection difficult.

The first image sequences collected showed immediately an undesired image jittering due to motion of the drone. This will difficult posterior processing specially camera calibration and player tracking since they use spatial coherence on the image. An important aspect is that players from different teams are wearing different colors which be useful for the team identification. Still, since not the entire field is being covered by the drone's camera, players will be continuously entering and leaving the image plane, which will bring more challenges to players detection and tracking.

The following chapters will describe in detail the various challenges and stages of the framework and show some of the methods proposed to solve them.

Chapter 4

Video Stabilization and Camera Calibration

In this chapter the methods developed to achieve video stabilization and estimate camera calibration parameters and the corresponding preliminary results will be presented. Briefly, the relationship between the camera and the world will be studied. As seen previously, due to unavoidable drone's motion the image sequences will have undesired motion noise. In our system video stabilization is especially important since calibration and player tracking methods will be using information of pixel's intensities values but also pixel's position, so it is important to keep spatial consistency among the image sequences. The transformation between camera and world coordinates can be described by the intrinsic and extrinsic camera parameters. This transformation is important to relate the position of the players on image coordinates to their real location on the field. Even with image stabilization procedure, not all the motion noise is removed so a dynamic calibration method is required to update the transformation between coordinate systems in each frame.

4.1 Video Stabilization

Image stabilization lies on the process of compensating undesired motion of the camera. The first stabilization system was presented by Canon ¹ in 1995 and was based on a moving lens with a 16-bit microcomputer controlling an ultrasonic motor. This approach is quite expensive and not suitable for the majority the cameras.

Instead, digital image stabilization is widely used due for the development of computational resources and image registration methods. Generally, the digital video stabilization process is split in 3 main stages [24] : motion estimation, motion compensation and image composition.

¹cpn.canon-europe.com/content/education/infobank/lenses

4.1.1 Motion Estimation

The first stage of video stabilization is to estimate frame-to-frame motion also denominated global motion [25]. Two main approaches are used in this step: feature-based and global intensity alignment. The last one is usually slower and less accurate because it does not take advantage of important visual clues between frames since it uses global pixel-to-pixel comparison. By collecting several interesting and invariant points or regions it is possible to match them on different frames.

4.1.1.1 Features Detection

Feature-based methods use detection of distinctive key points and posterior matching between frames to estimate their transformation [26]. A local feature is an interest point or region and it is associated to a change on a certain image property [27]. Some features detectors are invariant to the most of transformations, this means that their location does not vary over time or perspective (repeatability) [26]. Local features can also be evaluated in terms of informativeness, locality, quantity, accuracy and efficiency but repeatability can be defined as the most important of them all [27]. Local features can be divided in three main classes: corners, blob and region detectors.

Corner points are widely used as features in video stabilization because they are very sensitive to changes in both directions of image but they also provide a large set of stable and repeatable features. [27]. Blob detectors are other kind of features widely used. They are commonly applied complementary with corner points. Salient regions can provide useful information about frames transformations but they are not so accurately located on image as corners [27]. Since a blob is located by its boundaries that makes it less accurate and consequently less suitable for frames comparison or camera calibration applications. Region detectors are suitable to represent homogeneous regions but they lack accuracy in matching and description despite being quite acceptable on detection. To increase matching accuracy, region descriptors usually include information on its boundaries and capture the shape of the region. These detectors work well in a reasonable structured image with well defined objects, otherwise, they lack repeatability. They are often used to ease computation analysing using similar areas instead of single pixels.

All the features detectors presented above are computationally complex and barely respect real time constraints. However, there are some implementations that stand out with their computational efficiency [27]. Difference-of-Gaussians detector (DoG) implements an approximation of the Laplacian based on the difference of the image at different scales avoiding the second-order derivatives which are computationally expensive. The image is smoothed with a Gaussian filter several times and then Laplacian is approximated by the difference of the smoothed images. [27]. The regions extracted are the local maxima in the difference of Gaussian maps. This process can extract interest regions with an interesting frame rate, useful then to video processing.

Speeded Up Robust Features or SURF [28] is another example of a robust detector. It uses an approximation of Gaussian derivative kernels by box filters based on Haar wavelets. These filters

are used to calculate the Hessian matrix (partial second derivatives). The determinant of this matrix represents blob response at a certain location. Local maxima responses are then interpolated resulting on the detected features. This method presented results five times faster than DoG [27].

FAST detector is an efficient corner detector [29] based on the analysis of the neighbour pixels around a central location. It considers a circle of fixed radius and compares opposite pixels above that circle. Those pixels are classified taking in account their similarity. Measuring the entropy of negative and positive responses it is possible to create recursively a subset of null entropy. These will be the best candidates to be corner features. This detector can achieve a speed up to 30 times faster than DoG [27] despite not being invariant to scale changes.





In brief, feature detection is an essential step of motion estimation. Using interesting points or regions invariant to rotation or other transformations it is possible to detect them in different images as adjacent frames. Repeatability, location accuracy and quantity are, probably, the most important qualities of a feature detector for video stabilization. Point features as corner detector usually perform better on this kind of application even being not invariant to scale change but their repeatability and location accuracy stand out comparing to region or blob detectors. Feature extraction is a very complex process in terms of computational effort, so recent methods try to approximate some of the calculations behind the process speeding up the detections without losing accuracy. After detecting the interesting points/regions in different images it is then necessary to match them to extract the geometric relation between them.

4.1.1.2 Features Matching

The process of matching two or more sets of features collected from different images is a hard task since there is neither prior knowledge about the correspondence pairs of point nor the geometric transformation that relates both sets. Moreover, the features collected will have inliers and outliers, this means that there are features that do not have a reasonable correspondence in another set. This happens because different images cover different interesting points/regions and features detectors are not perfectly repeatable or accurate. Once there is the match between the sets, finding the geometric transformation is easy. However finding all the inliers matches can be an expensive task in terms of computational power so efficient methods are required in this step. RANdom SAMple Consensus (RANSAC) is an iterative method for fitting of data sets containing many outliers [30]. Since it is a non deterministic method there is no certainty to find the optimal result. However with enough iterations there is an high probability for a good fitting. The algorithm can be expressed in three steps: 1) N points are randomly selected and model parameters are estimated from them ; 2) the other points are compared to the model parametrized before. The points that fit the model are included in the consensus set; 3) If this set is large enough the fitting is concluded, if not, step 1) is applied including the consensus set in the initial estimation. The big limitation of this method is the inexistence of a computation time limit. However this is a very robust method for fitting models with many outliers (<50%).

Another method to fit a given model is Least Median Squares (LMS). The goal of this algorithm is to minimize the median of the squared residuals (difference of the points to the model

Table 4.1: 2D transformations hierarchy [2]

| Group | Matrix | Distortion | Invariant properties |
|---------------------|--|---|--|
| Projective 8 dof | $\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$ |  | Concurrency, collinearity, order of contact : intersection (1 pt contact); tangency (2 pt contact); inflections (3 pt contact with line); tangent discontinuities and cusps. cross ratio (ratio of ratio of lengths). |
| Affine 6 dof | $\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$ |  | Parallelism, ratio of areas, ratio of lengths on collinear or parallel lines (e.g. midpoints), linear combinations of vectors (e.g. centroids). The line at infinity, l_∞ . |
| Similarity 4 dof | $\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$ |  | Ratio of lengths, angle. The circular points, I, J (see section 2.7.3). |
| Euclidean 3 dof | $\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$ |  | Length, area |

fitted). The first step of this method is to pick a fixed number of subsets with different points. Then, for each subset model parameters are estimated. Next, the median of the residuals is calculated based on the estimated parameters. The parameters with minimum median of the squared residual are chosen. The fundamental issue in this method is how to guess the number of initial subsets and usually it is estimated in terms of the probability of outliers.

Once there is a match between the two sets of features, the next step is estimate the geometric transformation that relates both group of points.

4.1.1.3 Geometric Registration

The final step of motion estimation is to find the geometric relation that relate the features matched between the two image sources. Transformation can be defined by a 3x3 matrix and classified in terms of its degrees of freedom. The most general transformation is the perspective and relates the homogeneous coordinates \tilde{x}' and \tilde{x} :

$$\tilde{x}' = k * \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix} * \tilde{x} \quad (4.1)$$

All the transformation between 2D coordinate systems can be described with the matrix above: k is the scale factor and h_{ij} the parameters to estimate. However, other transformations can be considered as subsets of the one above with decreasing degrees of freedom. For instance, affine (6 degrees of freedom) transformations are widely used in video stabilization instead of perspective because of its model complexity and descriptive capability [31]. Table 4.1 shows the 2D transformations hierarchy.

Once a transformation model is chosen, the parameters can be estimated with the features matched previously. Linear regression methods as least squares can be performed to minimize the

sum of squared residuals [26]. Most robust methods can be performed assuming that not all the points are matched with same accuracy, which is true most of the times. In this case, the goal is to minimize the weighted least squares, where each point match has a variance associated [26]. In the case of a perspective transformation as the relation is non linear numerical methods have to be performed iteratively to find the parameters.

4.1.2 Motion Compensation

Several methods can be found to long term motion compensation from inter-frame estimated motion. Usually it is intended to eliminate or smooth high frequency jitter from estimated camera motion [32]. The model transformations presented above are approximations to modelize the camera movement. However, due to its lack precision there will be always noise in the process of estimation and also in the model itself.

The most simple method is to consider that the transformation at $frame=i$ is the chain of inter-frames transformation from $frame=1$ to $frame=i-1$. The biggest problem with this approach is that the error will accumulate over time. A solution to smooth this undesired effect is to use a fixed number of previous frame on the transformation chain [25]. Other more robust methods can be found in literature as Kalman Filter [31] or parabolic fitting [33] that will deal better with the cumulative error well as high frequency jitter from camera motion turning video reproduction softer and visually pleasant [33].

4.1.3 Image Composition

Image composition is the final step on video stabilization process and it resumes to how to present the compensated frame on the image plane. The most primitive method is to apply the inverse of estimated motion transformation in the image and fit the result on the coordinates of the first frame (blending). This results on the loose visible area over time. Other primitive method is to overlap the resulted compensated frame on the previous frames. This works well only on planar images, otherwise there will be a big degradation on the borders between frames. Robust methods were developed to deal with image degradation and stabilization as dynamic programming[33] or motion in painting [25]. The first method performs energy minimization of overlapped frames and weighting the contribution of boarder pixels resulting on a mosaicking with pleasant boarder fitting. The second method fills the missing area of compensated images after blending with local motion estimation relative to the previous frame.

4.2 Results of Video Stabilization

Due to external factor and its own dynamics, a Drone's camera has an undesired motion that needs to be compensated to ease some of the procedures to be carried out later. The stabilization process was implemented using common functions found in Matlab library and adapted to this application.

The method developed was based on feature matching. Corner points detected by the FAST algorithm were used because of its efficiency even though in this project time constraints are not considered. Other features detectors as SURF or Harris corner detector were tested with visible worse results. The detectors were performed on the grayscale map because of computational simplicity (figure 4.1).

M-estimator Sample Consensus (MSAC) a variant of the RANSAC method was used to match the features among the frames and exclude the outliers from the model fitting. As expected, most of the inliers are features from the pitch lines or walls draws. Some of the features were located on players and most of them considered as outliers.

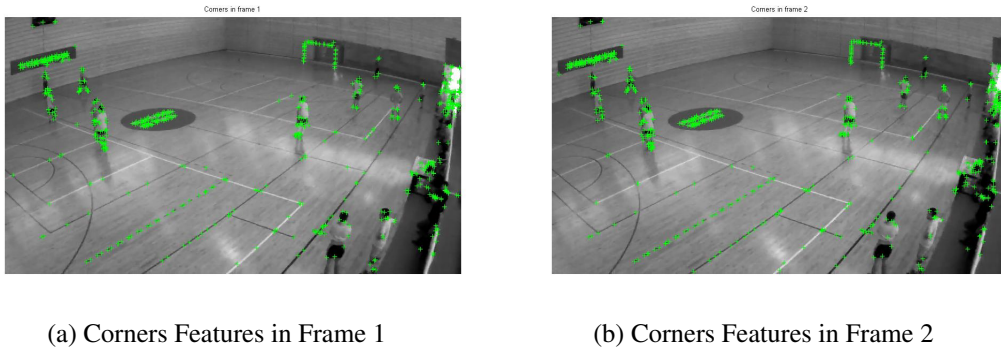


Figure 4.1: Result of FAST corner detection in Frame 1 and Frame 2 of video sequence nr.4.

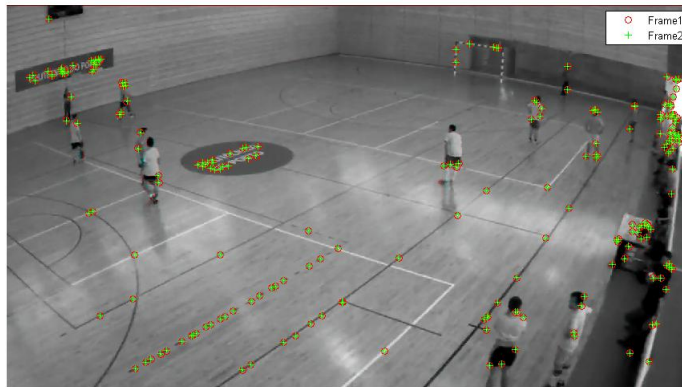


Figure 4.2: Result of Features Matching using RANSAC

After the features matching the transformation between the two sets of inliers was estimated using least squares method. On the first experiments, an affine transformation was chosen to model this transformation because it was descriptive enough of the general distortion between two 2D frames from a 3D scene frames. Thereafter it was noticed that if the model of the transformation was reduced in terms of degree of freedom to a Similarity (sRT) model (see table 4.1), the stabilization process would be more stable and more simple computationally. In a sRT there are only four parameters instead of the six of an affine transformation. Between adjacent frames the

difference could not be noticed and in long term the video would be smoother because noise of features detection and matching processes did not interfere the transformation parameters.

