**Faculdade de Engenharia da Universidade do Porto**

# Image Analysis of Sports Events

Catarina Maria Brito de Noronha Santiago

Dissertation developed under the
Integrated Master in Electrotechnical and Computer Engineering
Major in Automation

Supervisor: Prof. Dr. Armando Jorge Sousa
Co-Supervisor: Prof. Dr. Luís Paulo Reis

July 2009

**U. PORTO**

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES 2008/2009

A Dissertação intitulada

"IMAGE ANALYSIS OS SPORTS EVENTS"

foi aprovada em provas realizadas em 15/Julho/2009

**o júri**

Anibal Castilho Coimbra de Matos

Presidente Professor Doutor Anibal Castilho Coimbra de Matos
Professor Associado do Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

António Fernando Macedo Ribeiro

Professor Doutor António Fernando Macedo Ribeiro
Professor Associado do Departamento de Electrónica Industrial da Universidade do Minho

Armando Jorge Miranda de Sousa

Professor Doutor Armando Jorge Miranda Sousa
Professor Auxiliar do Departamento de Engenharia Electrotécnica e Computadores da Faculdade de Engenharia da Universidade do Porto

Luís Santos Reis

Professor Doutor Luís Paulo Gonçalves dos Reis
Professor Auxiliar do Departamento de Engenharia Electrotécnica e Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.

Catarina Maria Brito de Noronha Santiago

Autor - **CATARINA MARIA BRITO DE NORONHA SANTIAGO**

Faculdade de Engenharia da Universidade do Porto

# Abstract

More and more, sports are relying on technology to improve the sportsman performance, not only individually but also within a team, by identifying and evaluating the parameters that define a good movement and/or a good tactic.

The aim of this project is to study and perform image analysis of indoor sports events (like handball and basketball) through a non invasive automatic image processing system.

A system like this can be a very useful source of information not only to coaches but also to teachers, since it will aid them in the study of sports' teaching, by providing a systematic approach to sports' analysis. It's a very ambitious project due to its complexity, since it involves following individually several similar objects that are constantly moving, changing in shape as seen by the cameras in the presence of partial overlapping and/or occlusion.

The steps taken towards the implementation of such a system include a preliminary study of the requirements, solutions and technologies in order to develop a robust engineering solution.

Several colour image processing techniques are used namely background subtraction, blob colour definition in the RGB and HSL colour spaces and colour blob manipulation to detect the players. The tracking is based on past information and on future probable areas. The usage of the OpenCV library allows having a clean interface with the image.

In order to validate the algorithms and techniques developed tests were conducted in the FADEUP's (Faculty of Sports from the Oporto University) sports hall with a Sony SNCDM110 surveillance IP camera. These tests proved that the detection algorithm produces good results that range from 75.2% until 93.3% of correct identification.

From the data analysis it was also possible to see that the tracking algorithm already provides an elementary track of the players but there is still open space for future improvements.

# Resumo


Cada vez mais o desporto confia na tecnologia para melhorar a performance dos desportistas, não só individualmente mas como pertencendo a uma equipa, através da identificação e avaliação de parâmetros que definem um bom movimento e/ou uma boa táctica.

O objectivo deste projecto é estudar e realizar a análise de imagens de eventos desportivos em pavilhão coberto (como andebol e basquetebol) através de um sistema automático de processamento de imagem não invasivo.

Um sistema como este constitui uma valiosa fonte de informação não só para treinadores mas também para professores, uma vez que permite ajudá-los no estudo do ensino do desporto, fornecendo uma abordagem sistemática à análise do desporto.

Dada a sua complexidade constitui um projecto muito ambicioso, visto que os objectos sob análise muito similares, estão constantemente em movimento e a mudar de forma em relação às câmaras, para além de poderem frequentemente ocorrer sobreposições e/ou oclusões entre jogadores o que faz a tarefa de individualização de cada jogador ainda mais árdua.

Para a implementação deste sistema foi efectuado um estudo preliminar dos requisitos, soluções e tecnologias necessárias para garantir uma solução de engenharia robusta.

São usadas diversas técnicas de processamento de imagem a cores nomeadamente extracção do background, definição e manipulação de conjuntos de cor tanto no espaço RGB como HSL para detecção dos jogadores. O seguimento de jogadores é baseado em informação passada e na definição de áreas de probabilidade futura. O uso da livraria OpenCV permite um interface transparente com as imagens.

De modo a validar os algoritmos e técnicas desenvolvidos foram realizados testes no pavilhão da FADEUP (Faculdade de Desporto da Universidade do Porto) com uma câmara IP de vigilância (Sony SNCDM110). Estes testes demonstraram que o algoritmo de detecção produz bons resultados com percentagens de detecção correcta que variam entre 75.2% e 93.3%.

Da análise dos resultados foi ainda possível verificar que o algoritmo de seguimento consegue seguir os jogadores de uma forma elementar, existindo espaço para melhorias.

# Acknowledgments

I would like to take the chance to thank the people who helped me of any way during this project.

To my family and friends for always supporting and motivating me.

To my supervisors, Prof. Armando Sousa and Prof. Luís Paulo Reis, for supporting and helping me during the project.

To Prof. Luísa Estriga and Prof. João Cunha for aiding me recording the videos at FADEUP's sports hall.

To Sony Portugal for lending me the camera used during the test phase.

# List of Contents

# List of Figures

# List of Tables

# Abbreviations and Symbols

List of abbreviations

| | |
|---|---|
| 2D | Two dimensions |
| CCD | Charge Couple Device |
| CMOS | Complementary Metal Oxide Semiconductor |
| CMYK | Cyan Magenta Yellow Black |
| CPU | Central Processing Unit |
| FADEUP | Faculdade de Desporto da Universidade do Porto |
| FIFA | Fédération Internationel de Football Association |
| FOV | Field of view |
| Fps | Frames per second |
| GHz | Gigahertz |
| H | Height |
| HSL | Hue Saturation Luminance |
| HSV | Hue Saturation Value |
| IP | Internet Protocol |
| MB | Megabyte |
| MJPEG | Motion Joint Photographic Experts Group |
| PC | Personnel Computer |
| RGB | Red Green Blue |
| SDK | Software Development Kit |
| SNC | Sony Network Camera |
| USB | Universal Serial Bus |
| W | Width |

List of symbols

| | |
|---|---|
| " | Inch |
| cm | Centimetre |
| $f$ | Focal length |
| $f_H$ | Horizontal focal length |
| $f_V$ | Vertical focal length |
| m | Meter |
| mm | Millimetre |
| $\bar{x}$ | Average |
| $\sigma$ | Standard deviation |

# Chapter 1

# Introduction

This chapter intends to introduce the subject of the project, including its objectives as well as the factors that lead to the choice of such a theme. A brief description of the document structure is also given.

## 1.1 - Motivation

In recent years we have assisted to a growing interest in applying automatic techniques to address the problem of people tracking, this in a global sense, but also in some specific applications such as sports games.

In sports, especially at high level competition a small advantage of one team regarding another can be of great importance and decisive for a match or even a complete championship.

Therefore, teams are always trying to improve their performance and tactics in order to gain this advantage. So, it would be of greater help to record the game sequence from strategic points where the field can be completely surveyed and where the movements of all players may be recorded with precision. This way, after the game, the team's coach can identify their week points and define measures to improve the team global behaviour.

Besides acquiring high quality images it would also be interesting to track the players and analyse their behaviour during the game.

An automatic system such as this would bring many advantages, namely it can handle a huge amount of data and perform a systematic evaluation which would be a very time consuming task and not always systematic if performed by a human being.

## 1.2 - Objectives

The main objective of this project is to develop a non intrusive automatic system, based on computer vision, to collect and analyse images from indoor sports events.

Besides this objective, there is also the aim to perform a study of relevant existing systems and techniques applied to equal or similar situations and search for actual hardware available in the market in order to define a robust engineering solution.

There was also the concern to validate all the techniques applied and algorithms developed in a real situation with the support of FADEUP.

## 1.3 - Description of the Problem

An indoor sport is played in a sports hall that has very specific characteristics. Usually the same sports hall can be used for different kinds of sports such as handball, basketball or volleyball. Therefore the markings for all of these sports are superimposed on the floor.

Of the enumerated sports, handball is the one that occupies the largest area: the field measures 20x40 meters followed by basketball with 28x15m and last volleyball with 18x8m.

Team sports have in common the fact that players wear similar equipments, move very rapidly, are constantly changing their direction, interact with one another and often cause occlusions which makes player tracking a very complex problem.

A vision system to follow players around the field must also take into consideration that from the cameras perspective the players' shape is constantly changing, for example a player lying on the floor is different from standing up.

Of course each of these sports has its own specificities, for example handball is a very fast game, with the main action zone near the 6 meter line. Is a high contact game where occlusion between players of the same team and different teams occurs very frequently. Statistics indicate that a player can achieve velocities of 7m/s and the ball 28m/s.

On the other side, volleyball is a slower game where each team is completely individualized by the middle net, however occlusions and merging always occur between players of the same team.

Basketball is in between, the main action zone is near the basket and due to the small field dimensions this action can change very quickly from one basket to the other.

## 1.4 - Structure

This document is organized in 6 chapters. Each chapter comprises a small introductory text, a development of the topic and a brief summary at the end.

The first chapter, Introduction, is subdivided in 4 sections that contextualize and introduce the subject under analysis. The next chapter, State of Art, refers to the theoretical concepts adjacent to a vision system, including the most commonly used sensor types, data transmission protocols, colour spaces and image enhancement techniques . It also provides an overview of what has been done in the area of sports analysis.

Considerations regarding the problem characteristics, namely the topology of the sports hall and the hardware requirements are made in Chapter 3, Engineering Approach. In addition, the parameters of the cameras and the lenses are established; different solutions concerning the number and placement of the cameras are studied resulting in two engineering solutions.

One is based on the Firewire protocol and another on the GigEthernet. Finally is given a description of the test platform used.

Chapter 4, Software Approach, includes four main sections, the first deals with the software requirements; the second gives a brief overview of two processing image libraries, OpenCV and OpenFrameWorks and the library used to interface with the Sony camera; the third section is about the software architecture and finally the fourth section describes the different techniques developed and used on the image processing system.

The following chapter, Results, presents and analyses the results achieved sustaining the conclusions drawn in the last chapter.

# Chapter 2

# State of Art

This chapter intends to give the reader a brief introduction to the theoretical concepts adjacent to the thematic and also an overview of what has been done on this field.

## 2.1 – Vision Systems

A digital camera vision system combines several different parts that go from the sensor type until the image enhancement techniques. This section provides a short overview of the most common sensor types, transmission protocols, colour spaces and image enhancement techniques, as well as their advantages and disadvantages.

### 2.1.1 – Sensor Types

There are two main digital camera sensor types available in the market the CMOS and the CCD. CMOS stands for "Complementary Metal Oxide Semiconductor" and CCD for "Charge Couple Device".

Both types of sensors convert light into electrical charge, but in different ways. In CCD sensors when exposure is complete each pixel charge is converted into a voltage, buffered and sent off the chip as an analog signal, all the pixel area is used to light capture and therefore provides a high uniformity.

On the other hand in a CMOS sensor each pixel is a combination of a photodiode and a MOS transistor, which means that the charge to voltage conversion is done at pixel level. Often the sensor also includes amplification, noise correction and digitalization circuits that deliver a digital signal.

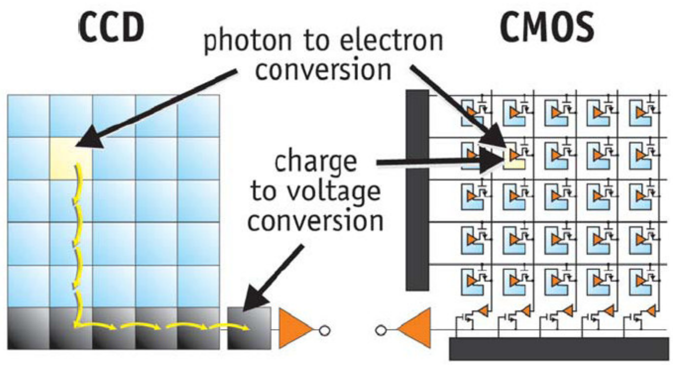



**Figure 2.1 - Inside the sensor: CCD vs. CMOS (1)**

There are several parameters that can be evaluated when comparing these two kinds of sensors, Litwiller (2) proposes the following:

- Responsivity corresponds to the amount of signal the sensor can deliver per unit of optical energy (photon). Usually CMOS are superior in this parameter, because gain elements are easier to place in it and are closer to the photoreceptor. Also the usage of complementary transistors allows low-power/high-gain amplifiers.
- Dynamic range consists on the ratio of a pixel's saturation level and its threshold signal. CCDs are better in dynamic range because they possess less on-chip circuitry, a high tolerance to bus capacitance variations and output amplifiers that due to their geometry can be easily adapted to minimize noise.
- Uniformity is the consistency of response for different pixels under identical illumination conditions. CMOS used to be much worse than CCD sensors, because each pixel had an open-loop output amplifier, whose gain varied considerably from pixel to pixel due to wafer processing variations. However, the introduction of feedback-based amplifiers[1] made the uniformity of some CMOS sensors closer to that of CCDs. Nevertheless, CCDs are still much better in darkness conditions.
- Shuttering is the ability to start and stop exposure arbitrarily, is a standard feature in all CCD cameras and highly important in machine vision applications. CCD have a higher performance in electronic shuttering, since CMOS sensors need an extra opaque shutter transistor per pixel (filling an area that would otherwise be active) that compromises the fill factor[2].
- Speed is the strong point in favour of CMOS since all the camera functions can be placed on the image sensor. However, CMOS technology has devoted its efforts to

[1] A feedback amplifier is one in which the output signal is sampled and fed back to the input to form an error signal that drives the amplifier.