Video stabilization of the video over time is performed with the chain of transformations until then. Let us define H_i as the affine transformation that models the distortion between frames i and $i + 1$ so that:

$$H_{cumulative} = \prod_{j=0}^{i-1} H_j \quad (4.2)$$

This is a very simple approach for motion smoothing because there is error being accumulated over time but the proposed method performed well enough for video sequences of 30 to 60 seconds and the main purpose of the stabilization is to eliminate high-frequency jitter because of local player tracking and camera calibration procedure. Low-frequency camera motion is not important in this system.

Finally, image composition is achieved warping the currently frame using the cumulative transformation $H_{cumulative}$ and using the coordinate system of the initial frame. It causes a decreasing of visible area during time.



(a) Mean of non-stabilized sequence (1 second)

(b) Mean of stabilized sequence (1 second)

Figure 4.3: Comparative analysis of the stabilization method. On a) the mean of the first 30 frames of the sequence nr.1. b) The stabilized version of the same sequence

The method developed allowed to eliminate high frequency jittering and to compensate almost all of the short term camera movement. However, low-frequency movement is not compensated since using a cascade approach, error is being accumulated over time. The main cause of this error is the geometric transformation chosen to model the frame-to-frame movement. A similarity model that allows to reduce the influence of the noise and also smooth image composition is used but it does not represent exactly the real transformation. The method developed fails also in cases of strong drone's oscillations because of the reduction of visible image area losing visual relevant information which is very prejudicial to player tracking and camera calibration.



Figure 4.4: Mean of stabilized sequence nr. 1 for the first 8 seconds.

On figure 4.4 is possible to observe that over time the method proposed can not compensate all the camera motion. This will lead to necessity of automatic camera calibration since the relation between field and image coordinates will not be static.



Figure 4.5: Response of video stabilization to a strong oscillation

Figure 4.5 is an example of an usual result of video stabilization method in case of strong oscillation. The use of a simple image composition method is not suitable when big movement is compensated. The disadvantage is the loss of relevant visible area which will lead to loose of player tracking and camera calibration method.

4.3 Camera Calibration

Camera Calibration is the process of finding the transformation parameters of a point in the world coordinates $M = [X, Y, Z]$ to the correspondent one in the image plane $m = [u, v]$. This is a necessary step when is intended to extract information from the image plane in the world coordinate context [34]. In our research this will be especially important since there is the necessity to extract the position of the players on the pitch from the image sequences.

The relation between a point in the world and its image projection can be expressed by:

$$s\tilde{m} = A \begin{bmatrix} R & t \end{bmatrix} \tilde{M} \quad (4.3)$$

$$\text{where, } \tilde{m} = \begin{bmatrix} m & 1 \end{bmatrix}' = \begin{bmatrix} u & v & 1 \end{bmatrix}' \quad (4.4)$$

$$\text{and, } \tilde{M} = \begin{bmatrix} M & 1 \end{bmatrix}' = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}' \quad (4.5)$$

This relation is taken from the pinhole model and s is a scale factor, $[R, t]$ represent the extrinsic parameters of the camera that is the rotation and translation which relate the camera to the world coordinates. A is the intrinsic camera matrix:

$$A = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.6)$$

where α and β represent the focal length in pixels unit, γ is the skew factor between the two axis and u_0 and v_0 are the centre of image.

4.3.1 Lens Distortion

Lens or radial distortion is a non negligible aspect of camera calibration and model the deviation of point in the image from the its rectilinear projection. It causes that a straight line on the world does not be straight on the image. The most common model for lens distortion assumes a non linear relation parametrized by the radial components [34]. Let us assume a point (x, y) a coordinate of the undistorted image and (x', y') the corresponding for the distorted image. They can be related by:

$$x' = x + x[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] \quad (4.7)$$

$$y' = y + y[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] \quad (4.8)$$

$$(4.9)$$

where k_1 and k_2 are the coefficients of the radial distortion. For this research we will keep the expression for the lens distortion to its quadratic form. More parameters could be required but for this project it will only be used k_1 and k_2 .

In all, there are 13 parameters (5 extrinsic, 6 intrinsic and 2 radial parameters) to relate a point in the world and image plane. Several different approaches to estimate these parameters can be found. The most simple one is Direct Linear Transformation (DLT) and it is performed where points location on world model and its correspondent location in the image is known. Usually the pair points are located manually in different images. Tsai proposed a two stages method for camera calibration [35]. It also uses known correspondent points in world and image coordinates and firstly, estimate position and orientation of the camera and then the internal parameters of the camera. An automatic camera calibration method was proposed by Zhang [34] it uses different 3D, 2D or 1D objects to be shot and estimate camera parameters from different images. The most usual technique is to shot a planar pattern on, at least, two different orientations.

Returning to the particular problem of this research as the relation between players position on image and pitch, we can assume the our model plane is on $Z=0$ and:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (4.10)$$

$$with, H = \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \quad (4.11)$$

H is the homography matrix and represents the perspective transformation of the 2D plane in the world coordinates to the image. As seen previously, this transformation model has 8 degrees of freedom and it is defined by a 3x3 matrix for a scale factor.

4.3.2 Solving Field-to-Image Homography

As seen previously, to find the relation of players' position in the image and in the field is required eight parameters of the homography matrix that model the geometric transformation of the plane field in the world and the plane of the image.

Several methods to solve Field-to-image homography can be found in literature and differ in terms of complexity attending to the architecture of image acquisition system. When fixed cameras

are used the homography in each camera is static and can be estimated using corresponding points usually picked manually (at least four pairs of points) [7, 4] and then solve the homography by least squares method. When moving cameras are used more complex methods are required to solve the homography. After a manual initialization solving the initial field-to-image homography the solution for the posterior frames can be estimated using the chain of frame-to-frame homography using some of the methods described in the previous section [12, 36, 13]. Over time, frame-to-frame homography tends to accumulate error and field-to-image homography starts drifting. Usually models of the field using the lines and other visual clues are utilized to fit them in the image using edge detectors or other borders enhancement methods [12, 36, 13] and with ICP or Lavenberg-Marquardt algorithms for point matching and homography estimation.

A simple but still robust group of methods lie on identification of particular points both in image and model [37, 38, 39]. In many sports the court is identified by multiple lines which intersect each other in different points. These lines can be easily identified using algorithms as Hough transform [40] and then calculate the intersection points between a group of two pairs of parallel lines and orthogonal between them on the world even though in image this relation is not achieved due to perspective transformation. This will give four pairs of corresponding points in image and model, enough to calculate the image-to-model homography.

4.3.2.1 Hough Transform

Hough transform is usually used for line detection, such as straight lines or circles and ellipses. Even being computationally expensive it is a very robust method for line detection, working in cases of occlusions and noise[41].

Taking Hough transform to the particular case of analytical shapes as straight lines, let us define a point $[x_i, y_i]$ in Hough space:

$$x_i \cos \theta + y_i \sin \theta = \rho \quad (4.12)$$

This means that a point in the image is described by the N lines passing through it (depends on the resolution of the accumulator) and it corresponds to a sinusoid in the Hough space. In the Hough space a line is defined by its polar parameters (ρ, θ) instead of the usual Cartesian parameters (m, b) :

$$\begin{aligned} y &= mx + b, \\ y &= \frac{-\cos \theta}{\sin \theta} x + \frac{\rho}{\sin \theta}, \\ \text{re-arranging, } \rho &= x \cos \theta + y \sin \theta \end{aligned} \quad (4.13)$$

In Hough space the intersection point of multiple sinusoids corresponds to a line in the image. To achieve robust results the algorithm to detect lines uses an array of discretized intervals of the

parameters ρ and θ . The resolution of these parameters is important: higher resolution allows to distinguish lines while lower resolution performs better in case of noise in the image. For each pixel on a binary image the array of the Hough parameters is incremented taking in account the expression previously described. Finally the algorithm needs to find the local maxima higher than a certain threshold, so that they correspond to all relevant lines on the image.

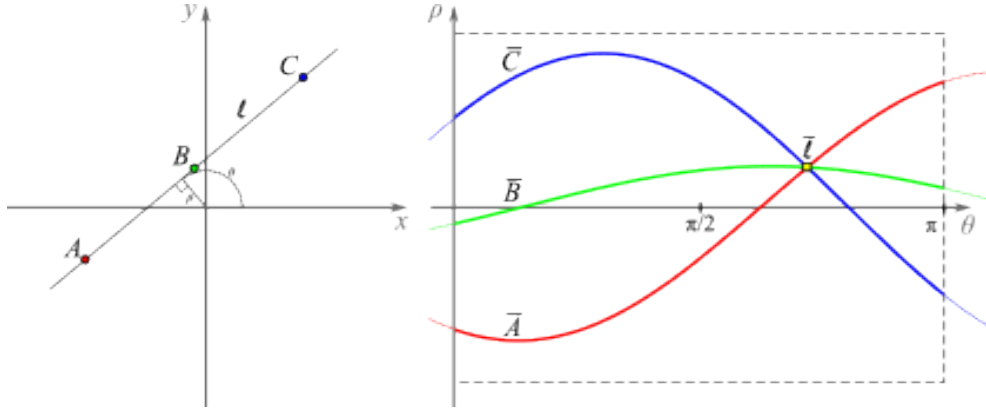


Figure 4.6: Transformation of line and points from image to Hough space. Points in image corresponds to sinusoids in Hough Space and Points in Hough Space to lines in image [1]

4.4 Results of Camera Calibration

In this section the method developed to perform camera calibration and the corresponding results will be presented. The homography between the field plane and the image that will relate players position on the image to their actual position in the world coordinates are intended in the specific case of this research.

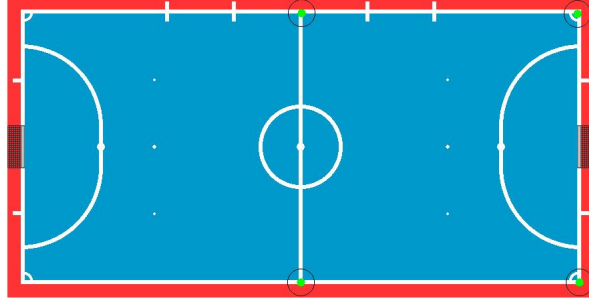
In our method an image of field scheme will be used to represent the world model. In this image we will only be interested in the ratio height/width of the field, other lines as penalty box or circles will be neglected. In this process manual initialization is required and then the algorithm automatically processes the image-to-world homography using line matching in Hough space. Radial distortion is compensated using the radial distortion parameters available on the Ar.Drone's documentation². Actually, this step is performed before the stabilization procedure and it allows to use straight lines on the calibration stage. However, visible area is highly reduced with this process. On the further stages of the system these calibrated images will be used as input but it can be discussed if the use of undistorted images is favourable or not.

4.4.1 Initialization

In the beginning of the process a manual intervention to estimate the initial image-to-field perspective and the creation of the virtual field model is required for the user.

²<https://projects.ardrone.org/>

On the following step, least squares algorithm is performed to calculate the eight parameters of the perspective transformation. Since Drone's camera is covering only half of the field the points to pick can correspond to the corners of half field (figure 4.9).



(a) The four corners on the model of the half field used in calibration initialization



(b) The four corners on the image of the half field used in calibration initialization

Figure 4.7: Example of the 4 pairs of corresponding points to calibration initialization

Back with the notation used before let us define the point in the image represented in homogeneous coordinates $\tilde{m} = [x, y, 1]^T$ and the corresponding point in the field plane $\tilde{M} = [X, Y, 1]^T$. They are related by the following expression:

$$\tilde{m} = H\tilde{M} \quad (4.14)$$

$$\text{where, } H = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix} \quad (4.15)$$

This relation can be written as:

$$x = \frac{h_{11}X + h_{12}Y + h_{13}}{h_{31}X + h_{32}Y + 1} \quad (4.16)$$

$$y = \frac{h_{21}X + h_{22}Y + h_{23}}{h_{31}X + h_{32}Y + 1} \quad (4.17)$$

Now, for the case of four pairs of corresponding points the relation can be expressed algebraically:

$$\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -y_1X_1 & -y_1Y_1 \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -x_2X_2 & -x_2Y_2 \\ 0 & 0 & 0 & X_2 & Y_2 & 1 & -y_2X_2 & -y_2Y_2 \\ X_3 & Y_3 & 1 & 0 & 0 & 0 & -x_3X_3 & -x_3Y_3 \\ 0 & 0 & 0 & X_3 & Y_3 & 1 & -y_3X_3 & -y_3Y_3 \\ X_4 & Y_4 & 1 & 0 & 0 & 0 & -x_4X_4 & -x_4Y_4 \\ 0 & 0 & 0 & X_4 & Y_4 & 1 & -y_4X_4 & -y_4Y_4 \end{bmatrix} * \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ x_4 \\ y_4 \end{bmatrix} \quad (4.18)$$

$$\text{simplifying : } A * h = b \quad (4.19)$$

The solution is given by least square method and can be expressed as:

$$h = (A^T A)^{-1} (A^T b) \quad (4.20)$$



Figure 4.8: Model of the field projected on the first frame using the initial homography

Another initialization step required to the user is the creation of the virtual model of the field which will be marked all the lines in the field. In this research we will assume that most of the indoor sports venues are used for multiple sports so that all the lines of the courts are painted in the

floor. In this research all the lines will be used to estimate automatically the camera calibration, even if they do not belong to indoor soccer court's lines. Because the line detector presented next cannot detect all the lines over the frames this step must be manual. This process could be done automatically with an iterative updating model; however, in order to keep simplicity in this process it was preferred to manually introduce all the detectable lines in the field identifying the horizontal and vertical lines. Lines selected are posteriorly extended until image borders to extract more intersection points.