[2] Photodiode area divided by the pixel's total area.

consumer applications that do not require so much speed and therefore this advantage is not that explored.

- Windowing is the capability of reading out a portion of an image sensor and an inherent feature of CMOS technology. This allows elevated frame or line rates for small regions of interest. CCDs generally have limited capabilities in windowing.

So in terms of image quality the CCDs still seem to be the best choice, nevertheless the CMOS technology is making efforts to improve.

Just by curiosity CCDs sensors were chosen to integrate the Hubble Space Telescope.

### 2.1.2 – Camera Interfaces

There are several camera interfaces provided by camera manufacturers, each has its own specificities and one cannot say that this is better than the other, it depends on the application.

From the available camera interfaces Firewire, Cameralink, USB 2.0 and GigEthernet are the most commonly used. Next a brief description of each of these interfaces is given, as well as their main advantages and disadvantages.

#### 2.1.2.1 – Firewire

Firewire, also known as IEEE1394 standard was developed by Apple, Texas Instruments in the $90^{ths}$.

It is an easy to use technology that allows “plug and play”, and is widely spread among consumer computers. However, it has a low bandwidth, from 400Mbits/s (IEEE1394a) to 800Mbits/s (IEEE1394b) and limited cable lengths (4.5 meters for IEEE1394a without repetitors, and 100 meters with shielded or optical cables for IEEE1394b).

#### 2.1.2.2 – Cameralink

Cameralink is the interface with the highest bandwidth, 2.38 Gbits/s. This is a well established technology and provides a wide range of cameras. However it needs a dedicated and expensive frame grabber, there is no standardized camera communication protocol and the cable length is limited to 10 meters, which can be extended at an added cost by the usage of extenders and fibre optic adapters.

#### 2.1.2.3 – USB 2.0

USB stands for Universal Serial Bus and was developed in the 1990s. It was incorporated as a standard in the same year as the Firewire.

In theory USB 2.0 allows a throughput of 480Mbits/s but the increased CPU and host control overheads due to the master-slave topology actually reduce it to values lower than the IEEE1394a. The maximum cable length allowed is 5 meters and can be extended with a maximum of 5 hubs until a total of 30 meters.

#### 2.1.2.4 – GigEthernet

Gigabit Ethernet, also called GigE, is a recent camera interface and consists on a serial network standard that has very high bandwidth, around 1000Mbits/s, very long cable lengths (100 meters) and offers a low cost interface that is widely used in standard computer networking hardware. All these features are making the GigE the most popular camera interface.

### 2.1.3 – Colour Spaces

A colour space is a mathematical abstraction describing the way colours can be represented as sets of numbers (these sets have typically three to four components) and mapped into a function. Therefore a colour space can be defined as the region where it is allowed to plot points that represent a colour.

There are several colour spaces and this comes from the fact that there are many applications where image is a key factor namely printing, monitor visualization and even image processing. The following subsections describe the most common used colour spaces.

#### 2.1.3.1 – RGB Colour Space

The RGB colour space consists of an additive tristimulus model in which Red, Green and Blue are combined in different ways to describe another colour.

This colour space is quite related to how the human eye physiologically perceives colours. The human eye is composed by two types of photoreceptors: the cones and the rods.

The rods are more numerous (around 120 million) and sensitive than the cones, they are responsible for our dark-adapted or scotopic vision, which means they are sensible to luminous changes.

The eye colour sensitivity is provided by around 6 to 7 million of cones, which can be divided into three types: large (64%), medium (32%) and short (2%), depending on their wavelength absorption peak. Due to the wavelength absorption peak they are often classified as red, green and blue cones.

**Table 2.1 - Cone wavelength sensitivity**

| *Cone type* | *Wavelength range* | *Corresponding colour* |
|---|---|---|
| **Small (S)** | 400-500 nm | Blue |
| **Medium (M)** | 450-630 nm | Green |
| **Large (L)** | 500-700 nm | Red |

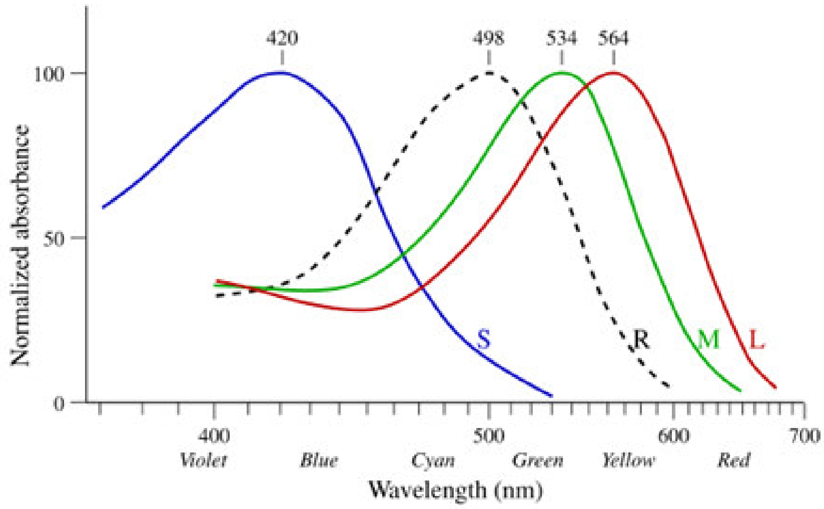



**Figure 2.2 – Normalised absorption spectra of human cone (S, M, L) and rod (R) cells (3)**

The cones are also able to detect finer details and more rapid changes in images, because their response times to stimuli are faster than those of the rods.

The RGB colour space is frequently represented by the RGB cube, that has on the vertices the primary colours (red, green and blue), their complementary colours (magenta, yellow and cyan), black and white. Equal values of red, blue and green produce greys and lie along the main diagonal of the cube (running from black to white).

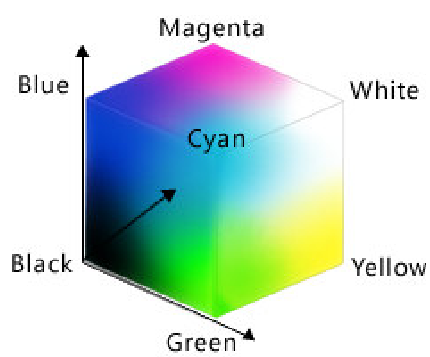



**Figure 2.3 - RGB colour space represented by the RGB cube (4)**

### 2.1.3.2 – HSV and HSL Colour Spaces

HSV and HSL colour spaces are transformations of the RGB colour space. These models try to define a more “intuitive” way of describing colours.

H stands for hue and allows to classify a colour into red, yellow, green, blue or an intermediate value between any contiguous pair of these colours (5), S represents saturation and gives a measure of how much a colour is diluted by the white colour, V for value and defines where a particular colour lies along the lightness-darkness axis, lastly L corresponds to the luminance and reflects the human subjectivity to perceive the brightness of a colour.

Geometrically the HSV colour space is usually represented by a cylindrical or conic object. The cylindrical form is a more accurate representation of this colour space, however most of the times it is represented with a conical form because it is not possible to distinguish between saturation and hue when their values are lowered.

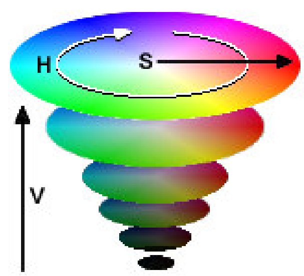



**Figure 2.4 - HSV geometrical representation (6)**

The relation between the RGB and HSV colour spaces can be defined by the following equations, as described by Travis (7) :

$$
\begin{aligned}
&S = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)} \\
&V = \max(R,G,B) \\
&\text{if } S = 0 \text{ then hue is undefined otherwise :} \\
&\begin{cases}
H = 5 + B' & \text{if } R = \max(R,G,B) \text{ and } G = \min(R,G,B) \\
H = 1 - G' & \text{if } R = \max(R,G,B) \text{ and } G \neq \min(R,G,B) \\
H = 1 + R' & \text{if } G = \max(R,G,B) \text{ and } B = \min(R,G,B) \\
H = 3 - B' & \text{if } G = \max(R,G,B) \text{ and } B \neq \min(R,G,B) \\
H = 3 + G' & \text{if } R = \max(R,G,B) \\
H = 5 - R' & \text{otherwise}
\end{cases}
, \text{where}
\begin{cases}
R' = \dfrac{\max(R,G,B) - R}{\max(R,G,B) - \min(R,G,B)} \\
G' = \dfrac{\max(R,G,B) - G}{\max(R,G,B) - \min(R,G,B)} \\
B' = \dfrac{\max(R,G,B) - B}{\max(R,G,B) - \min(R,G,B)}
\end{cases}
\end{aligned}
\qquad \textbf{(2.1)}
$$

The HSL colour space is similar to the HSV, but instead of representing the space with only one cone it uses two cones as the following picture denotes.

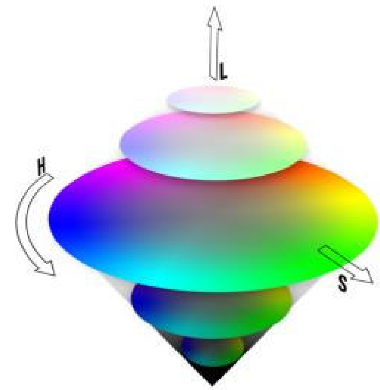



**Figure 2.5 - HSL geometrical representation (8)**

Gonzalez and Woods (9) propose the following conversion between RGB and HSL colour spaces:

$$H = \begin{cases} \theta & if \;\; B \leq G \\ 360 - \theta & if \;\; B > G \end{cases}$$

$$\theta = \frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)\right]^{1/2}} \tag{2.2}$$

$$L = \frac{R+G+B}{3}$$

$$S = 1 - \frac{3}{R+G+B}\min(R,G,B)$$

From both representations it is possible to see that the HSL colour space gives in fact more freedom, for example light can range from white to black in HSL while on HSV can only go from the desired colour until black.

#### 2.1.3.3 – CMYK Colour Space

CMYK (cyan-magenta-yellow-black) is a subtractive colour space (opposing to the previous colour spaces presented) and is commonly used in printing applications. The subtractive noun comes from the fact that this model works by partially or entirely masking certain colours on a white background.

The conversion between a RGB space and a CMYK space is not an easy task and for it to be satisfactory must make use of complicated polynomial arithmetic or lookup table interpolations.

#### 2.1.3.4 – YUV Colour Space

YUV colour space is used in the PAL and NTSC colour TV systems and is composed by a luminance component (Y) and two chrominance components (UV).

Its relation with the RGB colour space is given by the following transformations (10):

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.144B \\ U &= -0.147R - 0.289G + 0.436B \\ V &= 0.615R - 0.515G - 0.100B \end{aligned} \quad \textbf{(2.3)}$$

#### 2.1.3.5 – YST Colour Space

YST is a very recent colour space proposed by Benedetto et al (11) to watermark images of human faces and is robust to illumination changes. As in the previous colour space (YUV) Y corresponds to the luminance component while ST represent the two chrominance components.

The S (skin) component results by estimating the average values corresponding to a set of different colours of human faces and the T component is identified by the Gram-Schmidt procedure to guarantee an orthogonal colour space.

### 2.1.4 – Image Enhancement

During years several techniques and algorithms have been developed to aid in the image processing and therefore it becomes an advantage to have a previous knowledge of them. Some of these techniques and algorithms are explained next.

#### 2.1.4.1 – Image Subtraction

Image subtraction is a very useful technique applied on image processing enhancement. Its use implies the need of a background image where the object under analysis is not present.

Several medical applications use this technique, for example, when analysing the blood vessels of a person initially an image of the region under analysis is taken and after injection of a dye into the bloodstream a new picture is taken. The subtraction of these two images will give the location of the dye inside the human body and determine if there are any blocking points.

In mathematical terms image subtraction can be expressed as

$$g(x, y) = f(x, y) - h(x, y) \quad \textbf{(2.4)}$$

### 2.1.4.2 – Image Smoothing

An image can be smoothed by applying smoothing filters, also called low-pass filters. These filters let the low frequency components pass and reduce the high frequency components. Therefore they are indicated to improve the quality of pictures affected by noise. They also allow eliminating small irrelevant details.

In practice these filters compute a pixel value (colour) taking into consideration its neighbours. The set of neighbours forms what is called of a window. The higher the number of neighbours used the larger the window and the blur on the image.

The average and the Gaussian filters are linear smoothing filters. The first computes a pixel as being the average of its neighbours as demonstrate by equation (2.5).

$$f(x,y)=\frac{1}{(2k+1)^2}\sum_{u=x-k}^{u=x+k}\sum_{v=y-k}^{v=y+k}g(u,v) \tag{2.5}$$

The Gaussian filter is different because it doesn’t give the same weight to all the neighbours. It obeys to the Gaussian distribution where the weight of each neighbour related to position (x,y) is computed by:

$$w(u,v)=\frac{1}{2\pi\sigma^2}e^{-\frac{(u-x)^2+(v-x)^2}{2\sigma^2}} \tag{2.6}$$

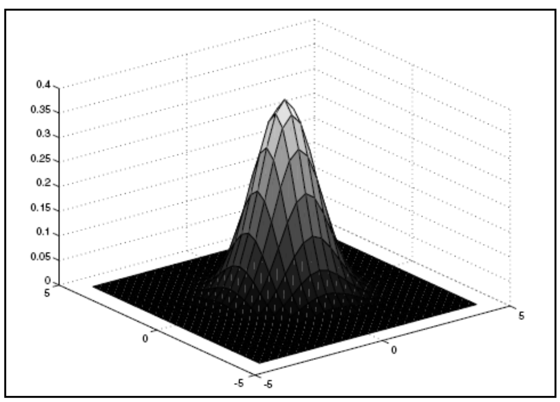



**Figure 2.6 - 2D Gaussian distribution with $\bar{x}$ = (0,0) and $\sigma$=1(12)**

The median filter is a nonlinear smoothing filter that computes the central pixel value as being the median of the pixels inside the defined window. This filter usually has very good results in images with salt and pepper noise.

These filters can also be applied in the frequency domain for that it is necessary to apply the Fourier transform to the original image, multiply the result by the filter transfer function and take the inverse transform to produce the smoothed image.

#### 2.1.4.3 – Image Sharpening

Sharpening filters are the opposite of smoothing filters and act as a high-pass filter highlighting the fine detail of an image. This makes them the perfect candidates for edge detection.

The most common methods to perform image sharpening use the gradient notion:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{2.7}$$

The Sobel operator computes the first derivative at any point in an image using the magnitude of the gradient at that point. This operator has the advantage of providing both a differencing and smoothing effect.

Mathematically, the operator uses a 3x3 window to calculate approximations of the derivatives - one for horizontal changes and another for vertical. Afterwards the pixel value (G) is computed as the magnitude of the gradient vector ($G_x$, $G_y$). Both masks as well as the pixel value can be determined using the following equations:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \qquad G = \left[G_x^2 + G_y^2\right]^{1/2} \tag{2.8}$$

The two masks are applied to each colour component independently.

Another well know and very frequently used sharpening filter uses the Laplacian operator that computes the second derivative at any point in an image.

The basic requirements in defining the Laplacian operator is that the coefficient associated with the central pixel must be positive, the coefficients associated with the outer pixels must be negative and the coefficients sum must be zero.

$$G = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \tag{2.9}$$

Like in the previous filter, the Laplacian mask is applied to each colour component independently.

Being a second order derivative filter, the Laplacian is typically too sensitive to noise. In addition it produces double edges and is unable to detect edge direction.

#### 2.1.4.4 – Edge linking

Edge detection consists on a first step in image processing because usually the edges detected are imperfect and have missing points which complicates the process of grouping points into a specific shape.

There is a very powerful method known as the Hough transform that addresses this problem. It is based on groupings edge points into object candidates by performing an explicit voting procedure over a set of parameterized image objects.

The image under analysis is scanned and each pixel that was identified as belonging to an edge is tested to see all the desired structures that could pass through it. Whenever this is true the structure receives a vote. Structures that really are on the picture will receive more votes and will become evident.

Figure 2.7 shows an example of two pixels defining a circle. In this case it’s the green circle that wins since it will receive two votes, one per each pixel. The blue circles will only receive one vote each.

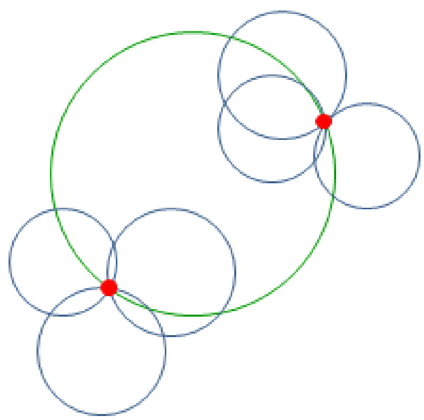


**Figure 2.7 - Applying the Hough transform to detect a circle**

The classical Hough transform is applied to features that can be specified in some parametric form (lines, circles, ellipses, …). There is also a generalized Hough transform that can be employed to features that cannot be described through analytic description but it involves a high computational effort.

## 2.2 – Sports Analysis

In recent years there has been a growing interest in developing automatic systems for detecting, tracking and identifying player’s movements. A system like this is complex since it involves objects that are all very similar, constantly moving, changing of shape and getting together which makes an hard task to individualize each player.

The two following subsections give a brief overview of what has been done in this area. The first is dedicated to soccer, the most popular outdoor sport and the second is dedicated to indoor sports like handball and basketball. A brief reference to robotic soccer is given since there are several similarities between it and indoor sports.

Although, in theory, indoor and outdoor sports can be viewed as the same object of study, when paying attention to the details it is easy to see that some small differences can make a huge difference.

For example, outdoor sports usually are played in a more vast area which makes the problem even more complex, a soccer field can measure as much as 120mx90m while a handball field measures 40mx20m.

To make it even worse outdoor sports have another particularity, there is no ceiling and therefore cameras cannot be placed above the players, which is a very useful place for them.

On the other side a sports hall is usually used for different indoor sports, which means that the floor has different superimposed line marks, also the artificial illumination system can introduce significant noise into the problem.

### 2.2.1 – Outdoor Sports

Like stated before in outdoor sports, placing the cameras is a very hard task and therefore many authors use the images provided by the TV cameras, for example Liu, Jiu et al (13) use an approach based on a broadcast moving camera system that doesn't need manually initialization since it has a learning capability. The background is characterized through a dominant colour learning algorithm and the players' model uses a Haar feature based boosting cascade. The players' detection and their tracking are performed by a Markov chain Monte Carlo data association.

FIFA World Cup 2006 was their test platform and from the results their method seems to have high detection and labelling precision, around 90%.

Dearden et al (14) also use a single moving camera system but their camera is dedicated and placed in the stadium by the halfway line. This disposition prevents the camera from covering the entire field and therefore they pay special attention to the fact that players can leave and enter the area covered by the camera.

They concentrate their efforts on the tracking algorithm and apply a sample importance resampling particle filter, which removes the need for modelling the process as linear and helps overcoming the occlusion's problems. The usage of a particle filter also allows that no previous assumptions on the players shape and size are made, since they are represented by the collection of particles.

Some authors, for example Ming et al (15) use a multi-fixed camera system which gives a better view but the image processing system gains a more complex structure.

In their approach each camera is initially treated as an independent source of information and performs change detection based on an adaptative background and image-plane tracking and only then the measurements from all the cameras are combined with a central tracking process based on a Kalman tracker to model the players' position and velocity.

Evidence on following the ball is also discussed but each player is not individualized and authors report several occlusion problems, which could be minimized by exploring the multi-camera views.

There are also some commercial solutions available such as ProZone (16), Amisco Pro (17), SportsCode (18), Orad TrakVision (19) and Trakus (20).

## 2.2.2 – Indoor Sports

Pers et al (21) present a two fixed camera system placed in the ceiling of the sports hall that provides a bird's eye view. Their system is in a very mature stage and they provide an almost commercial application, Sagit, very easy to use.

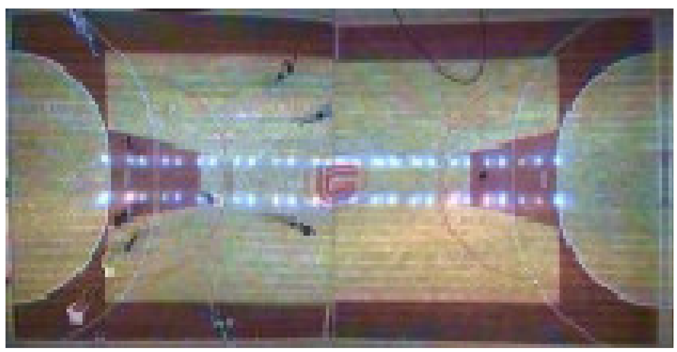


**Figure 2.8 - Bird's eye view from Pers et al (21)**

A detailed explanation for camera temporal and spatial calibration and error analysis is given and the main objective of their work was to follow trajectories. Three different algorithms are compared and described, one based on motion detection, another on RGB (colour) tracking and a last one on a colour and template tracking. They apply their method to several sports such as handball, basketball and squash.

Several authors apply Kalman filter techniques to address some of these problems, for example Liu, GuoJun et al (22) give a detailed explanation on their efforts to track skaters in a skating competition. They define a hierarchical model of two components: the player's helmet – identified by a template matching approach and body – detected by a colour histogram matching method and combine it with an unscented Kalman filter in order to detect and track the players throughout the field. They also describe how to create a panoramic image from a single panning camera.

Chris et al (23) use a single camera system and apply a multiple object condensation approach to a 5 player soccer game. Initially a sample (that corresponds to a bounding box) of each player is detected, then through a propagation algorithm the fitness of each bounding box is evaluated and adjusted. They also include an improved predictive stage that incorporates estimates of position from Kalman filters.

In the latter years there has also been a great progress in robotics soccer and numerous papers were published (24), (25), (26) and (27) describing several features developed for image processing and tracking algorithms tested with success. Some of these features due to the similarity between indoor sports can also be further explored.

## 2.3 – Summary

This chapter presented a brief overview of vision systems and a special emphasis was given to sensor types, camera interfaces, colour image analysis and enhancement.

It provides a description of:

- the two main types of digital sensors, CMOS and CCD as well as a comparison between them
- four camera interfaces Firewire, Cameralink, USB2.0 and GigEthernet
- several colour spaces and the transformations needed to convert from one colour space into another
- some image enhancement techniques

A study on what has been done in the sports analysis area is presented and divided in two major groups, the outdoor sports where soccer is on focus and the indoor sports including a brief reference to robotic soccer.

# Chapter 3

# Engineering Approach

This chapter presents a detailed explanation of the problem by describing the sports hall characteristics and the hardware requirements.

It gives a special emphasis on choosing an adequate solution, taking into consideration the choice of the camera, the camera interface and the lens.

It also provides a study on different possible engineering solutions that can meet the requirements and the sports hall characteristics. Finally describes the configuration used as test platform.

## 3.1 – Sports Hall Characteristics

In any engineering solution, before starting the development phase it is necessary to know and understand the object of study. In this particular case is important to know the characteristics of the sports hall since it defines the area of interest. Section 1.3 - Description of the Problem already gave a brief introduction.

In general a sports hall can be used for different kinds of indoor sports. The sports hall used as test platform (FADEUP's sports hall) can accommodate basketball, handball and volleyball in one single space.

Figure 3.1 shows the dimensions of the field for each sport. The sport occupying the biggest area is handball with dimensions of 40mx20m plus an extra border of 2 meters along the width and 1 meter along the length, also the height of the sports hall is of 8 meters. The handball goal has 3 meters width by 2 meters height.

The volleyball field is smaller and the net is placed in the middle of the field with a maximum distance to the floor of 2.43 meters. The basketball basket is placed 3.05 meters above the ground.

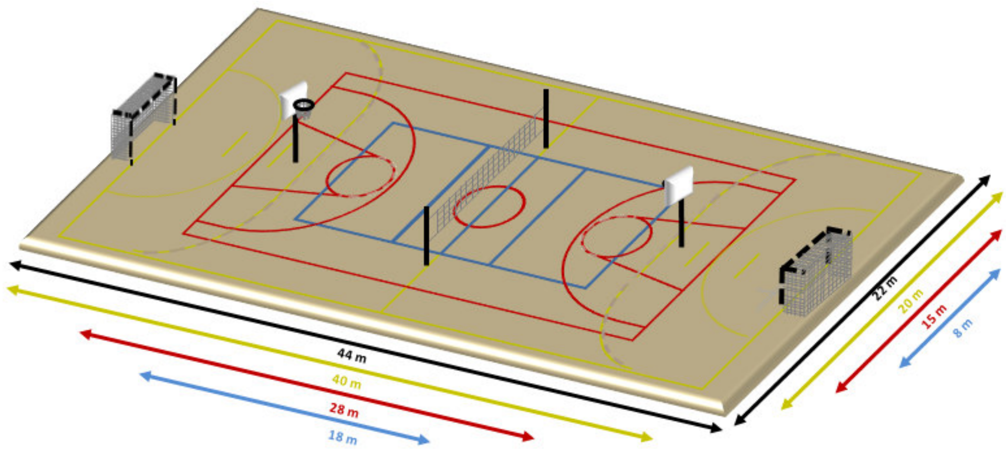



**Figure 3.1 - Field dimensions for handball (green), basketball (red) and volleyball (blue)**

In terms of illumination it is equipped with an artificial light system that provides clarity enough for a usual camera.