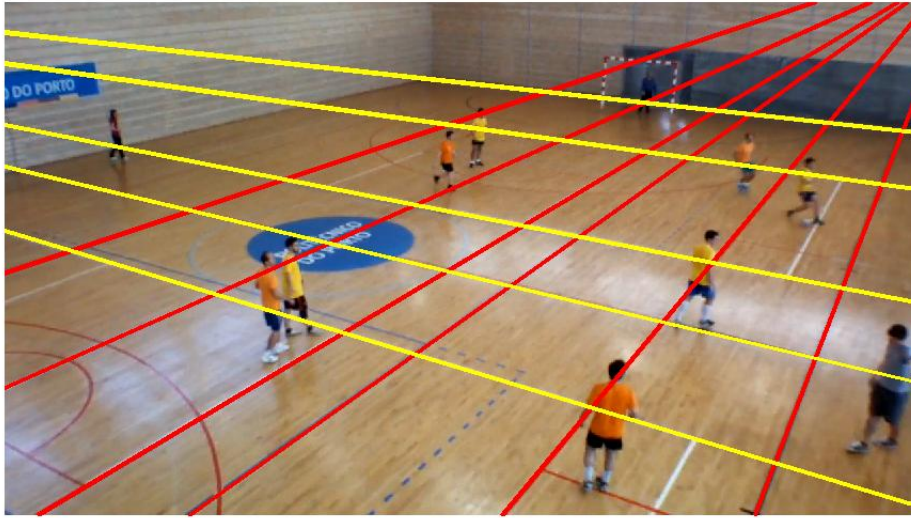


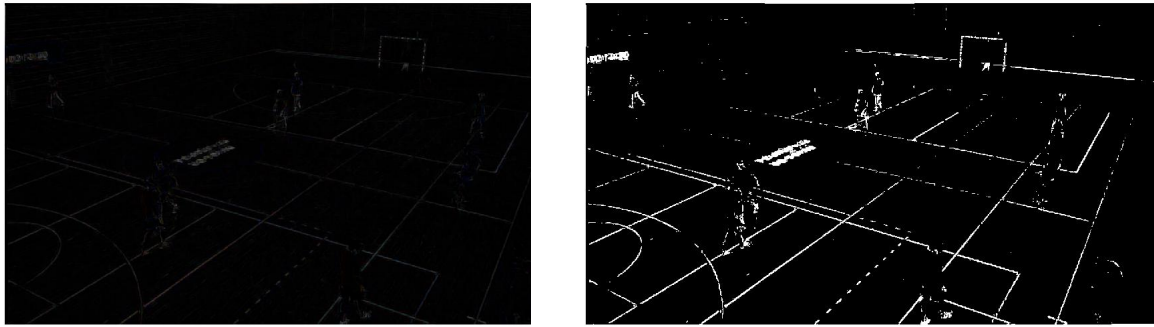
Figure 4.9: Example of virtual model created manually

4.4.2 Line Detection and Matching

To estimate automatically the image-to-world homography our algorithm will detect lines on the current frame and match them to the lines of the virtual model created on the beginning of the process. Then, two pairs of lines orthogonal to each other in world coordinates will be picked and their intersection points will be calculated. These four points will be used then to calculate the homography between the current frame and the model.

Line detection is performed using Hough transform as described before. The detector performs over a binary image obtained with a morphological gradient transformation (result of the subtraction of the result of opening operation by the result of closing on the current frame) enhancing the edges of the image. Posteriorly, image binarization is performed using Otsu's method [42]. This method performed better than other usual edge detectors as Canny or Laplacian of Gaussian. Being them more immune to noise and false detections, detection rate will be otherwise smaller.

After binarizing the image, Hough transform is performed with a MATLAB implementation over all the white pixels and the parameters accumulator updated taking in account the process described on the previous section. In this procedure both θ and ρ use a accumulator with unitary



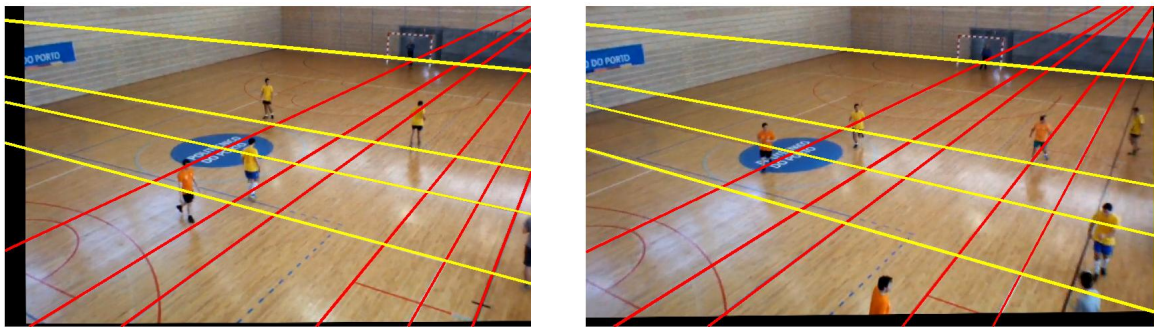
(a) Result of the Morphological Gradient

(b) Result of the binarization using Otsu's Method

Figure 4.10: Example the pre-processing method to obtain a binary image to be used in line detection

resolution. Then Local Maxima algorithm implemented also in MATLAB is performed to extract the most relevant points of the Hough transform accumulator. In this implementation ten local maxima are extracted if they are not on the neighbourhood of 51×51 positions of other maximum. This allows to delete duplicate detections and collect enough lines to run the algorithm. Next, is extracted the image boarder points which correspond to line limit points used to represent the detected lines on the image. In this stage the lines are also classified as vertical or horizontal taking in account the θ value of Hough transform. Empirically, it was defined that vertical lines correspond to $\theta < 0$ and consequently horizontal to $\theta > 0$.

On image 4.13 it is possible to observe the results of the algorithm in different frames and that not the same lines of the field are detected in different moments.



(a) Result of the line detector on frame 450 of sequence nr.4

(b) Result of the line detector on frame 850 of sequence nr.4

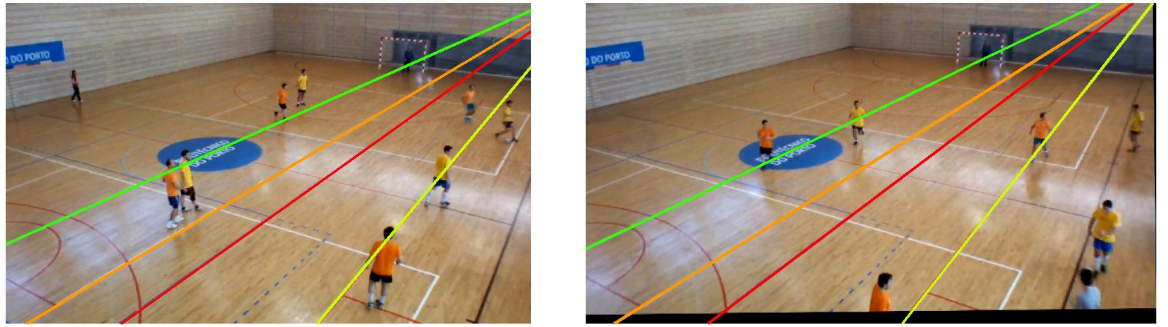
Figure 4.11: Two examples of line detection on different frames.

Since the line detector presented above performs with a detection rate not so favourable and there is also some false detection, is not possible to compare directly the results of the detection with the model. So the next step of the automatic calibration procedure is to match the lines detected into the model.

This matching problem will be modeled as an assignment problem and solved using Munkres algorithm [23] assuming that one detected line can be assigned to a single model line and a model line can be assigned to only one detection line. This method is performed minimizing the total cost of assignment. The cost of assigning a detected line to a model line will be calculated using the distance between their parameters in Hough Space:

$$cost_{ij} = \sqrt{(\theta_i - \theta_j)^2 + (\rho_i - \rho_j)^2} \quad (4.21)$$

This assignment is performed separately for vertical and horizontal lines. The use of the line parameters in Hough space was preferred since the cost calculation is simpler and more robust than in the cartesian line parameters $[m, b]$ case.



(a) Result of the line assignment on the model (b) Result of the line assignment on detected lines

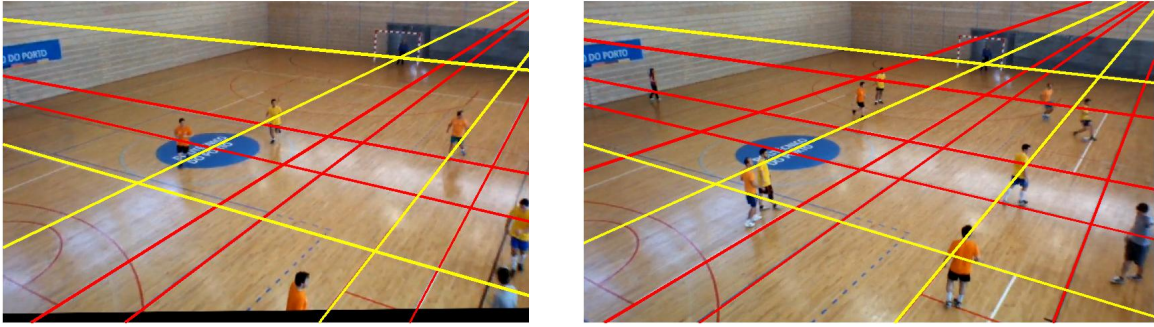
Figure 4.12: Example of line matching using an assignment problem. Horizontal lines detected (right) are matched to the lines on the virtual model (left). Each color represents an assignment

After all the horizontal and vertical detected lines on current frame are matched to the model the next step is to select the four pairs of corresponding points which will be used to calculate the homography. A good criterion to this selection is that the points shall be dispersed on the image. So horizontal and vertical lines will be sorted by their ρ parameter and allow to select the top, bottom, most-right and most-left line and consequently calculate their intersection points which will be the most dispersed point selection available.

From the intersection points of the selected lines both in image and in model, the homography from the current frame to the model will be calculated.

First step is to backproject the four intersection points to the world coordinates using the initial homography. Then the image-to-field homography of the current frame is calculated using the same method of the initialization and using the four points of the intersection of the selected lines and the four points projected on world coordinates using the model and homography initialized by the user.

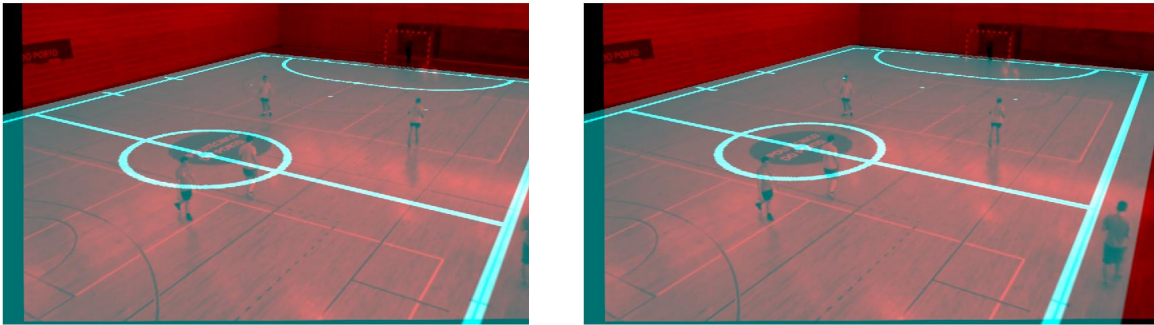
Figure 4.14 allows to observe the final result of the automatic calibration method. This method is very robust for the circumstances of this research in which we assume that the Drone is recording



(a) Detection of the top,bottom,most right and most left of the assigned lines

(b) Corresponding assigned model lines of top,bottom,most-right and most-left lines detected

Figure 4.13: Example of line selection for picking the best intersection points. Note that in left image there is a right line on the right of the selected as the most-right. That happens because that line is not in the model.



(a) Frame 850 of sequence nr.4 with the model projected using initial Homography

(b) Frame 850 of sequence nr.4 with the model projected using automatic calibration method.

Figure 4.14: Example of the final result of the automatic calibration method. On the left it is possible to observe the non calibrated projection of the field model. On the right the model is correctly projected using the calibration method presented above.

always the same area of the field with undesired camera motion even with a previous stabilization process that can eliminate high-frequency jitter. In this procedure we must notice that Hough Transform is very expensive computationally so this calibration step shall not be performed in every frame.

The camera calibration method was tested on sequences 1 and 4 and the the estimated calibration in each frame is compared to ground truth annotated data using the error equation presented on chapter 3. The method was performed with different correction rates, namely: $\frac{1}{5}, \frac{1}{50}, \frac{1}{100}$ corrections per frame. The error evolution will be also compared with the initial and static homography.

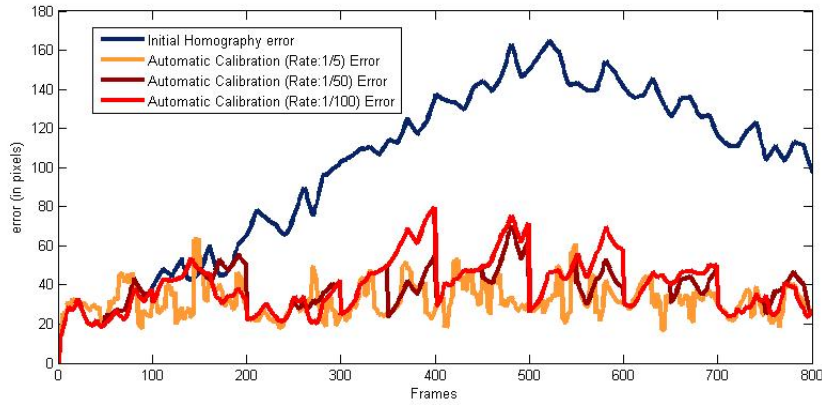


Figure 4.15: Error of camera calibration method with different correction rates on sequence 4.

Figure 4.15 shows the efficiency of the method under different correction rates. While initial homography starts to drift, the proposed method stabilize the error over time. It is also possible to observe that with the decrease of correction rate, maximum calibration error increases as expected.

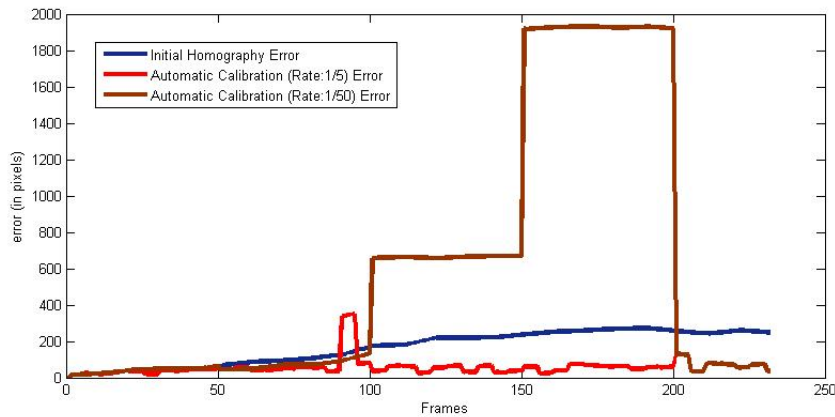


Figure 4.16: Error of camera calibration method with different correction rates on sequence 1.

On figure 4.16 it is possible to observe the influence of detection rate and position of the drone to have an efficient calibration. The sequence 1 is shot with low altitude and far from the covered half field (figure 4.17). This causes that almost all the lines detected are not well sparse on the field increasing the probability of bad lines matching and consequent calibration drift. Low calibration rates can also lead to loose the correct assignment of the lines. In this specific case the method performed with a rate of $\frac{1}{50}$ do not assign correctly the lines and consequently estimate the homography wrongly.

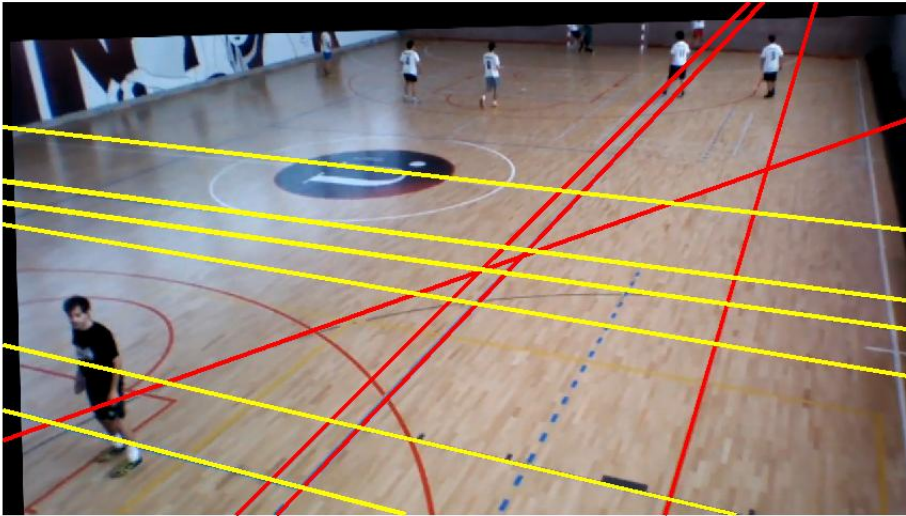


Figure 4.17: In sequence 1 almost all the detected lines are not in the expected half field increasing the probability of bad assignment and wrong calibration.

The proposed method works well on approximately static views but the use of Hough transformation to line detection is very expensive computationally so it shall not be performed in every frame. The results allowed to observe the importance of calibration rate on the efficiency of the method and how this parameter shall be chosen carefully. It is also important to the aimed half field cover almost all the image plane to lines be well detectable and sparse since final results will be strongly dependent on the process of line detection and matching.

4.5 Conclusions

In this chapter the first two stages of the system framework - video stabilization and camera calibration - were presented. For both stages a small review of the literature about the subjects was presented and also the methods developed with preliminary results. Video stabilization is required to compensate undesired drone's motion and eliminate high-frequency jitter. The method developed was based on FAST features matching on adjacent frames and affine transformation estimation with outliers elimination performed by a variant algorithm of RANSAC. This method performed well for compensating high-frequency motion, still not all the motion camera is compensated and the background will move smoothly over time. Compared to other methods, this approach is quite simple but it is suitable for this application since is assumed that drone must be hovering the same location and covering the same area of the field getting a more and less static view of the field. Another problem of this method is the decreasing of visible area over time, but that can be neglected since the sequences are relatively short (30s to 1min).

Since not all the motion is compensated and thinking of further applications of the system that won't use a static perspective of the field, automatic calibration is required. Camera calibration is

the process to relate image coordinates with the world. On the initialization the user is required to mark manually four pairs of corresponding points both in the image and in the model which will be used to directly estimate the homography between the image and the field plane. The creation of a virtual model of the field using all the lines on the floor of the indoor sports venue that will include the court's lines of many sports is also required. These lines will be posteriorly be used to compare with the lines automatically detected on the current frame using Hough transform. After assigned the detected lines to the model four pairs of points corresponding to the intersection of the lines are used to calculate the homography between the image and field plane. The method proposed was robust and suitable for the application despite its simplicity keeping the calibration of the camera during all the sequence. Calibration rate is an important parameter since Hough transform is computationally expensive it shall not be performed in every frames, on other hand a low correction rate can lead to loose of lines matching and consequently to loose the calibration.

Finally, is important to refer the influence of the correction of lens distortion on the rest of the system framework. The method for automatic can only be performed if the lines of the field are straight in the image. However, the reduction of the visible area is prejudicial for player detection and others stages of the process. In this project it will be used the calibrated images on the further steps of the framework but one can discuss if it is the best solution. Storing the transformations expressions in each step would turn possible to use original steps in some of the stages preserving useful information and also performing the necessary corrections.

Chapter 5

Player Detection

The detection of the players on the image is a critical stage of the project. Having found in literature several methods to perform players detection, these approaches differ from each others mainly due to the image acquisition architecture, number of players and the format and color of the playfield. In this chapter a method for player detection will be proposed and the different stages of the process discussed. In the end of this chapter the proposed method and also some preliminary results will also be presented.

5.1 Overview of Detection Methods

5.1.1 Basic Segmentation Methods

Image segmentation is the process capable to split the image in different regions with similar features. These regions usually correspond to different objects, shapes, colors or textures presented on the image. In the context of this project it is intended to segment the image corresponding to the player's location on the image and simultaneously deal with other objects as the lines of the field or the spectators. Most of these techniques are performed when the image acquisition system is composed by fixed cameras or the playfield is of the same color (i.e outdoor soccer grass field). A background model can be created from initial frames without presence of players and posteriorly be subtracted to the current frames [7]. Updating models using Gaussian Mixture models [4] or recursive algorithms with foreground and background pixels with weighted importance [5] allow a segmentation more immune to changes on the illumination or other external conditions. When non fixed cameras are used, another approach is required for background subtraction. Taking advantage of a fairly similar background color and of prior knowledge about it, is possible to create background pixels detectors. Using efficient models through color information and identifying pixels that are not similar to the background is possible to classify them as foreground region and consequently probable players' regions [12]. Using the histogram of the image is possible to detect the dominant color with a certain deviation [11] and posteriorly subtract those pixels from the image. Other more robust methods are needed when illumination changes over time or in

different field zones, in that situation histogram backprojection and update can detect changes on the background pixels colors [9].

This kind of methods are not suitable for our application since image acquisition is performed with a moving camera that would cause much noise on the foreground pixels detection and there is not a dominant color present on the field since indoor sports venues usually have the lines of many sports courts painted which would also create noise on the process of foreground pixels detection.

5.1.2 Sliding Window Detection Methods

Some methods use special features and classifiers previously trained to detect objects on image using shape or intensities information. In this section the HOG detector [43] and Viola and Jones method [19] will be reviewed.

5.1.2.1 Viola and Jones

Viola and Jones detector was one of the first and is still one of the most used method for object detection in real time. It is used in many applications such as face detection, pedestrian detection and also players detection on soccer games depending on the dataset used to train it.

This detector uses a search window with a pre-defined size where a set of Haar-like features are used to classify that region as object to detect or not. Haar-features calculate the sum of intensities of the image pixels within a rectangle of the image and then compare it to the sum of adjacent rectangle. These features are advantageous as they run in constant time and integral image representation can be used decreasing time of computation considerably. Otherwise a single feature cannot provide useful information and for instance in a 24×24 search window there are 162,336 possible features being impossible to evaluate them all taking in account real time constraints. So the big innovation of this method is its learning algorithm which uses a dataset of positive and negative image samples to train features and evaluate those that are stronger and important than other on the classification process by the AdaBoost method [44] where a set of "weak" features are combined to get a robust classifier. In the end, the classifiers are rearranged in a cascade in order of importance. In this procedure a feature is only evaluated if the previous ones were classified as positive.

This method is more suitable for detection based on pixel intensities, for instance faces static planar objects as road signs or logos. In case of objects with very distinctive colors as for example people with different kinds of clothes, the shape of the object can differentiate it better instead of intensities difference on the object region as proposed by Viola and Jones.

5.1.2.2 HOG Detector

Histogram of Oriented Gradients is a feature descriptor used to object detection assuming that objects' shape can be identified by its distribution of gradients. It is used to detect a diverse set of objects from people to cars or animals. The big assumption in this method is that an object shape

can be described by its gradient's intensities and orientation represented in a single features vector instead of other approaches that use a collection of features to represent an object.

The first step to find the HOG descriptor is to calculate image gradient on entire image using simple discrete derivative kernels on both image directions. The second stage is orientation binning that consists on the creation of small cell histograms of the gradients' orientation where each pixel contribution is weighted by its orientation magnitude or function of it. The next step is essential to the robustness of the general descriptor and to avoid the influence of illumination: the cell gradient's histograms are normalized splitting the descriptor in overlapped blocks used to normalize the cell histograms. Finally, the classifier must be trained using Support Vector Machine (SVM) a supervised learning model which will be used to calculate an optimal decision function based on a dataset of positive and negative samples. The output of this linear classifier will be a vector of features coefficients, in common sense, it will select the most important cells of the descriptor to classify a certain object, in this case as a person/non-person [43].

The detection is achieved by running the classifier in a sliding search window over the entire image. This method is very popular in human detection and there are available multiple templates for classifiers to people detection, namely one available in MATLAB for detection for people in upright position.

The two examples of complex object detectors presented above were chosen because of being widely used and both of the detector and classifier learning are implemented in most of computer vision software as MATLAB or OpenCV. HOG classifier is more suitable to people detection, since it is more immune to external conditions and also to the pose of the person.

5.1.3 Mean Shift and Camshift

Mean shift is an algorithm that iteratively moves a search window in the direction of its center of mass until it coincides with the geometric centroid. Despite its simplicity this is a method widely used for object tracking on the image sequences [45]. The calculation of the center of mass of the search window can be performed using any feature space as per example color space, object shape among others. For an initial estimated location x and its neighbourhood $N(x)$, for each x is possible to calculate the density mean using a kernel function $K(x_i - x)$:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x) x_i}{\sum_{x_i \in N(x)} K(x_i - x)} \quad (5.1)$$

Iteratively, x will move to $m(x)$ until $m(x) = x$ reaching the convergence. Multiple Kernels can be used to density mean calculation but usually it is usually expressed in function of $\|x\|^2$ [16]:

$$K(x) = k(\|x\|^2) \quad (5.2)$$

Where k is called the profile of K and must satisfy the following properties: be nonnegative, non increasing and limited.

In image tracking application k is usually a function of quantification in a given colorspace so that the search window will follow the region of the image that maximizes the similarity to the object color histogram.

Camshift [46] is a variant of mean shift that assumes that the object to track will not keep the same size on the image over the sequences. In mean shift algorithm the search window has always the same size which is not suitable for the last assumption. So Camshift uses an adaptive search window changing the size properly. On a first iteration it uses normal meanshift until it converges and then updates the size of the window and repeats the procedure until it reaches global size convergence.

These methods are useful since they do not assume any prior shape and several models can be used to calculate probability density for the kernel function. Otherwise in the case of fixed size search window its choice is not trivial and in Camshift window size easily expands without criteria.

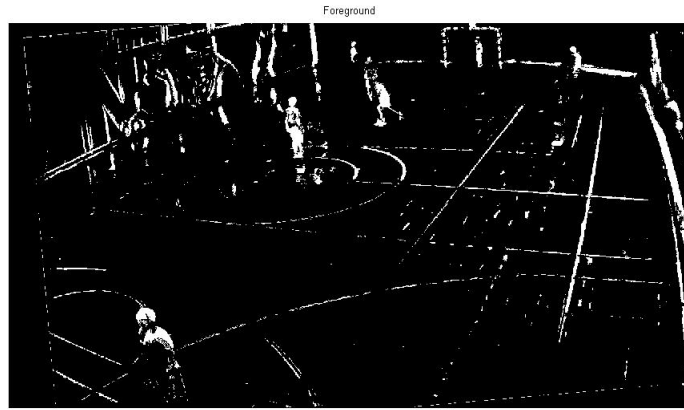
This solution is suitable for short term player tracking once properly initiated. But in long term it tends to fail since players appearance will change with the movement and multiple players with same equipment color will confuse the automatic tracker.

5.2 Proposed Method

In this section the methodology developed in the stages of the system framework for player detection, team identification and trajectories tracking will be presented. Some preliminary results and a brief discussion on the methods utilized will also be presented. The methodology is based on short-term tracking with mean shift algorithm corrected by detections of HOG classifier using generalized template for people detection for long term reliability and finally with histogram comparison in RGB channel for team identification and false positives handling.

Due to the camera motion even after compensated and to the existence of simultaneously multiple colored lines on the court basic segmentation methods based on background segmentation and subtraction are not suitable for this application and some preliminary results showed poor results with low detection rate and high number of false positive detections. In figure 5.1 it is possible to observe a demonstration of basic a method based on background modelization using Gaussian Mixture Model with automatic upgrading.

These results could be improved through the application of some constraints or other complementary algorithms but it was assumed that another methodology was required in this application. Sliding windows detectors are computationally efficient and presented robust results in different external conditions. In this application the detector utilized was the one based on HOG descriptor because of available templates for human detection and since it is based on the detection of object shape by its gradients is more robust to change in intensities information as for example person and equipment colors.