## 3.2 – Hardware Requirements

In terms of hardware requirements the system must be able to cover the entire field including the extra border and the image should have resolution enough to perceive details such as the ball and the lines of the field.

Typically the lines measure 5cm and the smallest ball, of handball, has a radius of 8cm.

The system must also take into consideration that players and ball can move at very high velocities, a ball can achieve 28m/s and a player 7m/s.

Another important fact is that for handball and basketball most of the action occurs in more or less the same space, the 6 meter line for the first and near the basket for the second. For volleyball most of the action happens near the net in the middle of the field.

An additional requirement is that the cameras must take into consideration the height the players and the ball can achieve, namely in zones such as the basket and the volleyball net.

## 3.3 – Choosing the Camera

When choosing a camera there are several aspects to bear in mind, one is intimately related to the sensor and another to the camera interface and in many cases price is also determinant. This section will take care of the issues associated with the sensor.

Regarding the sensor the key factors are type of sensor CCD or CMOS, colour or monochrome, noise immunity, spatial resolution and frame rate.

The sensor type chosen was the CCD and took into consideration the tendency they have for higher quality images and lower noise level.

CCDs also have a better performance in terms of light sensitivity because they have a better fill factor, since their active area is entirely composed of optical elements, opposed to the CMOS sensors that also need extra transistors near the pixel. These factors allied to a more mature stage give CCD sensors an advantage.

Regarding colour or monochrome in this specific application there is no margin for doubts, since the players must be identified by team it is necessary to detect the colour of the equipment they wear.

There are two possibilities for colour sensors, the Bayer sensor, where each pixel is represented by a single byte and gives information of one of the component colours (red, blue or green), or a truly colour sensor (that is basically three Bayer sensors placed together in one single camera), where each pixel is represented by three bytes, one per colour.

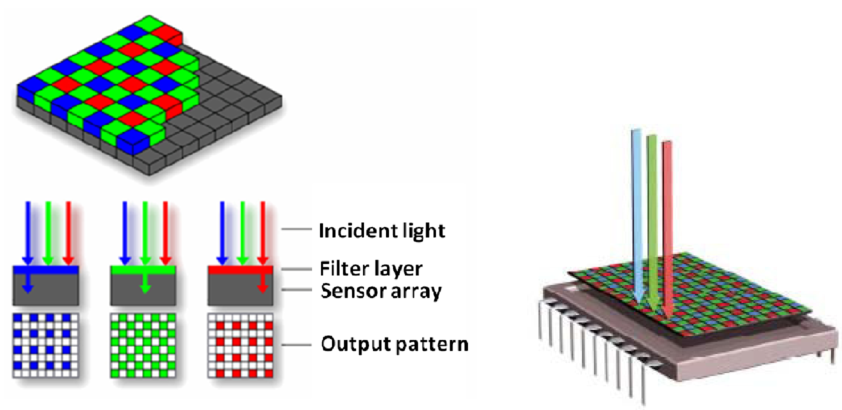



**Figure 3.2 - Bayer sensors (28)**

Of course a truly colour sensor gives more information, which seems a nice to have in a colour image processing system where the players' equipments can have similar colours.

Another factor when choosing a camera has to do with the choice of the spatial resolution. In this case, two points that were specified in the requirements must be taken into consideration:

- First the camera must be able to detect the lines of the field and the ball. Given the dimensions involved it seems valid that each pixel should not be responsible to cover more than 2cm. Figure 3.3 gives an idea of the number of pixels involved on the detection of the ball.

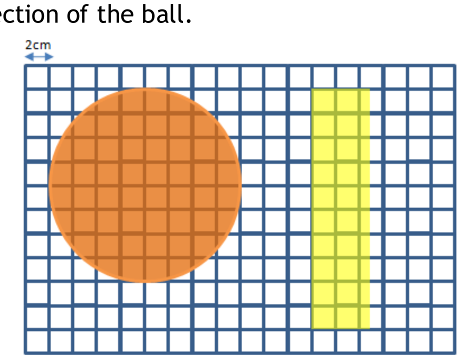



**Figure 3.3 - Pixel resolution to detect the field's lines and the ball**

- Second it must cover the entire field, which according to the field dimensions and the real area covered by each pixel would give, for a single camera, a resolution of:

$$W = \frac{44m}{0{,}02m/\,pixel} = 2200\,pixels$$
$$H = \frac{22m}{0{,}02m/\,pixel} = 1100\,pixels \qquad (3.1)$$

Cameras with such a resolution are hard to find and usually very expensive, therefore a multi-camera system must be considered. This topic will be further discussed in the next section.

The last factor when selecting a sensor is the frame rate, which must deal with the high velocity a ball can achieve. Typical frame rates for cameras are 15, 30 and 60fps. Higher values are not common and imply a huge increase in the price of the camera.

Table 3.1 gives an overview of what these three frame rates can achieve according to the values specified in the requirements for the velocities of the players and the ball.

**Table 3.1 - Delta movement seen by each frame with frame rates of 15, 30 and 60**

| *Object* | *15fps* | *30fps* | *60fps* |
|---|---|---|---|
| **Ball (cm)** | 180 | 90 | 45 |
| **Player (cm)** | 47 | 24 | 12 |

As expected the best choice is a camera with a high frame rate, so that the image can have high fidelity. However higher frame rates imply higher data transferences which can be a limiting factor when choosing the camera interface.

## 3.4 – Camera Placement and Details

The characteristics of the problem place the ceiling as the best spot to set the camera system, since there is no interference from the crowd, a single player does not fill the entire field of view of the camera and a bird’s eye perspective can carry numerous advantages.

Nevertheless, it poses some constrains namely in the length of the cables and the place where the PC can be, which can influence the choice of the camera interface technology.

As seen before a single camera system is not enough to cover the entire field, unless it can move.

Although the price of a system based on a single moving camera would be lower it would embrace a much more higher complexity, would not prevent dark areas in the game that could eventually be of interest to the coach, would be prone to aging and mechanical problems, therefore a multi-camera system must be considered.

A study of a multi-system composed by two, three and four cameras will be performed.

Given the parameters defined in the previous section, several colour CCD models may accomplish the requirements.

A thorough search in the manufacturers revealed several cameras namely the X-Pri from AOS with a spatial resolution of 1280x1024pixels and a maximum frame rate of 500fps, several models from Prosilica like for example GE1050C supporting a maximum frame rate of 60fps and a resolution of 1024x1024 pixels and some Dalsa cameras from the Genie series.

Despite accomplishing the requirements needed and some having even better results, these cameras all have very high prices.

There is another supplier, Imaging Source, which provides cameras with similar characteristics at much lower prices and has a long cooperation with the faculty. The following table shows the specifications of the models provided by this manufacturer that may be used for this application.

**Table 3.2 - Imaging Source camera's specifications**

| *Camera* | *Spatial Resolution* | *Frame per seconds* | *Sensor size* |
|---|---|---|---|
| **DFK 21Bx04**[3] | 640x480 | 60 | $\frac{1}{4}$" |
| **DFK 31Bx03** | 1024x768 | 30 | $\frac{1}{3}$" |
| **DFK 31Bx02** | 1280x960 | 15 | $\frac{1}{2}$" |

Figure 3.4 shows the dimensions, in centimetres, for every kind of sensor size considered.

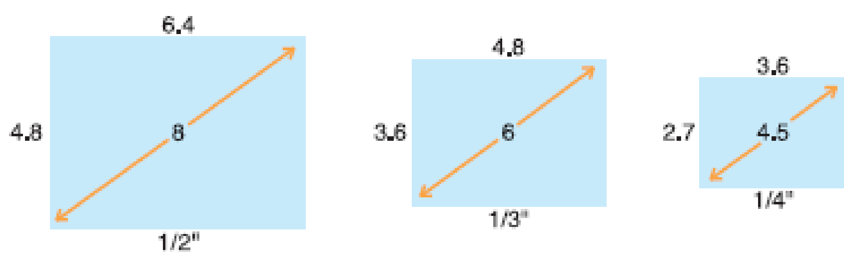



**Figure 3.4 - Sensor dimensions in centimetres**

For the three configurations, no camera mixing is considered so that the global image has the same resolution which allows a clearer view of the entire field.

[3] x – refers to the camera interface, it can be Cameralink, Firewire, USB2 or GigEthernet

### 3.4.1 – Two Camera System

Taking the field symmetry, the middle of each half handball field seems a good spot to place each camera. The following picture gives an idea of the area covered by the cameras including an overlap region between them. This overlap is needed since the players are not on the floor level and also because of the volleyball net, without the overlap a dead zone would be created.

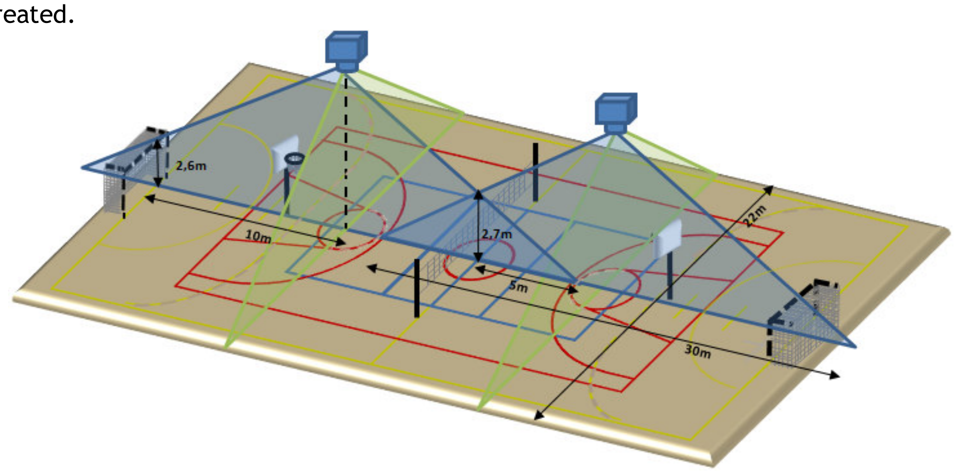



**Figure 3.5 - Two-camera system**

The overall disposition seems to cover the entire field for each considered sport without dead zones. Nevertheless the overlapping of the two regions gives a height of only 2.7m which in some volleyball situations can lead to loosing the track of the ball.

It is also important to cover the handball goal and the area behind it, which is accomplished with this configuration.

According to the figure each camera must cover an area of 30mx22m which results in a spatial resolution per camera model of:

**Table 3.3 - Spatial resolution for a two-camera system**

| *Camera* | *Theoretical* | *DFK 21Bx04* | *DFK 31Bx03* | *DFK 31Bx02* |
|---|---|---|---|---|
| **Length** | 2cm | 2.34cm | 2.93cm | 4.68cm |
| **Width** | 2cm | 2.29cm | 2.86cm | 4.58cm |

From the results the camera DFK 31Bx02 doesn't seem appropriate since the spatial resolution is too far from the theoretical value, the other two models present good values and the DFK 31Bx03 has the advantage of allowing a higher frame rate.

### 3.4.2 – Three Camera System

When thinking of a three camera system it makes sense that one is placed above the middle line, while the other two are placed symmetrically in each half field. Like before, attention must be paid in the transition between cameras to avoid dead zones.

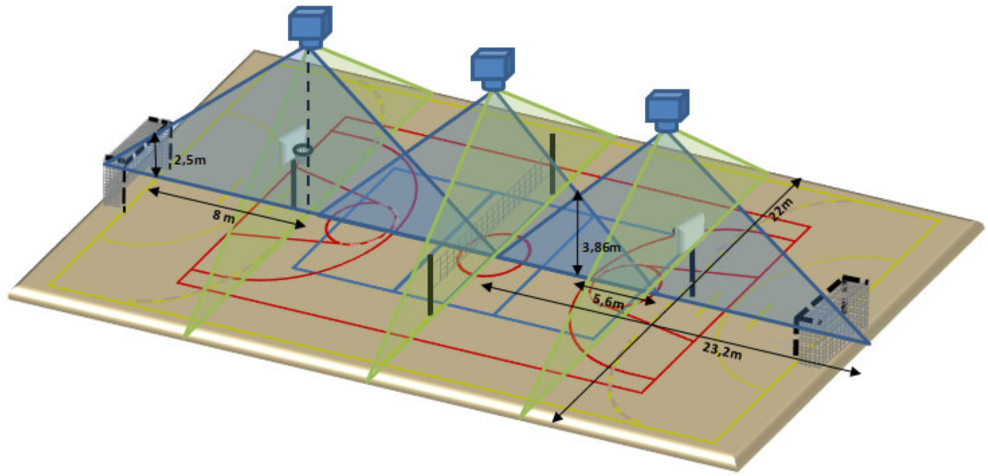



**Figure 3.6 - Three-camera system**

With a three camera system the dead zone probability decreases a lot and cameras are placed in the areas where most of the action takes place: for volleyball above the middle line net and for basketball almost above the basket. Yet handball gets a little bit penalized at the extremities of the field.