(a) Result of background subtraction



(b) Detections based on the basic segmentation method

Figure 5.1: Example of bad results from background subtraction using a Gaussian Mixture Model for its representation and corresponding detections. This method produced a low rate of true detections with an high percentage of false positives

5.2.1 HOG Detector Implementation

To detect the players on the image an implementation existing in MATLAB Computer Vision toolbox of a SVM classifier for HOG descriptors trained to detect people on upright position and based in Dalal et. al[43] it was used. First, it is important to refer that the template used lies on the detection of people standing statically or in smooth movement. However, in sports context players are many times running, tackling, occluding each others. These are propitious situations to fail the detection. The alternative in this case would be to train our own SVM classifier but for that proper annotation software would be necessary, a big variety of training sequences and samples would be necessary besides the time needed to perform, which would not be appropriate for the scope of this dissertation.

This classifier was trained with samples of people in upright position on different light conditions, in several different places to make the detection reliable. The images size was 96×48 pixels

this means the smallest search window will have that size. However, this implementation re-scales the search window iteratively with a user-defined increasing rate. In this application the default value of 1.05 performed properly. The biggest search window is also parametrized and it would be defined taking in account the samples collected on the initialization process. Another parameter is the window stride and it defines the spatial frequency of the search window on the image on both directions. A small value will lead to increase accuracy, otherwise the computation time will also increase. Since this project is not dealing with time constraints a small value was specified. For instance 4×4 pixels attain good results and smaller values did not improve the results.

Figure 5.2 is representative of some common outputs of the HOG detector in terms of the expected precision and recall. The output of the detector is the bounding box of the region containing a possible person. It is very accurate to detect players standing but it usually fails when they are running or shaking arms and legs (orange player). Other usual cases of false positives detections lie on vertical structures as goal post or in human body parts. Another common situation that is not represented on the figure 5.2 are occlusions, players partially occluded will lead almost always to miss the detection.

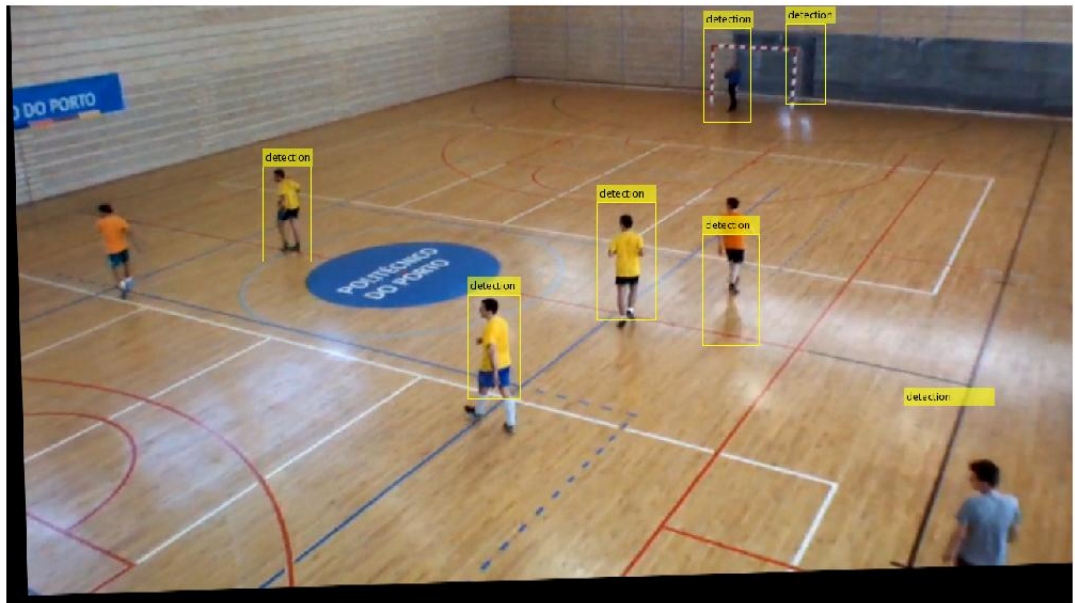


Figure 5.2: Representation of usual HOG detector results. Frame 450 of sequence nr.4

Preliminary results of the detection based on the available model for people in upright position are satisfactory but both precision and recall need to be refined to extract useful information of players positions and trajectories. Another stage of player detection is to identify the team by its equipment color and discard other people appearing on the image namely the referees, goalkeepers and spectators.

5.2.2 Team Identification and False Positive Handling

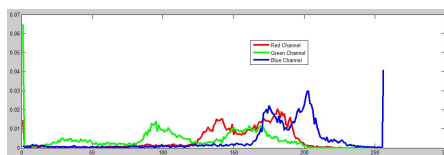
The results obtained from the HOG detector will be verified taking in account the appearance of the region detected represented by its color histogram in RGB colormap. It will allow to discard false positive detections and also group the detection in two groups representing both teams in the field.

Before the tracking process starts the user is asked for an initialization procedure in which initial player positions will be annotated and a set of samples of players appearances also collected. Annotation is carried out selecting the bounding box of the region containing the player and identifying his team. Then the color histograms of two regions are extracted: one for the entire bounding box and another for the region containing the t-shirt, since that team equipments usually have different colors over the different parts (t-shirt, shorts and socks). On the last step of the initialization procedure, the HOG detector is performed on a small number of frames and posteriorly asked to the user to identify positive and negative samples. Once more, two histograms are collected for each sample and grouped in one of the following groups: "Team A", "Team B" and "No Player". Each histogram is saved instead of averaging results since, specially in case of negative samples, histogram distributions are very different and information would be lost with that approach.

For each bounding box an histogram of pixels intensities on each channel of RGB colormap is created and grouped using 256 bins. The number of bins will affect both the accuracy of the model and the computation time. Since in this project time constraints are not the main criteria it was used this large number of bins. For instance, in the example given if a small number of bins were used "Yellow" and "Orange" team would be easily confused. Finally, all the histograms are normalized according to bounding box size.



(a) Example of Region Detected



(b) Histograms RGB channels of the region detected

Figure 5.3: Example of the histograms extracted from one output of the HOG detector

The classification method is based on the k-nearest neighbor classifier [47]. This is one of the most simple classifiers with robust results but without computational efficiency, since all the training data must be stored and compared at each time. The method is performed in two steps: first it is classified as "Player" or "Not Player" and then if first classification passes, it runs the second: "Team A" or "Team B". The method used is a distance classifier where the tested histogram h_i

will be compared with all the histograms h_j collected from the initialization process. The distance between each channel of two histograms is calculated using the Bhattacharyya method [48]:

$$d_{i,j} = \sqrt{1 - \frac{1}{\sqrt{h_i h_j N^2}} \sum \sqrt{h_i(n) h_j(n)}} \quad (5.3)$$

The result of this expression is a value between 0 and 1 where 0 means a perfect match and 1 a total mismatch between both histograms. The total distance between two image histograms is the mean of the distances of each channel and the value is then stored in an array depending on the class compared. The arrays are then sorted ascending by the value of the distances. Only the five shortest distances to each class array will be considered to classify the histogram. The score is then calculated as:

$$D_{i.classA} = \sum_{j=1}^5 d_{i,j} \quad , j \in classA_{sorted} \quad (5.4)$$

$$D_{i.classB} = \sum_{j=1}^5 d_{i,j} \quad , j \in classB_{sorted} \quad (5.5)$$

$$Class_i = \operatorname{argmin}(D_{i.classA}, D_{i.classB}) \quad (5.6)$$

$$Score_i = \frac{\min(D_{i.classA}, D_{i.classB})}{\max(D_{i.classA}, D_{i.classB})} \quad (5.7)$$

On figure 5.4 it is possible to observe some typical results for team identification and false positives deletion. The proposed method is simple and performed fairly well. However, the method is neither computationally efficient nor robust for hard examples of similar teams identification, like the example on image 5.4. More robust methods as Linear Regression, SVM or others could be developed at this stage of the process.

5.2.3 Mean Shift and Short term tracking

The step explained above allowed to increase the precision of the detection but on the other hand, the recall percentage decreased. With a low frequency rate of detection a complementary is necessary method to predict players' position on the field. Some works use a dynamic linear model for players trajectories [13, 21] to predict player's position when a new detection is unavailable. But when image perspective strongly differs from the field plane, dynamical models lose their accuracy leading to wrong predictions. Mean shift, as seen in the previous section, provides reliable prediction of an object position based on its appearance but it fails over time once that player's appearance and size change. At this stage, there are two sources of data available: one with an high precision but not always available and another reliable only for short time periods. The method

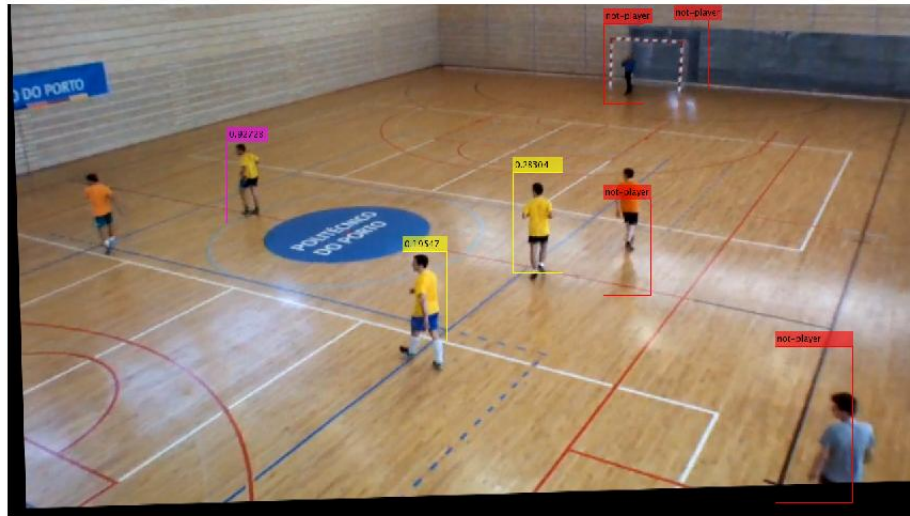


Figure 5.4: Example of the results of team identification and false positive handling. Red boxes represents false positives deleted. Yellow boxes for players of Team A and Orange ones for player of Team B. Must notice that a Team B player is wrongly classified as player of Team A. Frame 450 of sequence nr.4

will be performed by a Matlab implementation of Mean shift available. It performs the algorithm using for probability density calculation the Bhattacharyya distance on indexed colormap between the target and the model. The use of indexed colormap allow to use only one channel histogram easing the calculations.

In this methodology, the players' positions are initialized by user interaction by selecting the bounding box of the region containing the player and player's team is also identified, each initialization will create a new track. For each track a sub-window is selected corresponding to the t-shirt from which the indexed colormap histogram will be extracted. Shirts usually have a different color from the rest of the equipment, decreasing the target region of the algorithm will also decrease the range of indexed values to track increasing the robustness of the method. Finally, the new bounding box position can be estimated applying the same translation as the regions containing the shirts obtained from mean shift algorithm. In this method we will assume that players' size will remain equal between consecutive frames. On figure 5.5 it is possible to observe a short-term track based on mean shift for one player. The region used to extract the histogram and posteriorly perform the track is highlighted. These regions were chosen using the criterion previously presented.

Over time, due to change on the player appearance and size, mean shift tracking starts to drift and lose the target. The next stage of the process is to correct and reboot the tracking using the detections from the HOG detector, previously presented.

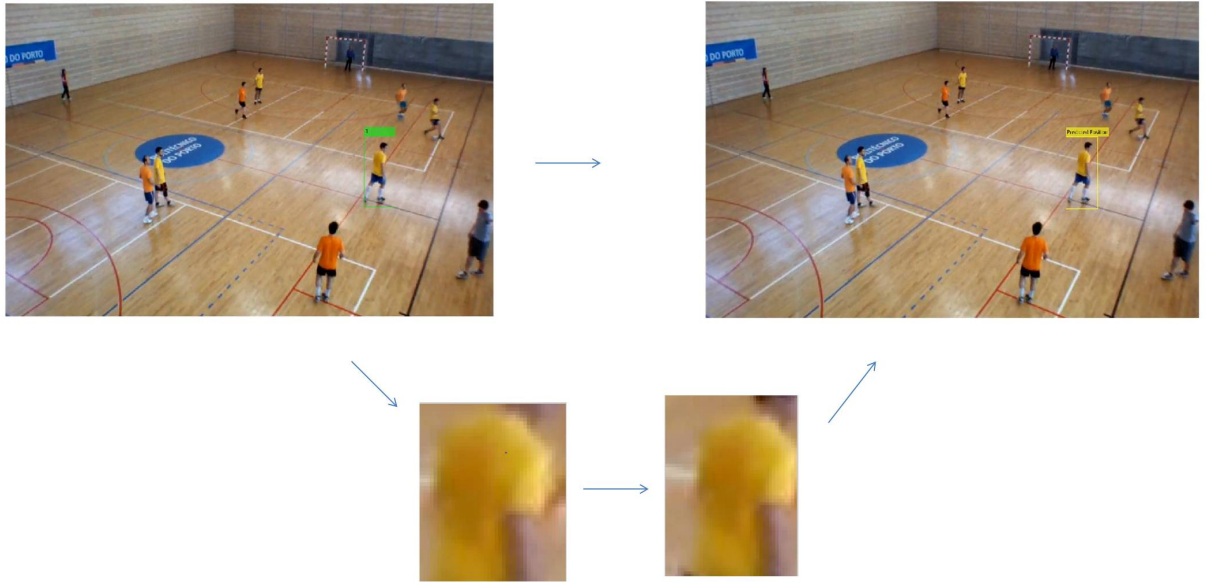


Figure 5.5: Representation of mean shift algorithm on player position prediction on two consecutive frames. In this case only the track for one player is represented. On the left image player initial position is represented by a green bounding box. The subregion of the box containing the shirt is represented on the next image. This region will be used by mean shift algorithm to predict the new location of the player, represented on the third image. Finally player position is predicted applying the same translation of the mean shift algorithm to the initial bounding box.