Although cameras are spotted right above the action regions it may not be an advantage since a vertical angle will provide a good view of the head of the player but not of his body, which in this case is the part that has more interest since it allows identifying the player as being part of either team.

Like before, the spatial resolution required for the cameras can be estimated. Each camera is responsible for an area of 23.2mx22m, and the results are compiled in the following table.

**Table 3.4 - Spatial resolution for a three-camera system**

| *Camera* | *Theoretical* | *DFK 21Bx04* | *DFK 31Bx03* | *DFK 31Bx02* |
|---|---|---|---|---|
| **Length** | 2cm | 1.81cm | 2.26cm | 3.59cm |
| **Width** | 2cm | 2.29cm | 2.86cm | 4.58cm |

As expected the spatial resolution of the cameras increases. The DFK 21Bx04 model having even better results than the theoretical and the DFK 31Bx02, which has the highest frame rate is approximating the desired values.

### 3.4.3 – Four Camera System

A four camera system can help improve the image acquisition in the handball 6 meters line, which is where most of the handball action happens and where the two previous systems don't give so much coverage. Although trying to improve this zone, the other areas must not be penalized therefore this four camera system must also take care of the height of the basket and the volleyball net.

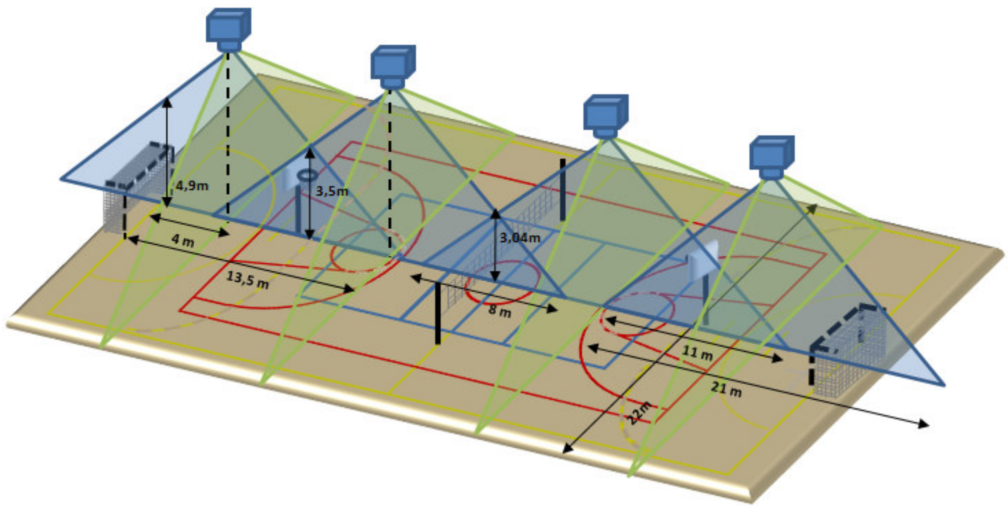



**Figure 3.7 - Four-camera system**

This configuration is very balanced and seems the best choice, first because it allows a better coverage of the entire field and provides a good overlap between zones covered by each camera, second the height on the action zones (6 meter line and goal for handball, key area for basketball and central net for volleyball) is adequate and last the angle the camera performs with the players in those action zones is not vertical. As stated before an oblique view is very important because it permits to see more area of the players' body and therefore better identify the team and better distinguish players in case of occlusion or player merging.

With this configuration each camera should cover an area of 22x21m which is a smaller area than in the previous configurations. The spatial resolution gets the following values:

**Table 3.5 - Spatial resolution for a four-camera system**

| *Camera* | *Theoretical* | *DFK 21Bx04* | *DFK 31Bx03* | *DFK 31Bx02* |
|---|---|---|---|---|
| **Length** | 2cm | 2.18cm | 2.73cm | 4.38cm |
| **Width** | 2cm | 1.72cm | 2.15cm | 3.44cm |

The DFK 31Bx03 camera seems the most suitable model in terms of spatial resolution. However it only provides 30fps which only allows seeing 90cm of ball displacement if travelling at the maximum speed. Therefore model DFK 31Bx02, although providing less spatial resolution has higher frame rate and therefore seems the most appropriate in this situation.

### 3.4.4 – Lens Choice

The choice of the lens is a very important step when designing an image processing system. The lens must be able to expose the entire sensor and allow a field of view that covers completely the area each camera is responsible for.

To address the first issue the lens must have the same or bigger dimensions than the sensor. So a sensor of 1/3" requires a lens of 1/3" or bigger.

Although a bigger lens implies a higher price, it has the advantage of providing increased resolution and picture quality as only the centre of the lens is being used. According to Table 3.2 a sensor of ½" would be possible for every camera and a sensor of 1/3" would be adequate for the other two models (DFK 31Bx03 and DFK 31Bx04).

In order to understand the concepts behind the second issue a few image notions, schematized in Figure 3.8, will be introduced.

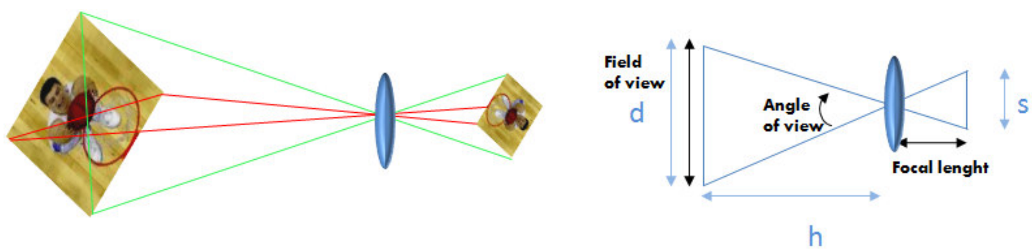



**Figure 3.8 - Image notions regarding the choice of a lens**

Focal length - The focal length of a lens is defined as the distance in mm from the optical centre of the lens to the focal point, which is located on the sensor or film if the subject (at infinity) is "in focus". Depending on the sensor (s) and object (d) sizes and on the distance to the object (h) the focal length must be different to accomplish a given angle of view. Usually there are three measures for the focal length: the horizontal, the vertical and the diagonal (29).

Formula 3.2 gives a generic way of finding the focal length.

$$f = s\frac{h}{d} \tag{3.2}$$

Angle of view – Describes the angular extent of a given scene that is imaged by a camera, and like the focal length has three measures (29). It’s relation with the focal length is the following:

$$\alpha = 2 \times \arctan(\frac{s/2}{f}) \tag{3.3}$$

Field of view – Is the part of the world that is visible through the camera at a particular position and orientation in space. Objects outside the FOV when the picture is taken are not recorded in the photograph (29).

The focal length is one of the most common and important measures manufacturers use to specify a lens, therefore to buy a camera it is necessary to know this *a priori*.

All the parameters required to calculate the focal length were determined in the previous sections and are summarized in the next table.

**Table 3.6 - Configurations overview**

| *Configuration* | *Length* | *Width* | ***Height*** |
|---|---|---|---|
| **2 camera system** | 30m | 22m | 8m |
| **3 camera system** | 23.2m | 22m | 8m |
| **4 camera system** | 21m | 22m | 8m |

Each camera has a different sensor type (see Table 3.2 and Figure 3.4) and therefore will require a lens with a different focal length as shown by Table 3.7.

**Table 3.7 - Focal length values for the different system configurations**

| Camera | ½" | | 1/3" | | ¼" | |
|---|---|---|---|---|---|---|
| *Configuration* | $f_V$(mm) | $f_H$(mm) | $f_V$(mm) | $f_H$(mm) | $f_V$(mm) | $f_H$(mm) |
| **2 cameras** | 1.74 | 1,71 | 1.31 | 1.28 | 0.98 | 0.96 |
| **3 cameras** | 1.74 | 2.21 | 1.31 | 1.65 | 0.98 | 1.24 |
| **4 cameras** | 1.74 | 2.43 | 1.31 | 1.83 | 0.98 | 1.37 |

The values of focal length involved in an application of this kind are very small and is not easy to find lenses with these characteristics.

After a market survey, two lenses' models with focal length near the estimated values were found: the 13FM22IR from Tamron with a focal length of 2.2mm and size of 1/3" and the Computar varifocal lens TG2Z1816FCS with focal length between 1.8-3.6mm and size of 1/3".


**Figure 3.9 - Tamron (left) and Computar (right) lenses**

### 3.4.5 – Camera Interface Technology

Before jumping into the analysis of the previously referred camera interfaces technologies it is important to have an idea of the bandwidth necessary, since each interface has very specific maximum bandwidths.

The information so far is the following:

1. a colour camera needs three bytes per pixel to carry the information
2. three cameras were chosen due to its relation price/characteristics that have resolutions of 640x480, 1024x768 and 1280x960 pixels
3. each camera can deliver 60, 30 and 15 frames per second, respectively

This gives a total bandwidth required per camera of:

**Table 3.8 - Bandwidth required by each camera model**

| *Camera* | *640x480 @ 60fps* | *1024x768 @ 30fps* | *1280x960 @ 15fps* |
|---|---|---|---|
| **Bandwidth** | 55Mbits/s | 71Mbits/s | 55Mbits/s |

Let’s then analyse each camera interface technology and determine the most adequate(s) for this specific application.

Cameralink biggest advantage is the high bandwidth, which in this case seems unnecessary since the values involved are not that high. Also Cameralink needs optical cable for long cable lengths and a very expensive frame grabber per camera.

USB 2.0 only allows cables lengths of 5 meters, with a maximum of 30 meters if using hubs. The usage of hubs in this kind of environment doesn’t seem very adequate since a loose ball can hit one and break the connection between cables. Also the increased CPU and host control overheads don’t place this camera interface in a good position.

The two remaining technologies, Firewire and GigEthernet, are the best candidates for this application. Both consist in relatively cheap technologies, which are included by default in the majority of computers, have good indicators in terms of CPU and host control overhead, data transfer reliability and the bandwidth is adequate to this problem.

The GigEthernet interface allows for a simpler solution since cable lengths of 100 meters are allowed, nevertheless a Firewire solution must not be discarded, since it is a technology with given proofs.

The architecture of the system considering these two camera interfaces will be discussed in the next section.

## 3.5 - Architecture

As observed in section 3.4 – Camera Placement and Details, the hardware solution that best fits this problem is based on a four camera system. As a result this section combines a four camera system with GigEthernet and Firewire technologies in order to define two different architectures.

### 3.5.1 – Firewire Architecture

The Firewire technology doesn't allow for long cable lengths except if using optical or shielded cables. The usage of such cables makes the solution much more expensive therefore and since the cameras are already placed in the ceiling it is an option to also place the PCs there.

This architecture uses two PCs, one per two cameras, placed in the ceiling in the mid point between the two cameras. The connection between the cameras and the PCs is made with a normal Firewire cable and the PCs are connected to the local intranet through a common internet cable.

The following figure illustrates the architecture:

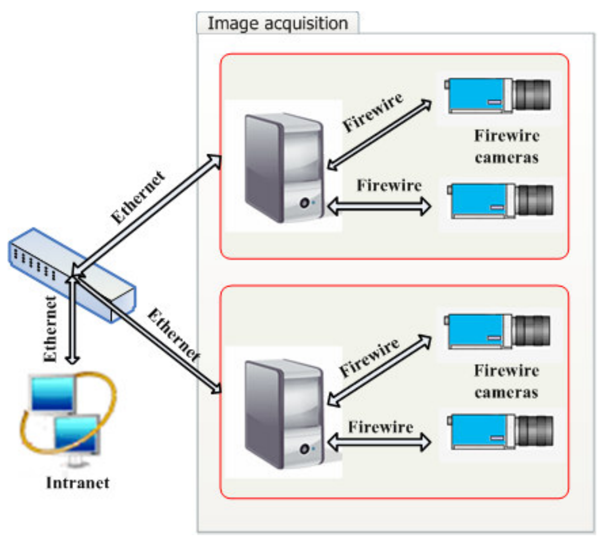



**Figure 3.10 - Firewire architecture**

This architecture has a main drawback, the PCs are placed in a low accessibility place and in case of problems it is necessary a crane for reaching them.

During a game situation it is not possible to place a crane in the middle of the field to solve the problem therefore special attention must be given to the PCs and the connections in case of such an implementation.

### 3.5.2 – GigEthernet Architecture

The GigEthernet technology allows long cable lengths (until 100meters). This means that the PCs can be on the floor level, which is a huge advantage in case of problems.

Like in the previous architecture two PCs, one per two cameras, are considered in order to minimize the data traffic between the cameras and the PCs and the CPU congestion.

All the connections are performed via Ethernet cables.

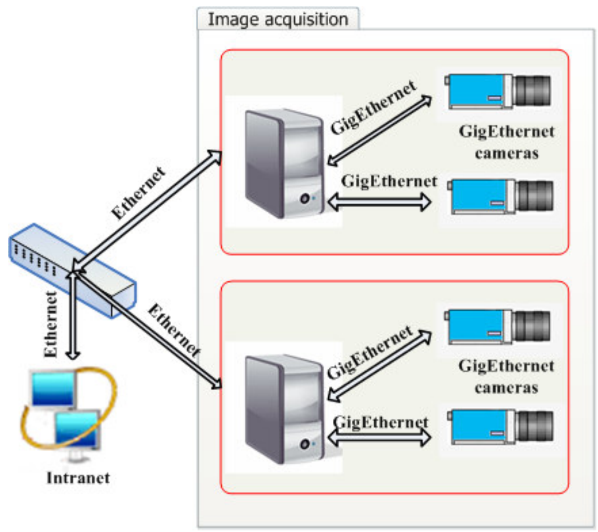



**Figure 3.11 - GigEthernet architecture**

Another advantage of this architecture is that it allows the integration of more cameras without many changes since it is only needed to have a hub with more ports.

Due to the considerations and conclusions gather throughout this chapter it seems that the final engineering solution should have the following characteristics:

- fixed four camera system
- GigEthernet technology
- DFK 21Bx04 cameras from Imaging Source with spatial resolution of 640x480 pixels and 60 frames per second
- lenses model 13FM22IR from Tamron or TG2Z1816FCS from Computar

## 3.6 – Test Platform

Due to several constraints it was not possible to implement the projected engineering solution. A test platform based on a single Sony SNCDM110 IP surveillance camera was mounted at FADEUP's sports hall. This camera was lent by Sony Portugal during a short period of time.

The camera was placed in the ceiling above the 6 meter line as shown by the following figure.

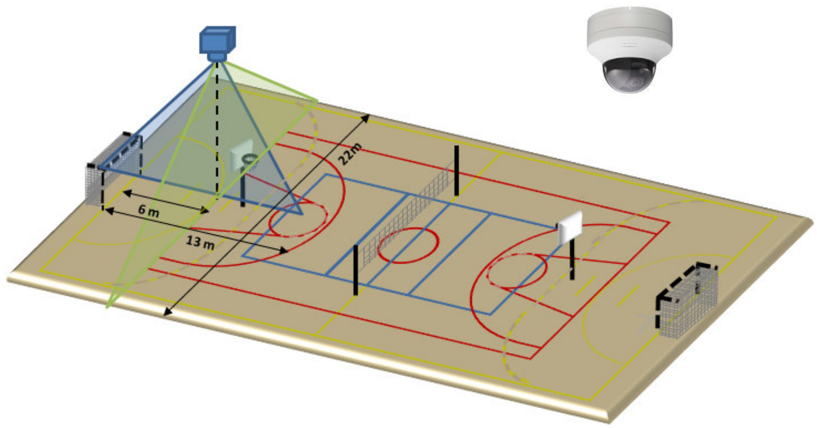



**Figure 3.12 - Mounted camera system**

The camera has a CCD sensor with maximum image size of 1280x960 pixels and frame rates of 30 fps (640x480) or 15fps (1280x960).

The camera connection to the PC is performed through an Ethernet cable.

## 3.7 – Summary

This chapter identifies the characteristics of a sports hall and the hardware requirements.

According to those two points a camera market study was performed, considering not only the cameras capabilities but also its costs.

A detailed exposition of the following three points is given:

- camera placement system, which includes an in depth analysis of the area covered by each camera giving special attention to areas where most of the game action occurs
- lenses properties in order to achieve the values determined in the previous point
- camera interface technology to guarantee a high fidelity signal without loss of information

Two possible architectures are presented, one based on the Firewire camera interface technology and another based on GigEthernet. Although both architectures seem promising, the latter presents a simpler solution.

At the end is given a description of the test platform implemented and used to test the algorithms and techniques developed during the project.

# Chapter 4

# Software Approach

This chapter introduces the software requirements as well as the software architecture and the user interaction with the application.

Two image processing libraries are analysed OpenCV and OpenFrameWorks, including their structure and most common application areas.

It can also be found a detailed explanation of the several methods and techniques applied to the images collected as well as the algorithms developed.

## 4.1 – Requirements

In terms of software requirements two situations must be defined, one that can provide very detailed information, the ultimate solution, and another that is suited for the duration of this project.

The requirements for this last situation include:

- collect and store images from sports events
- detect players from one of two teams with different colour equipments
- elementary player tracking
- store the players' positions throughout the time (this allows for some simple statistics such as covered area and time spent running)

The ultimate solution besides addressing these requests should also be able to:

- detect and track the ball
- provide good player tracking
- analyse the players' behaviour
- identify and classify individual and collective game concepts such as shoots, number of passes, players disposition in offensive and defensive situations
- store and provide visualization mechanisms for these metrics

## 4.2 – Development Tools

### 4.2.1 – Image Processing Libraries

There are several image processing libraries that provide an easy and transparent way for dealing with images and performing data image processing. Two of these libraries were considered, the OpenCV and the OpenFrameWorks. The choice for these libraries was intimately related to being open source and therefore usable by everyone.

#### 4.2.1.1 – OpenCV

OpenCV is an open source computer vision library developed by Intel and first launched in 1999. Its main objectives as stated in (30) are:

- provide not only open but also optimized code for basic vision infrastructure
- disseminate vision knowledge by providing a common infrastructure that developers could build on, so that code would be more readily readable and transferable
- advance vision-based commercial applications by making portable, performance-optimized code available for free

To accomplish such objectives OpenCV provides numerous functions and is divided in five main components:

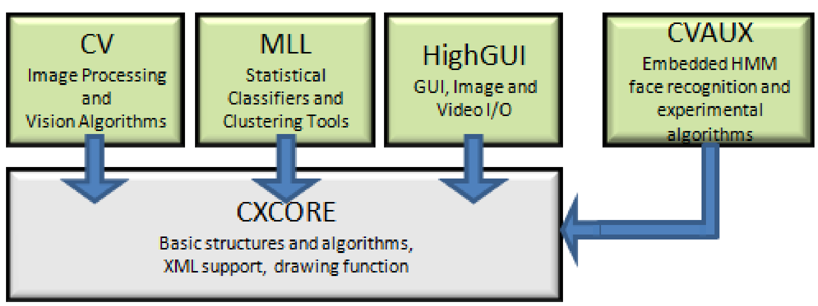



**Figure 4.1 - OpenCV main components (adapted from (31))**

Just to have an idea of how powerful a library like this is a few examples of functions provided by it are given below:

- cvConvertScale - copies one array to another with optional scaling and/or optional type conversion
- cvSub – computes per-element difference between two images
- cvSobel – calculates first, second, third or mixed image derivatives using extended Sobel operator
- cvErode – erodes image by using arbitrarily structuring element
- cvCaptureFromFile – initializes capturing video from file

#### 4.2.1.2 – OpenFrameWorks

OpenFrameWorks has a different ideology regarding its applications and its authors describe it as a free application framework designed for “creative coding”.

The library intention is to work as glue and combine together several commonly used libraries under a tidy interface: openGL for graphics, rtAudio for audio input and output, freeType for fonts, freeImage for image input and output, quicktime for video playing and sequence grabbing.

It is organized into classes as can be seen by the following diagram.

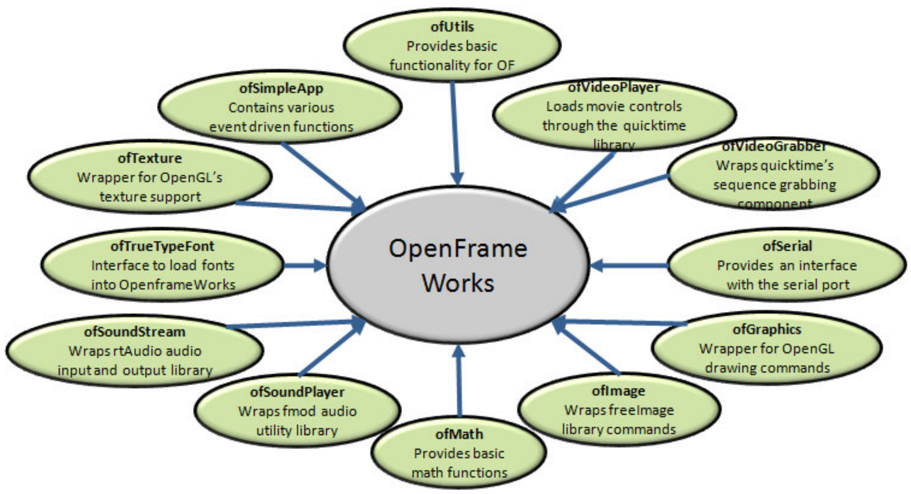



**Figure 4.2 - OpenFrameWorks class organization**

OpenFrameWorks is a more generic library having a vast spectrum of applications.

For this kind of application OpenCV seems a better tool since it is specific for image processing and provides a great number of functions that implement the main image processing algorithms.

### 4.2.2 – SNC SDK Software Development Kit

Sony provides a software development kit for the client application, Sony Network Camera SDK software, which allows operating the camera.

This software consists of multiple libraries developed in C++ for Windows Operating System.

Each library has very specific functions that are schematized in the following diagram:

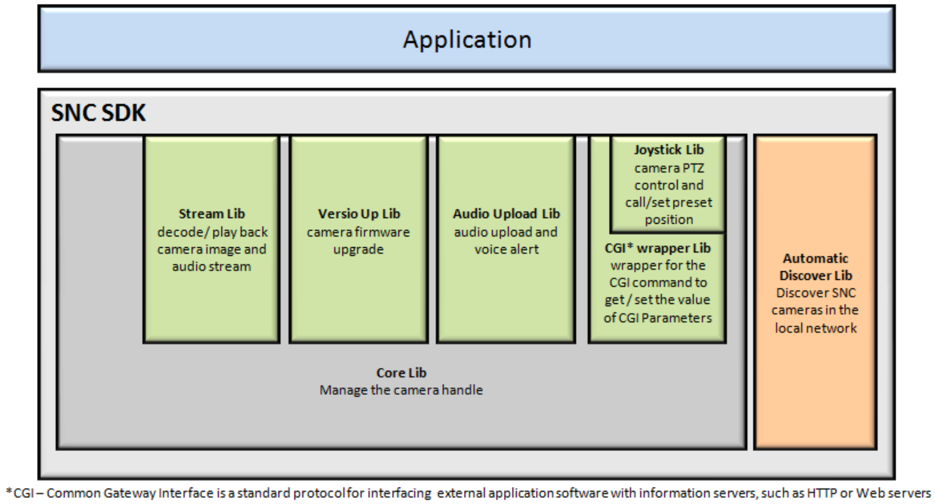



**Figure 4.3 - SNC SDK structure**

## 4.3 - Architecture

The software architecture is build upon two main systems: the Image Acquisition and the Image Processing systems.

The Image Acquisition system is responsible for enabling the acquisition of images and for storing the video stream in a data store.

The Image Processing system uses the video stream stored previously and then performs the players' detection and tracking. Its final result, the players' positions, is again stored in the same data store of the video stream.

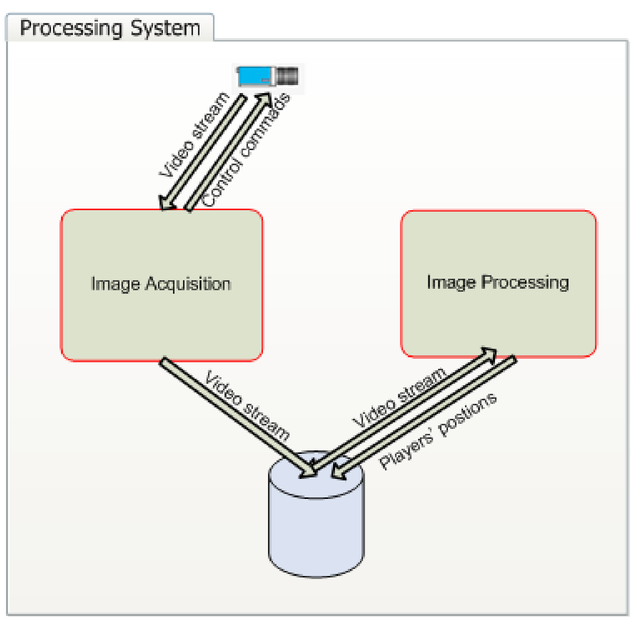



**Figure 4.4 - Software architecture**

Figure 4.4 gives a schematic overview of the interactions between these two systems.

## 4.3.1 – Image Acquisition

Before starting the image acquisition the user must provide the camera IP address, the location and name of the file that will contain the video stream.

It is also the user responsibility to trigger the start of the acquisition process. The frames that continuously arrive are stored in the specified file until the user issues the stop command.