5.2.4 Assigning detections to tracks

To achieve long term reliable tracking it is necessary to fuse HOG detection information with the predictions of the mean shift algorithm. However, in every frame not all the players will be detected and false positives will occur. Another situation that the system will have to deal is players entering and leaving image plane, so that tracks must be added and deleted as players enter and leave the image. Because of these reasons, assigning detections to the corresponding tracks is not a trivial procedure. In this subsection the methodology of detections assigning based on Munkres algorithm will be presented.

Let us define the set of tracks $T = (t_1, t_2, \dots, t_i)$ and the set of new detections $D = (d_1, d_2, \dots, d_j)$ with $j \neq i$. Each track will contain the following data: bounding box; team; uncertainty. On the other hand each new detection will have data on: bounding box; team; score of team classification. On Munkres algorithm the assignment is performed so that it minimizes the total cost of assignment. In our method, the cost of assigning a detection d_j to a track t_i will depend on the distance between the coordinates of their bounding boxes but also team classification and the corresponding score. Taking in consideration team classification in the cost function will help to solve cases of occlusions from players of different teams and using the score is important not to exclude detections with a wrong team classification. Let the bounding box be defined as $boundingbox = [x, y, w, h]$ where (x, y) is the upper-left corner of the box and (w, h) are the width and height of the box. Let $bbox_i$ be the bounding box of track t_i and $bbox_j$ the bounding box of detection d_j on a given frame

of image sequence. The Euclidean distance between these two bounding boxes is defined by:

$$db_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (w_j - w_i)^2 + (h_j - h_i)^2} \quad (5.8)$$

We will define the cost of assigning a detection d_j to a track t_i as:

$$cost_{ij} = db_{ij} \cdot \frac{e^{score_i^k}}{e^{score_i}} \quad (5.9)$$

$$where, \quad k = 1 \quad if, team_j = team_i \quad (5.10)$$

This expression allows to assign detections with tracks from another team since it's score is near one and otherwise, excludes detections with a different team classification and with low score.

In Munkres algorithm it is also possible to define the cost of non assignment C_n so that it will find the assignment that minimizes the total cost and respecting $cost_{ij} < C_n$.

The tracks with assigned detection will update their bounding boxes for the one of the detection assigned and will reinitialize the histogram model for the mean shift algorithm. Since there are players entering and leaving the image planes the system must automatically add and delete tracks.

5.2.4.1 Adding and Deleting Tracks

Detecting players leaving and entering the image is not a trivial task since the presence of false positives and missed detection can lead to misclassification of new or lost tracks. To solve this it is necessary to assume some rules based on empirical knowledge.

Tracks will be assumed lost taking in consideration the number of consecutive frames without any detection assigned and also the current localization of the bounding box since it is more probable that a track is lost when players are near the image border.

In this system it will be assumed that there are only four players from each team on the field so a new track will be created only when one player from one team is missing. The new tracks will be creating based on unassigned detections. A provisory track is created for each unassigned detection since its score on team classification is lower than 0.5. This is an important rule to avoid new tracks to be wrongly classified. A new track is created when a provisory track has more than R assigned detections. Where R is a pre-defined fixed number, if this number is low false detections can lead to wrong tracks and if it is high there is a possibility of players never being tracked.

On figure 5.6 is possible to observe an example of the accuracy of the proposed method. As seen previously on figure 5.2, resulting from the detection and team classification one player was not detected, one was wrongly classified and, finally, a bounding box location was not well located.

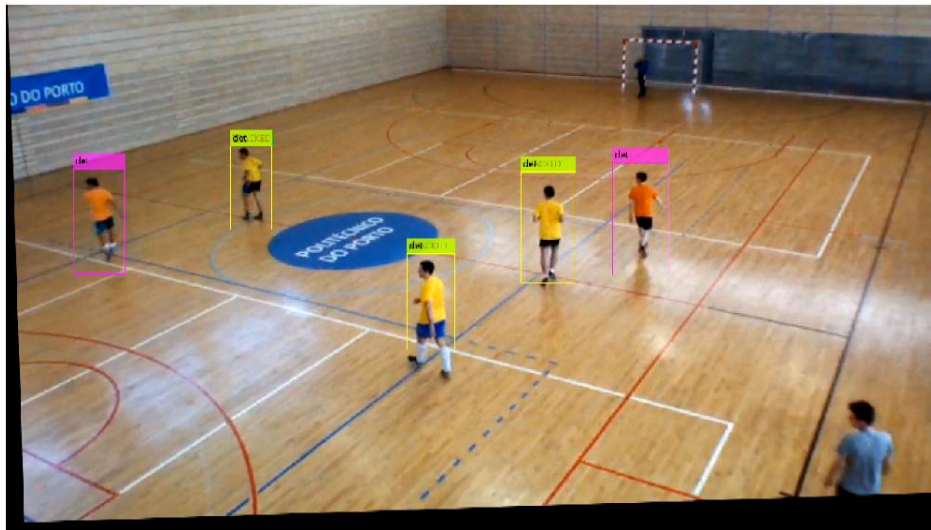


Figure 5.6: Example of final output of player detection stage. Player position are represented by its bounding box and the team by the color of the box.

5.2.5 Results

The different stages were evaluated using the criteria presented on chapter 3. The sequences tested were the sequences 3 and 4. These two sequences deal with complex situations such as bad illumination, overcrowded scenes, similar team equipments, among others.

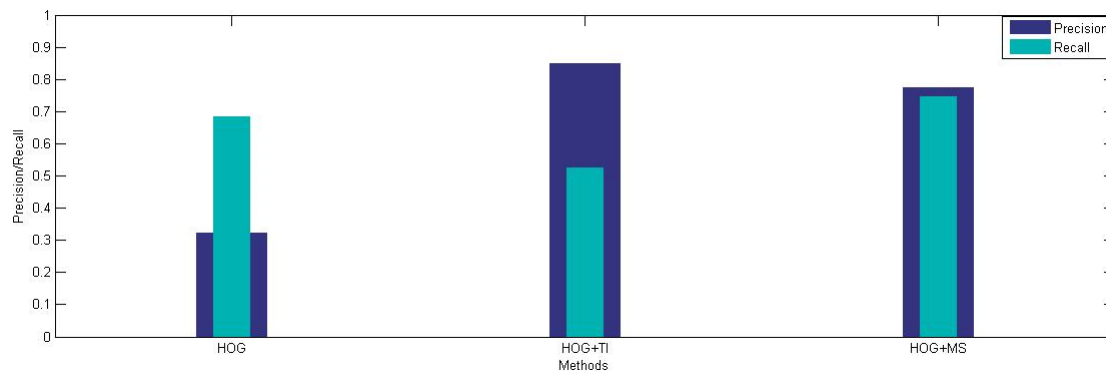


Figure 5.7: Evolution of Player detection results through the different stages of the method on sequence 3.

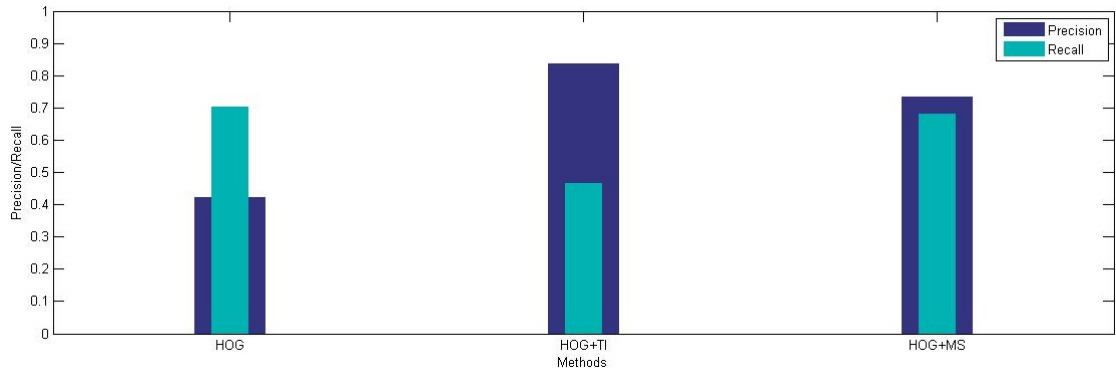


Figure 5.8: Evolution of Player detection results through the different stages of the method on sequence 4.

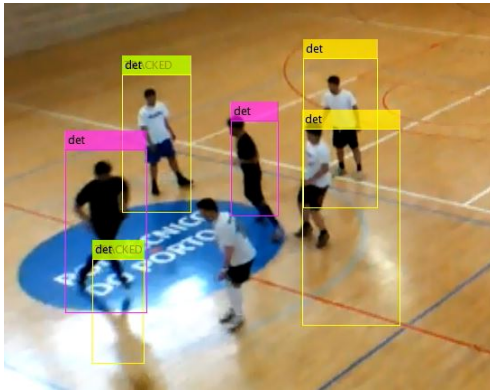
On figures 5.7 and 5.8 the results of player detection on sequence 3 and 4 on three different stages are presented. It is presented the raw results from HOG detection (HOG), the results of the detector after applying team identification and false positive handling (HOG+TI) and finally the final results of the complete method based on HOG detection and mean shift tracking (HOG+MS). The precision of the raw detector is very low producing too much false positives. Using information of the players appearance is possible to delete a big part of these bad detections. The results with false positive handling showed a notorious increase on the precision but on the other hand the recall decreased. Finally with mean short term tracking is possible to estimate players position in the case of missing detections increasing the recall. Otherwise, because of tracking and the difficulty to deal with new and lost tracks the influence of false positives will be higher and that is the reason why precision falls.

5.2.6 Failure Situations

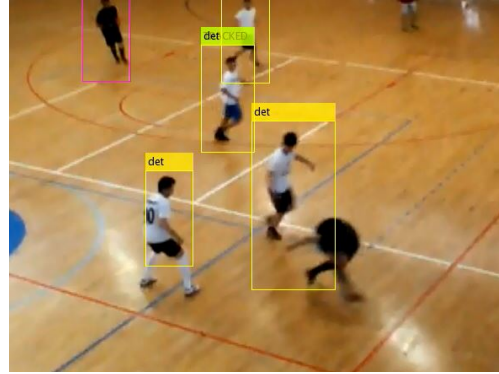
Preliminary results showed a precision and recall of the proposed method around the 75%. The use of an HOG detector not dedicated for this purpose, overcrowded scenes and bad illumination makes player's detection a hard task.

On figure 5.9a is possible to observe an overcrowded scene where players from different and same team are closer to each other inside a small region of the field. Both detector and short term track fails on this situation. For the first player occluded changes its shape on the image and detector will not recognize it. Mean shift can also fail since it is highly probable to a player from the same team be confused by the tracker.

Figure 5.9b illustrates one common and quite undesired situation for player's detection. New players entering the image are difficult to track since there is no prediction of when and where a new track must be created. Because of this the algorithm has to be prepared to accept new tracks at any time. This will increase the negative effect of false positives because they can be confused as new players.



(a) Overcrowded Scene



(b) Player entering the image plane



(c) Confusion with equipment color

Figure 5.9: Examples of usual failure situations.

Finally, figure 5.9c shows a situation where team classification fails because orange and yellow are similar colors on RGB colormap turning its classification not accurate.

5.3 Conclusion

Using the mean shift algorithm and keeping a short term tracking of players, missed detections and wrong team classification will be neglected. However, this methodology has some difficulties to deal with new and lost tracks and with detection of players near the goal line where they occupy a small region and colors models do not fit properly. Team classification uses a very simple approach not being very robust in cases of teams using similar colors (as orange and yellow). A more robust solution based on linear regression methods could be used taking advantage of more characteristic features. Mean shift algorithm to perform short term track of player while detections are unavailable is an intuitive and robust idea but other information could be included to improve the accuracy of the method, as: Optical Flow estimator as KLT or Horn's method. Features tracking or local segmentation would be interesting solutions to integrate in method taking in account this

stage of the methodology. The HOG detector was chosen to correct short-term tracking and also detect new players on the image, this is a very robust method to detect an object shape on the image being immune to lighting or even the colors of the object. However, the implementation used was trained to detect people in upright position which was not the ideal since players are always running, tackling as many other different poses. A better precision and recall can be achieved with training of a dedicated classifier learned with samples of players in different positions and positions. Finally, linear or non linear filtering can be performed to smooth the results and increase the precision and recall of players' tracking.

Chapter 6

High Level Interpretation

In this chapter the methods and results for high-level data interpretation will be presented. These will use the results of automatic player detection and camera calibration methods described on the previous chapters. The methods presented in this chapter are difficult to be evaluated objectively so that only qualitative criteria will be used. The methods were developed considering just the author's common knowledge about the game rules and dynamics. Since images from the collected sequences only cover one half of the field and the detection methods only regard team identification and not individual recognition the information intended to extract is related to general **teams attitude**, **occupation zones** and **defensive team tactics**.

6.1 Occupation Map

Occupation or Heat maps are an usual method to evaluate teams or player performance during the game [13]. It shows how players occupied the field and can give important clues about teams strategies and performance. In this process it will be used the position of the players mapped on the world coordinates, then the field model is divided in a grid of 10×10 pixels in which it will be projected the actual position of the players. Each grid cell will accumulate the number of players there located and then it will use a spatial histogram to show the most occupied zones of the field for each team.