The following sequence diagram exemplifies the interaction between the user, the Image Acquisition system, the data store and the camera.

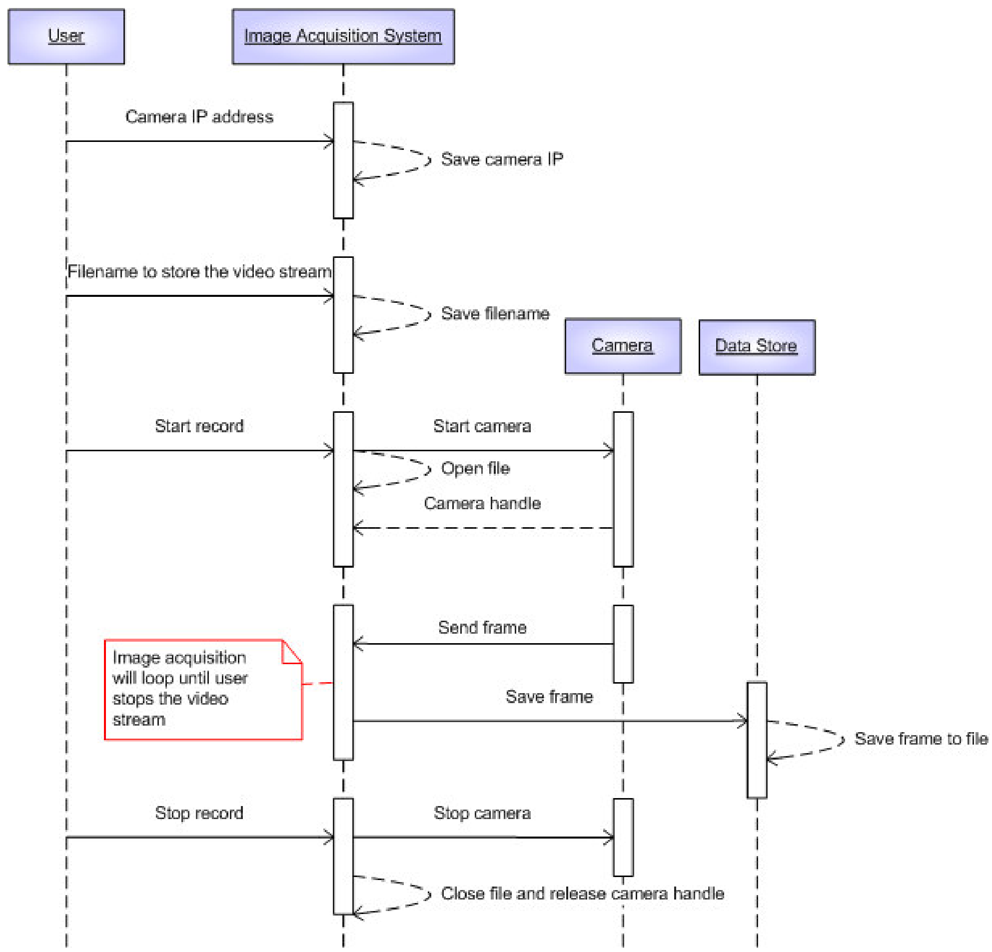



**Figure 4.5 - Image acquisition sequence diagram**

## 4.3.2 – Image Processing

The Image Processing system is more complex and asks for a more demanding user interaction. This section presents a brief overview regarding the interactions but doesn't go into the detail of the functions and methods used, nor the specific steps taken inside it.

The main interactions the user can have with this system are exemplified in the use case of Figure 4.6.

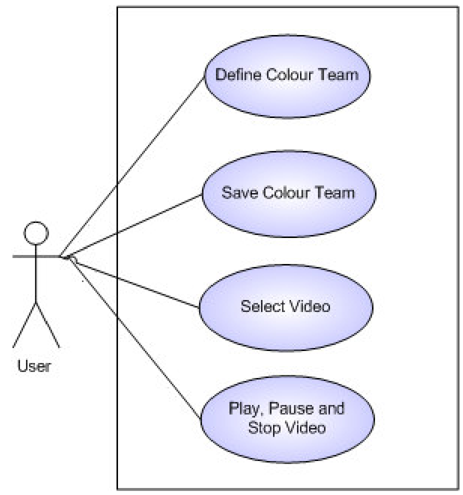



**Figure 4.6 - User interaction with the Image Processing system**

Like before this system also obeys to a specific sequence. So initially the user selects the desired video, after which he can choose only to see or really analyse it.

In case of choosing to analyse the video, the user must calibrate the colours of the team(s) he wants to track. This colour calibration can be achieved in two ways, either by loading an older calibration that was stored on a file or by doing it on the fly.

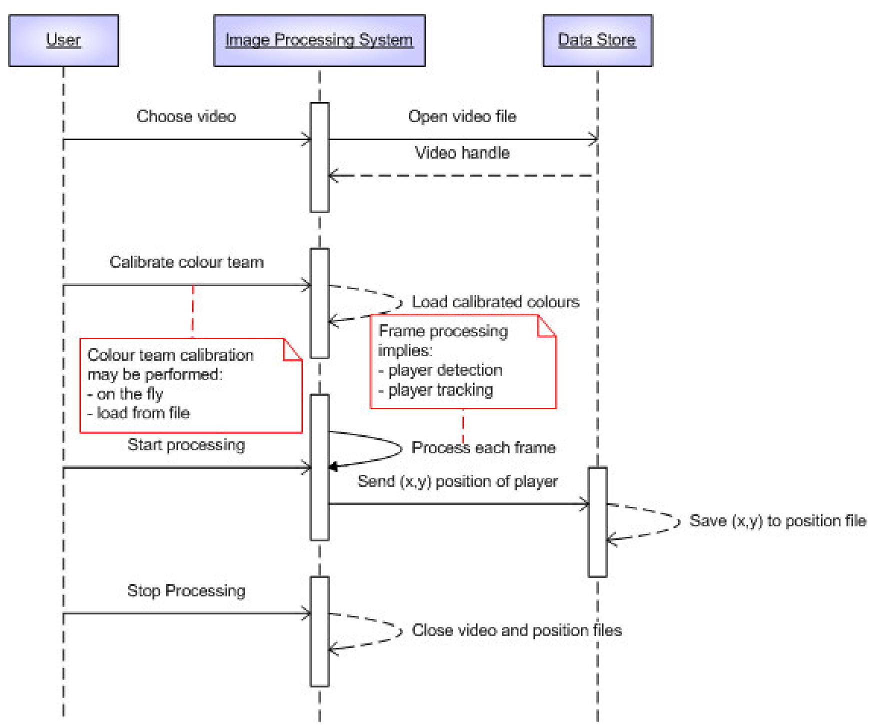



**Figure 4.7 - Image processing sequence diagram**

During the processing the positions of the players ($x_{cm}$,$y_{cm}$) and their respective team colours are being saved to a file allowing the possibility to evaluate the players' performance, namely the length they run in a game. The sequence diagram of Figure 4.7 exemplifies the processing flow in a typical situation and Figure 4.8 represents the processing sequence for one frame.

The frame processing sequence is composed of five steps: background subtraction, colour detection, pseudo-mode filter, blob aggregation and characterization, real world transformation and last player tracking.

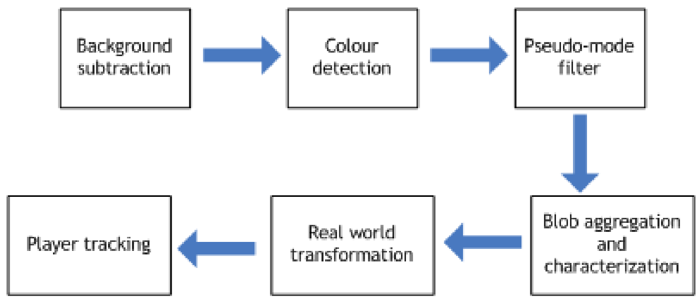



**Figure 4.8 - Frame processing sequence**

Next section provides a detailed explanation of each frame processing steps and also of the team colour definition.

## 4.4 - Player Detection and Tracking

This section presents the experiments made in term of software in order to implement a system able of analysing images from team sports.

The application was developed in C++, using Visual Studio and based on a Windows Operative System. OpenCV was the library chosen to aid in the images handling.

### 4.4.1 – Team Definition

Before starting the players tracking it is necessary to define each team colour. This is a very important task since a good colour calibration will influence the success of the subsequent steps.

The colour calibration is performed with the help of the mouse. Clicking the mouse left button above a pixel will trigger an event that stores the colour pixel as well as the team it belongs into a lookup table.

This lookup table corresponds to a three dimensional vector composed by 32k elements (32 positions in each dimension). The following diagram and equation help to understand how this lookup table is filled in.

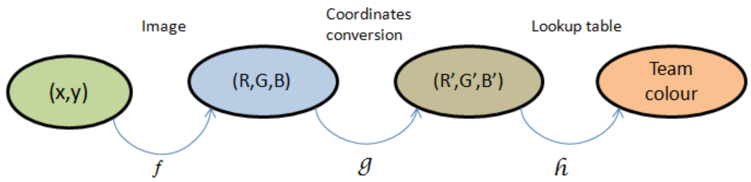



**Figure 4.9 - Relation between 32K lookup table and the image**

$$g(R',G',B') = \begin{cases} R' = R >> 3 \\ G' = G >> 3 \\ B' = B >> 3 \end{cases} \tag{4.1}$$

Although the image from the camera represents a pixel colour with 24 bits (8 bits per component – R, G, B), the lookup table only uses 15 bits ( 5 bits per component – R', G', B'), that correspond to the most significant bits of each component. The least significant bits of each colour component represent very slight changes in the colour itself, for that reason as well as for computational purpose they are ignored.

In this lookup table each point defined by the R', G' and B' components can have a few values, corresponding to a team colour or no team. The set of points belonging to the same team colour constitute a colour region.

Besides this, each team colour has a colour identifier that is used to mark a pixel that belongs to a given team colour.

It is a very time consuming task to click in every pixel of a player's equipment to fill in the lookup table, therefore three approaches were developed and are explained in the following subsections.

#### 4.4.1.1 – Physical Range

This is the simplest method tested and consists on defining a square area around the point where the mouse was placed. All the colours of the pixels within this area will be collected and, after passing by the g function, introduced into the team colour lookup table.

The area around the central point can be bigger or smaller depending on the user choice. For the following image, we get a team colour region, for the yellow team, represented by the cube on the right side.

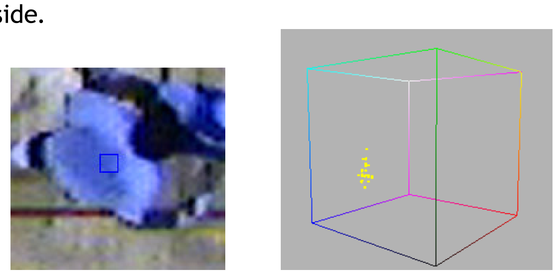


**Figure 4.10 - Defining a team by the physical range method**

Although improving the time necessary to define each team colour it is still very time consuming and not always produces good results.

#### 4.4.1.2 – Physical Flood

This method introduces two novelties regarding the previous one. The first is that a click on one single pixel (consider it the seed pixel with the seed colour) will be propagated to the neighbour pixels if they have a colour that is similar to the seed colour and to its previous neighbour. The second is that this colour difference can be measured either on the RGB colour space or on the HSL colour space.

The propagation among pixels will happen in all physical directions (in this case x and y, which means all 8 neighbour of a pixel are considered) in a recursive way until reaching a pixel that has a colour too far away from the seed or from the previous neighbour.

So in terms of colour, two regions are defined one that has a wider area and represents the colour region around the colour seed and a smaller that represents the colour region around the neighbour pixel.

These regions are three dimensional for the RGB colour space and two dimensional for the HSL colour space as denoted by Figure 4.11.

The HSL space is only expanded in two dimensions, saturation and luminance. The expansion is not performed in the hue direction (remember that hue is responsible for defining the colour as red, blue or orange) since in this case the intention is not to expand the team colour to other colours but expand the same colour in terms of white dilution (saturation) and brightness (luminance).

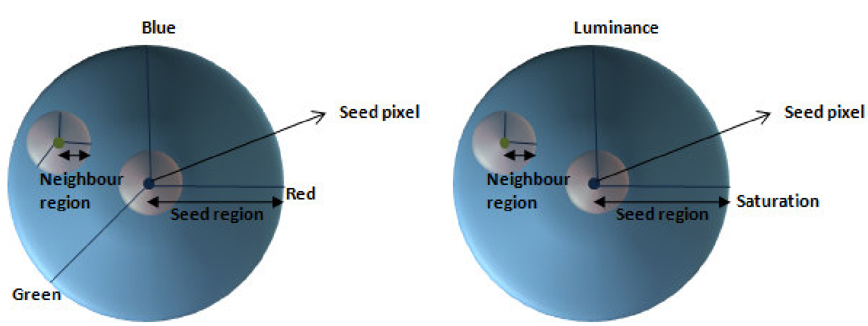



**Figure 4.11 - Physical flood colour regions (RGB on the left and HSL on the right)**

Although the starting point is a physical pixel the consequence is a flood in the colour space. Figure 4.12 shows the result of expanding a pixel colour through the physical flood method using the RGB and HSL colour spaces. The results are always presented on the RGB colour space.