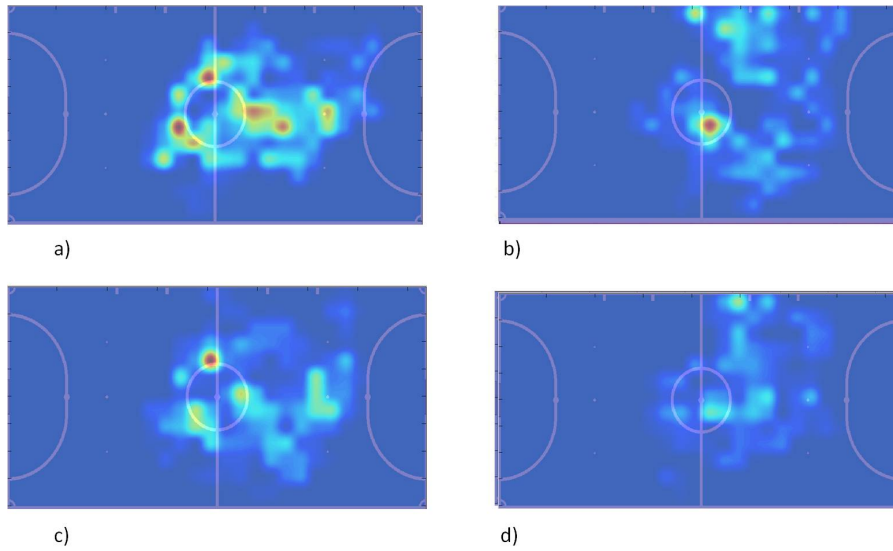


Figure 6.1: Occupation map of player on sequence 3. a) and b) relates do information extracted from ground-truth data for each one of the teams, and c) and d) the proposed detection and calibration methods.

In figure 6.1 is possible to observe the results of the occupation map. a) and b) illustrate occupation from players of team 1 and 2 in the sequence 3, inferred from ground data annotation. c) and d) are the analogous for the automatic player detection and camera calibration methods developed in this project. In this sequence the defending team (team 1) were more compacted on the field while the attacking one (team 2) were spread on the field exploring the sidelines.

The similarities between the two results are notorious but it is also possible to observe the influence of precision and recall of players' detection method.

6.2 Team Attitude

Team attitude or offensive/defensive trends can illustrate which team is being more dangerous or more close to score a goal. Although football can be very unpredictable, it is possible infer which team is being more aggressive and closer to the goal from the position of the players on the field. In this research will not be possible to have the positions from all the players in each instant so for this method we propose to deduct team attitude from partial information on players' positions. This means that with just only one player of each team is possible to predict an offensive/defensive trend.

We will assume that in each frame will be available the position (x_i, y_i) on the field of one to four players from each team and is known *a priori* which one is attacking and defending. For each team the offensive trend is calculated taking in account the relative position of the rearmost player on the ground to the midfield line and the relative position of the most advanced player to the opponent goal line. The equation that give the offensive trend to a team is:

$$Off_{trend} = \frac{x_{back}}{x_{midfield}} + \frac{x_{front}}{x_{goal}} \quad (6.1)$$

where x_{back} is the x coordinate from the rearmost player and x_{front} is the analogous for the most advanced player.

After each team trend is calculated they are normalized such that:

$$Off_{trend_1} + Off_{trend_2} = 1 \quad (6.2)$$

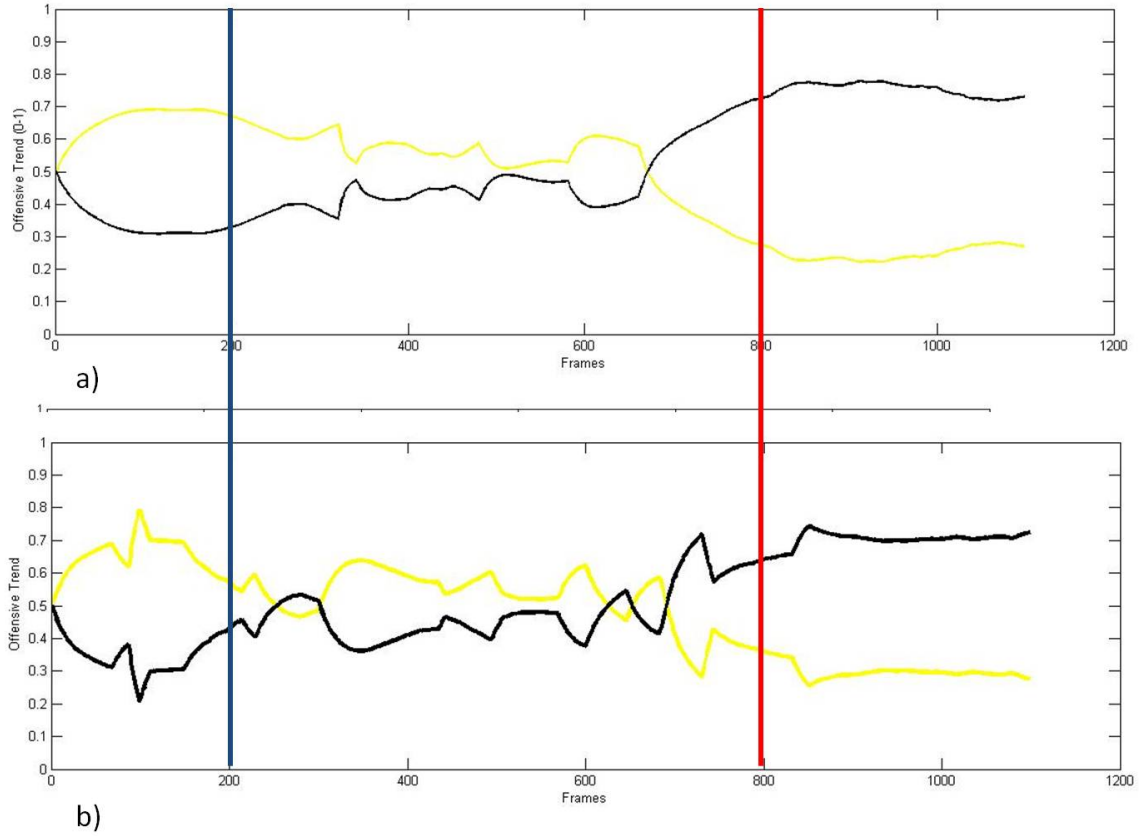


Figure 6.2: Result of the method to extract information on teams attitude and offensive trend. a) Illustrate the result for ground truth data and b) with the proposed detection and calibration methods. At blue and red are highlighted two frames which will be shown on figure 6.3

Figure 6.2 illustrates the results of offensive trend of team 1 and team 2 generated both from the ground truth data and the results of automatic player detection with short term tracking. In the beginning of the sequence is where the recall and precision are lowest and it reflects on the result of offensive trend profile. Figure 6.3 shows two examples of the results used to deduct the offensive trend. In the first case is possible to observe that white team even without the control

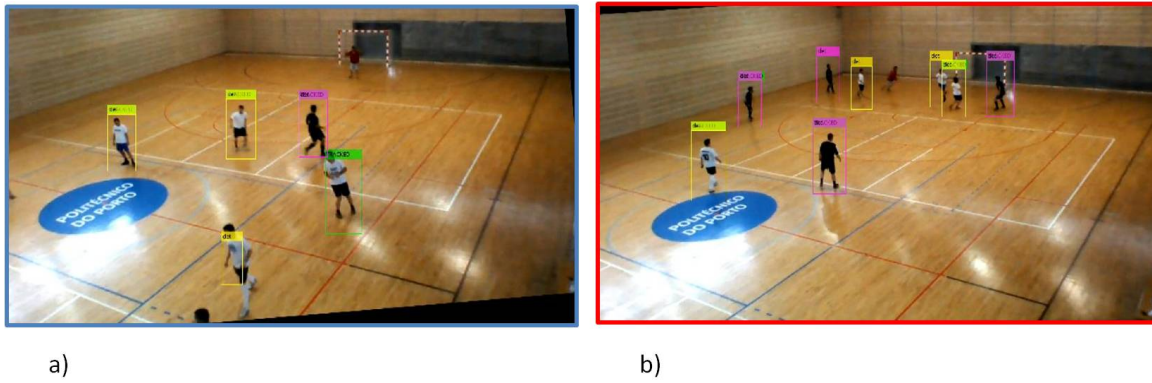


Figure 6.3: Example of two results of player detection on sequence 3: a) is the Frame 200 of sequence 3. b) is the Frame 800 of sequence 3.

of the ball is far from their goal line and that is the reason why white team has a bigger offensive trend even without being with the possession of the ball. In the second case all the players of black team are in the opposite midfield increasing their offensive trend.

This method lies only on partial information of players' position. To achieve more robust estimation of teams attitude during the game it would be necessary also include information about ball possession.

6.3 Team Tactics

Since only one half of the field is being covered by drone's camera only the defending team is probable to have all the players appearing in the image. So it is possible to make an analysis to team tactical behaviour and its evolution over time.

In indoor soccer there are two main defensive formations: "2-2" is based on two front player making high pressure and two back players. Other usual formation is "1-2-1" where only one player is making high pressure and a back player assumes most of the defensive tasks.

Our method intends to detect when a team is defending using one of these two formations using the relative position of its players. The approach used is quite simple and it is based on the spatial distribution relative to the most front and most rear players. A 3-bin histogram of players x coordinates is created considering the distance to the most rear and most front players. Then the histograms are compared to the model, for instance: if histogram is $[1, 2, 1]$ we will assume that the formation at that instant is the "1-2-1", if the histogram is $[2 - 0 - 2]$ the formation "2-2" is assumed. Finally each formation counter is accumulated and normalized being possible to observe its evolution over time.

Figure 6.4 represents the evolution of the utilization of each one of the formations during the game. These results illustrate the relative utilization of a defensive formation in smaller periods of time (in this case around 100 frames). For this sequence is possible to observe that in the beginning team 1 was using more the "1-2-1" formation and then changed to "2-2". Figure 6.5 illustrates two examples of different results. In the figure 6.5a is possible to observe a correct detection of "1-2-1"

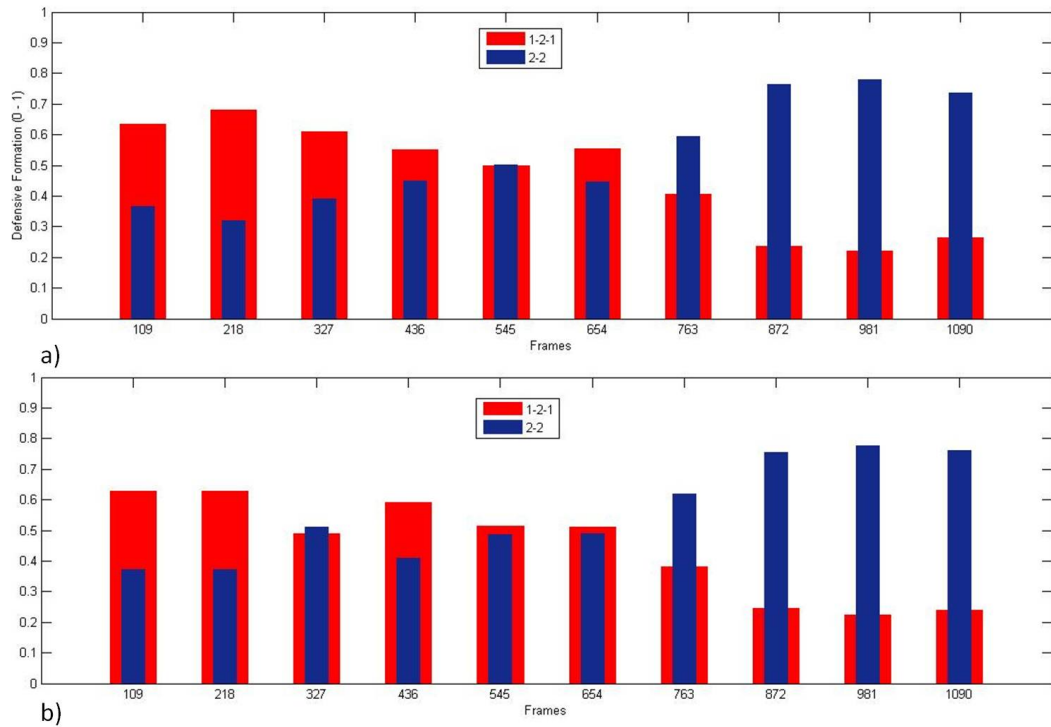
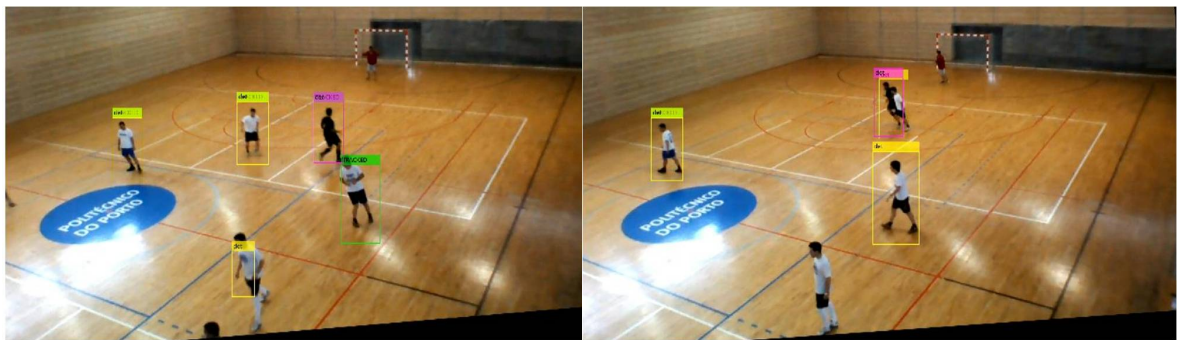


Figure 6.4: Evolution of the tactics counter for the sequence 3. a) Results extracted from ground truth data. b) Results from the proposed method.

occurrence. However, in the figure 6.5b because one of the players is not detected, the system does not detect the occurrence of a "1-2-1" formation.



(a) Frame 200 of sequence nr.3 with the results of player detection (b) Frame 250 of sequence nr.3 with the results of player detection

Figure 6.5: Two examples of how player detection results will influence the results of the tactic analysis. If in the first image is possible to see that "1-2-1" is the formation used, in the second image the system cannot detect because one player is not being detected.

The method presented is very simple and uses just simple relations between players' positions. A deeply analysis on the subject could be performed including more types of formations and more data as distance between players and also the interaction of the opponent team.

6.4 Conclusions

From partial information on player's positions is possible to extract yet some useful information about collective performance. The methods presented previously should give a general information on game and teams attitudes with low precise results from players positions. The results showed the similarities between the data inferred from ground-truth annotation and the results of the methods presented on chapter 4 and 5. However, it is also possible to observe the influence of missed detections and false positives.

To achieve more precise and reliable information on the game is required data with more precision and recall on players' positions.

The methods proposed for high level interpretation were evaluated qualitatively and most of the results were suitable for what was happening during the games of the corresponding sequences. It is important to remember that the methods proposed rely on common knowledge about the game. For more robust interpretation it would be required the expert knowledge on the construction of the methods and also more accurate low level data.

Chapter 7

Conclusions and Future Work

In this project a new approach to capture images in indoor sports venues which will be used to extract valuable information about collective performance was presented. Unmanned Air Vehicles as Quadcopters allow a cheap, portable, flexible and reliable platform to acquire images from indoor sports events, in the particular case of this research, indoor soccer. However, due to their own dynamics and also external factors it is impossible to avoid the drone's undesired motion which will result in a series of problems not typically found neither in the literature nor in the commercial solutions available on the market. In this work, a framework was set to extract useful and reliable information from indoor soccer games composed by different stages:

- Video Stabilization was required to maintain spatial coherence of pixels intensities despite the drone's motion. Using FAST features and RANSAC matching between adjacent frames is possible to estimate the inter-frame motion and consequently compensate it. The method proposed relied on its simplicity and efficiency on the test sequences. It can deal with the high frequency jittering of the camera but over time error is being accumulated and not all the movement is compensated. If longer sequences were tested or if it was intended camera motion to cover all the action of the game, more complex methods for stabilization would be required.
- Camera calibration is an essential stage of computer vision systems, in this project it is essential to map the position of the players in the field from their coordinates on the image. Since camera movement is not totally compensated is necessary an automatic and dynamic method for calibration. In this project is proposed a simple method based on detection of the lines marked on indoor sports venues and the posterior match with the lines of the virtual model created manually. The results proved that calibration does not drift. Correction rate is an important parameter to be set. If it is high, it will demand huge computational power and if it is low lines matching can fail and consequently the calibration too.
- Since most of the common methods to player detection are not suitable to this project due to nature of the image acquisition system a methodology based on HOG people detector with short term position estimation with mean shift tracking is proposed. The detection is based

on HOG descriptor and a classifier trained with a dataset of people on upright position. This detector has low precision to detect players in sports scenes. False positive handling and team identification was carried using histogram comparison in the RGB colormap and with a classifier based on k-Nearest Neighbour. This is presented as an simple and robust solution for histogram comparison but not efficient computationally. After this stage precision increased notoriously, on the other hand recall decreased on the same proportion. To estimate players' position while HOG detections are not available it is performed mean shift tracking which will find on the posterior frame the location of the image that maximizes the similarity with the current appearance. Final results presented a precision and recall rounding the 75%. The algorithm shown difficulties to deal with players entering and leaving the image since is there non prediction of where and when a new track must be created. This was the main cause to precision decrease from the HOG detection with false positive handling.

- Finally some methods are proposed to extract high level information from the data corresponding to players' positions on the field. These methods were based on the common knowledge of the authors about the game and evaluated considering only subjective criteria. Even with a not totally precise low-level information it was possible to infer some high level interpretation related to field occupancy, offensive trends and defensive tactics.

Considering the final results presented it is possible to assume that the proposed goals for this work were achieved despite most of the methods presented can be upgraded and refined to achieve most accurate results mainly on players' detection and tracking. In order to extract robust and truly useful information, drone's camera has to be able to cover the entire field. This can be achieved using multiple drones or with an automatic flight control such that it could follow game action based on ball or players' position.

7.1 Future Work

Among the main goals of this research there was the creation of a framework identifying some of the main problems and stages of an automatic vision system for sports analysis. A set of methods to the different stages was presented achieving some positive preliminary results. However, is possible to refine the results and upgrade the functionalities. Some of the future work should include:

- Video Stabilization capable to deal with intentional camera movement and without decreasing visible area over time.
- More robust camera calibration method without the need for Hough transform that is very expensive computationally
- Creation of a dedicated classifier based on HOG descriptor for players' detection. By collecting a large set of positive and negative samples is possible to develop a more precise detector turning detection easier.

- Results of detection must be refine using linear or non liner filtering as Kalman or Particle Filter.
- Finally, real time constraints must be applied and the methods shall be developed in a more computationally efficient platform such as in C++ with OpenCV. MATLAB is a flexible and powerful tool to image processing but it is not efficient to process video making the testing of sequences a long time consuming task.

References

- [1] Adam Herout and Markéta Dubská. *Real-time Detection of Lines and Grids: By PClines and Other Approaches*. Springer, 2013.
- [2] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [3] Catarina B. Santiago, Armando Sousa, Maria Luísa Estriga, Luís Paulo Reis, and Martin Lames. Survey on team tracking techniques applied to sports. In *AIS'10*, pages 1–6, 2010.
- [4] Pascual J. Figueroa, Neucimar J. Leite, and Ricardo M.L. Barros. Tracking soccer players aiming their kinematical motion analysis. *Computer Vision and Image Understanding*, 101(2):122 – 135, 2006. URL: <http://www.sciencedirect.com/science/article/pii/S1077314205001293>, doi:<http://dx.doi.org/10.1016/j.cviu.2005.07.006>.
- [5] Jinchang Ren, James Orwell, Graeme A. Jones, and Ming Xu. Tracking the soccer ball using multiple fixed cameras. *Computer Vision and Image Understanding*, 113(5):633 – 642, 2009. <ce:title>Computer Vision Based Analysis in Sport Environments</ce:title>. URL: <http://www.sciencedirect.com/science/article/pii/S107731420800043X>, doi:<http://dx.doi.org/10.1016/j.cviu.2008.01.007>.
- [6] Kyuhyoung Choi and Yongduek Seo. Automatic initialization for 3d soccer player tracking. *Pattern Recognition Letters*, 32(9):1274 – 1282, 2011. URL: <http://www.sciencedirect.com/science/article/pii/S0167865511000742>, doi: <http://dx.doi.org/10.1016/j.patrec.2011.03.009>.
- [7] Sachiko Iwase and Hideo Saito. Parallel tracking of all soccer players by integrating detected positions in multiple view images. In *IEEE International Conference on Pattern Recognition*, pages 751–754. IEEE Computer Society, 2004.
- [8] Wei-Lwun Lu, Kenji Okuma, and James J. Little. Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *Image and Vision Computing*, 27(1–2):189 – 205, 2009. <ce:title>Canadian Robotic Vision 2005 and 2006</ce:title>. URL: <http://www.sciencedirect.com/science/article/pii/S0262885608000462>, doi:<http://dx.doi.org/10.1016/j.imavis.2008.02.008>.
- [9] A. Dearden, O. Grau, and Y. Demiris. Tracking football player movement from a single moving camera using particle filters. *IET Conference Proceedings*, pages 29–37(8), January 2006. URL: http://digital-library.theiet.org/content/conferences/10.1049/cp_20061968.

- [10] A. Ekin, A.M. Tekalp, and R. Mehrotra. Automatic soccer video analysis and summarization. *Image Processing, IEEE Transactions on*, 12(7):796–807, July 2003. doi:[10.1109/TIP.2003.812758](https://doi.org/10.1109/TIP.2003.812758).
- [11] Seyed Hossein Khatoonabadi and Mohammad Rahmati. Automatic soccer players tracking in goal scenes by camera motion elimination. *Image and Vision Computing*, 27(4):469 – 479, 2009. URL: <http://www.sciencedirect.com/science/article/pii/S0262885608001455>, doi:<http://dx.doi.org/10.1016/j.imavis.2008.06.015>.
- [12] Suat Gedikli, Jan B, Nico V. Hoyningen-huene, Bernhard Kirchlechner, and Michael Beetz. An adaptive vision system for tracking soccer players from variable camera settings.
- [13] Wei-Lwun Lu, J.-A Ting, J. J. Little, and K. P. Murphy. Learning to track and identify players from broadcast sports videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1704–1716, 2013. doi:<http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.242>.
- [14] Jia Liu, Xiaofeng Tong, Wenlong Li, Tao Wang, Yimin, and Hongqi Wang. Automatic player detection, labeling and tracking in broadcast soccer video. *Pattern Recognition Letters*, 30(2):103 – 113, 2009. Video-based Object and Event Analysis. URL: <http://www.sciencedirect.com/science/article/pii/S0167865508000627>, doi: <http://dx.doi.org/10.1016/j.patrec.2008.02.011>.
- [15] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005. URL: <http://dx.doi.org/10.1023/B%3AVISI.0000045324.43199.43>, doi:10.1023/B:VISI.0000045324.43199.43.
- [16] Leow Wee Kheng. Mean shift tracking. Technical report, Technical report, School of Computing, National University of Singapore, 2011.
- [17] Greg Welch and Gary Bishop. An introduction to the kalman filter, 1995.
- [18] Arnaud Doucet and Adam M. Johansen. A tutorial on particle filtering and smoothing: fifteen years later, 2011.
- [19] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511–I–518 vol.1, 2001. doi: [10.1109/CVPR.2001.990517](https://doi.org/10.1109/CVPR.2001.990517).
- [20] A multiple camera methodology for automatic localization and tracking of futsal players. *Pattern Recognition Letters*, 39(0):21 – 30, 2014. <ce:title id=.
- [21] C.B. Santiago, A. Sousa, and L.P. Reis. Vision system for tracking handball players using fuzzy color processing. *Machine Vision and Applications*, 24(5):1055–1074, 2013. cited By (since 1996)0. URL: <http://www.scopus.com/inward/record.url?eid=2-s2.0-84879553147&partnerID=40&md5=57c4539872845367ab6fec24b5c179b>.

- [22] Paul Pounds, Robert Mahony, Joel Gresham, Peter Corke, and Jonathan Roberts. Towards dynamically-favourable quad-rotor aerial robots. In Nick Barnes and David Austin, editors, *Australasian Conference on Robotics and Automation 2004 ACRA2004*, Australian National University Canberra, December 2004. Australian Robotics & Automation Association. Proceedings of ACRA available on CD-ROM, at a cost of Aus\$200 each. URL: <http://eprints.qut.edu.au/33833/>.
- [23] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial & Applied Mathematics*, 5(1):32–38, 1957.
- [24] C. Morimoto and R. Chellappa. Evaluation of image stabilization algorithms. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 5, pages 2789–2792 vol.5, May 1998. doi:10.1109/ICASSP.1998.678102.
- [25] Yasuyuki Matsushita, Eyal Ofek, Xiaoou Tang, and Heung yeung Shum. Full-frame video stabilization. In *In Proc. Computer Vision and Pattern Recognition*, pages 50–57, 2005.
- [26] Richard Szeliski. Image alignment and stitching: A tutorial. Technical report, MSR-TR-2004-92, Microsoft Research, 2004, 2005.
- [27] Tinne Tuytelaars and Krystian Mikolajczyk. *Local Invariant Feature Detectors: A Survey*. Now Publishers Inc., Hanover, MA, USA, 2008.
- [28] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.
- [29] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *In European Conference on Computer Vision*, pages 430–443, 2006.
- [30] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [31] Andrey Litvin, Janusz Konrad, and William C Karl. Probabilistic video stabilization using kalman filtering and mosaicing. In *Electronic Imaging 2003*, pages 663–674. International Society for Optics and Photonics, 2003.
- [32] Ken-Yi Lee, Yung-Yu Chuang, Bing-Yu Chen, and Ming Ouhyoung. Video stabilization using robust feature trajectories. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1397–1404, Sept 2009. doi:10.1109/ICCV.2009.5459297.
- [33] Rong Hu, Rongjie Shi, I fan Shen, and Wenbin Chen. Video stabilization using scale-invariant features. In *Information Visualization, 2007. IV '07. 11th International Conference*, pages 871–877, July 2007. doi:10.1109/IV.2007.119.
- [34] Zhengyou Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330–1334, Nov 2000. doi:10.1109/34.888718.
- [35] Roger Y Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, 1987.

- [36] David G. Lowe Kenji Okuma, James J. Little. Automatic rectification of long image sequences.
- [37] D. Farin, J. Han, and P.H.N. de With. Fast camera calibration for the analysis of sport sequences. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 4 pp.–, July 2005. doi:10.1109/ICME.2005.1521465.
- [38] Jean bernard Hayet, Justus H. Piater, and Jacques G. Verly. Incremental rectification of sports fields in video streams with application to soccer. In *in Proc. of the Advanced Concepts in Intelligent Vision Systems (ACIVS'04, 2004*.
- [39] Jean bernard Hayet, Justus H. Piater, and Jacques G. Verly. Fast 2d model-to-image registration using vanishing points for sports video analysis. In *In: ICIP 2005. Proc. of IEEE Int. Conf. on Image Processing*, pages 417–420, 2005.
- [40] Paul VC Hough. Method and means for recognizing complex patterns, December 18 1962. US Patent 3,069,654.
- [41] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, January 1972. URL: <http://doi.acm.org/10.1145/361237.361242>, doi:10.1145/361237.361242.
- [42] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
- [43] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [44] Yoav Freund. An adaptive version of the boost by majority algorithm. *Machine learning*, 43(3):293–318, 2001.
- [45] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 142–149. IEEE, 2000.
- [46] G.R. Bradski. Real time face and object tracking as a component of a perceptual user interface. In *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on*, pages 214–219, Oct 1998. doi:10.1109/ACV.1998.732882.
- [47] Sahibsingh A. Dudani. The distance-weighted k-nearest-neighbor rule. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-6(4):325–327, April 1976. doi:10.1109/TSMC.1976.5408784.
- [48] Euisun Choi and Chulhee Lee. Feature extraction based on the bhattacharyya distance. *Pattern Recognition*, 36(8):1703 – 1709, 2003. URL: <http://www.sciencedirect.com/science/article/pii/S0031320303000359>, doi:[http://dx.doi.org/10.1016/S0031-3203\(03\)00035-9](http://dx.doi.org/10.1016/S0031-3203(03)00035-9).