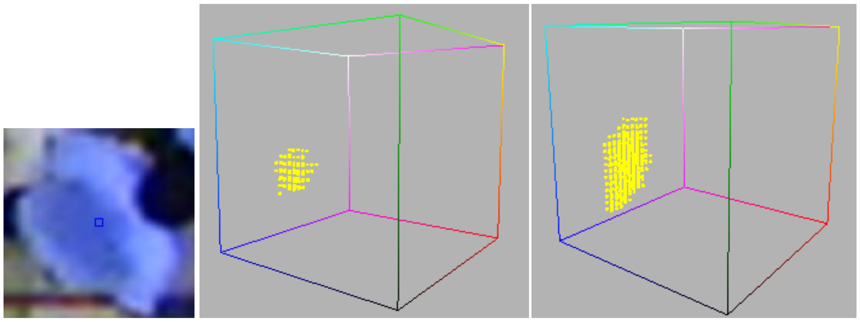


**Figure 4.12 - Defining a team by the physical flood method using the RGB colour space (middle cube) and the HSL colour space (right cube)**

The two RGB cubes show that the expansion on the HSL colour space is more effective and gives a wider region.

#### 4.4.1.3 – Colour Grow

This method allows an expansion based uniquely on the colour of the selected pixel, the area of colour to be expanded can be wider or slimmer based on the user choice. Like before an RGB or HSL colour expansion is allowed. With this method the colour expansion is not restricted by the physical neighbourhood of the pixels and therefore, depending on the size of the allowable expansion area, it gives a faster team definition method.

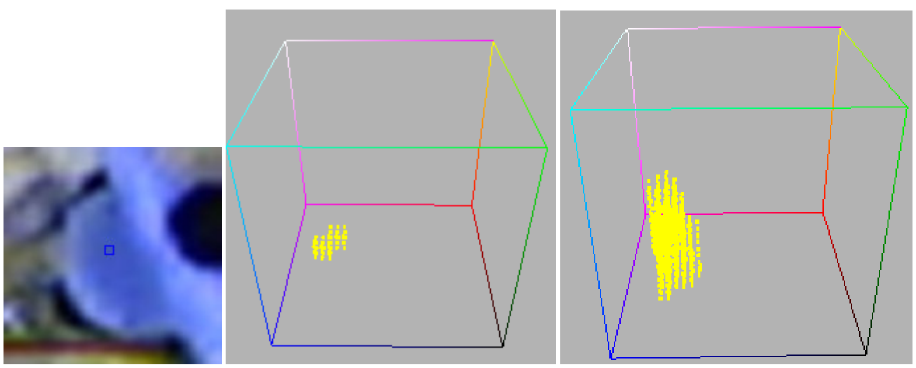


**Figure 4.13 - Defining a team by the colour grow method using the RGB colour space (middle cube) and the HSL colour space (right cube)**

As in the previous situation the expansion on the HSL colour space is wider and allows a faster team colour definition.

A comparison of these three approaches is given in the next figure. The two last methods were expanded in the HSL colour space due to its best performance.

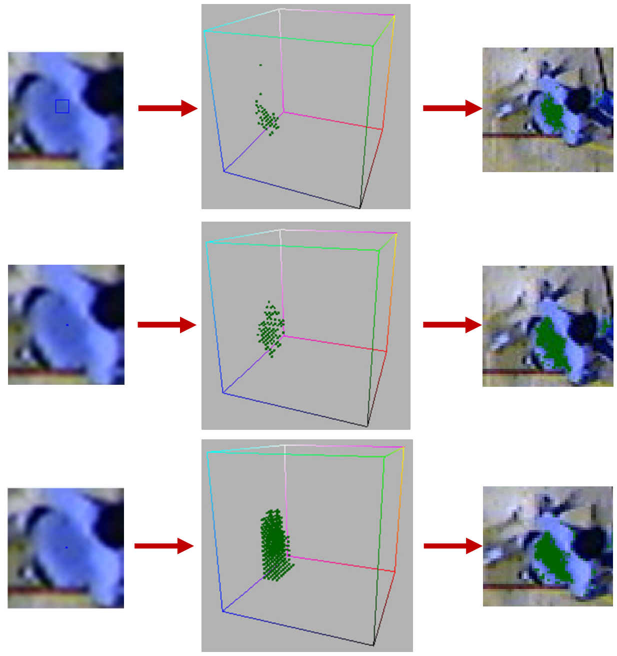


**Figure 4.14 – Team definition: physical range (up), physical flood (middle) and colour grow (bottom)**

### 4.4.2 – Background Subtraction

Most of the image area has non-useful information. In fact, the regions of interest are the ones that include a player so it makes sense to perform a background subtraction in order to highlight the zones where the players are.

To perform this subtraction an empty image of the field was collected *a priori*.

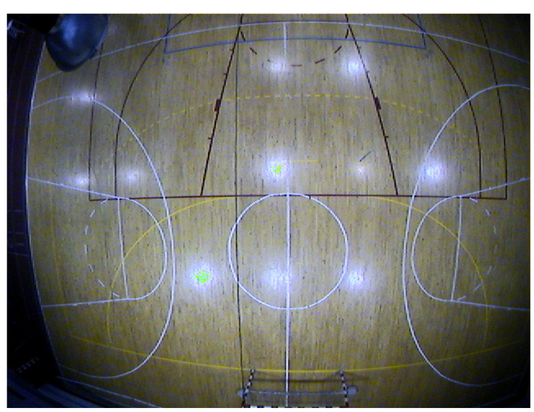


**Figure 4.15 - Empty image of the field for background subtraction**

The first step towards the background subtraction consisted in a pure background subtraction where the background image is subtracted to the image to be analysed. This technique usually has very good results in images where the background is much darker than the regions of interest, which is not the case.

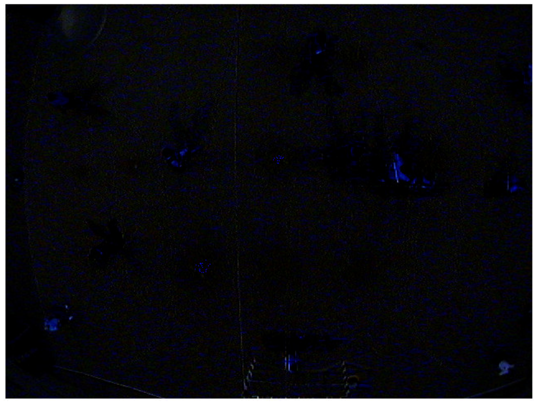


**Figure 4.16 - Pure background subtraction**

Due to the brightness of the background the pure subtraction removes not only the field but also has a huge impact on the players' figures almost eliminating them from the image as can be seen in Figure 4.16.

In order to avoid players of vanishing from the image another method was tested and consists on a per pixel conditional subtraction.

The subtraction is only performed if the pixel of the image under analysis has a colour similar to the colour of the same pixel in the empty field image. To be more specific the two pixels are not exactly subtracted but the pixel in the image under analysis will be updated with the white colour.

The threshold used to determine if a pixel must be subtracted or not was determined experimentally.

This method presented very good results and a great improvement as can be seen in Figure 4.17.

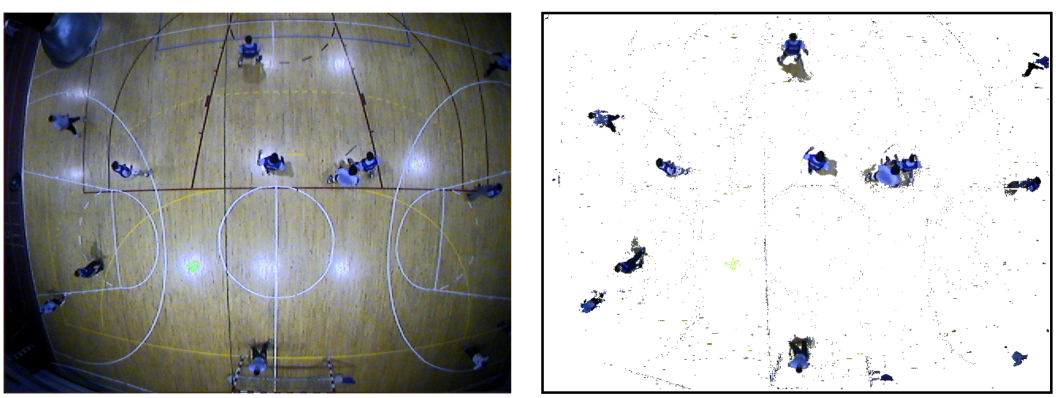


**Figure 4.17 – Conditional background subtraction before (left) and after (right)**

Usually performing background subtraction based on a static image can carry some problems due to light changes. Therefore an approach based on an adaptative background image was tested.

With this approach each time a new frame is analysed its content is used to perform an update into the background image. The first background image used is the one from the previous method.

For the update, both images (the empty field and the analysed one) are divided into a grid as exemplified in Figure 4.18. Each zone is subtracted from the background image, if the result is lower than a given value (determined experimentally) an average between the background image and the analysed image is performed and that section of the background image is updated. Otherwise the section will remain untouched.

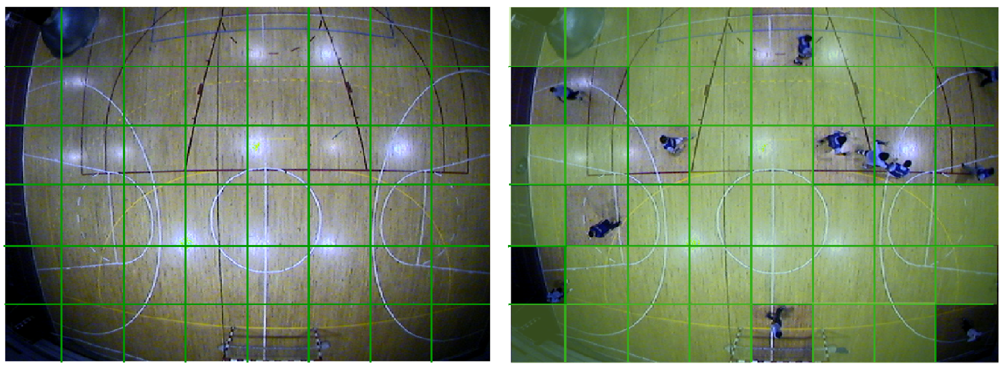


**Figure 4.18 - Adaptative background subtraction**

In this example the zones shaded in green are the ones updated into the background image.

It was expected an improvement with this approach, however the results didn't show such a significant one. This may be related to the fact that the video length was not enough to have light changes from the outer world, the artificial illumination is able to minimize the outer world light influence or the experimental value determined for the previous approach is large enough to also take care of these variations.

## 4.4.3 – Colour Detection

Once the team colours are defined and the background is subtracted the image can start being analysed, and the first step is to detect pixels that belong to each team colour. For that, the entire image is scanned and the colour value of each pixel is tested to see if it belongs to a given team colour.

This test consists on verifying if the entry on the colour lookup table is filled with a team colour or is empty. If it corresponds to a team colour then the colour of the pixel is replaced with the team colour identifier.

Remember that each team colour is defined in the colour lookup table and has a colour identifier. The following picture exemplifies this concept for two team colours: the red team, which is identified by the red colour (RGB: 255,0,0) and the green team identified by the green colour (RGB: 0,255,0). The RBG cube on the left is a 3D representation of the lookup table.

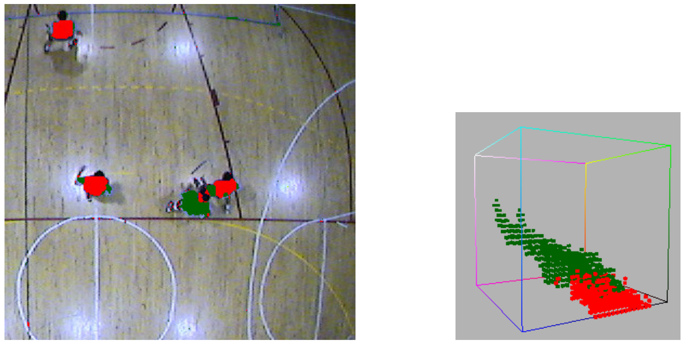


**Figure 4.19 - Two team colours**

The pixels are classified and updated into a given team or no team in a single scan passage.

## 4.4.4 – Pseudo-Mode Filter

Once the pixels are classified a pseudo-mode filter is applied in order to remove noise.

This filter is based on a 3x3 window and consists on analysing the 8 neighbours of a pixel and detecting if half or more of them belong to a given team colour. If this occurs then the pixel under analysis is replaced by that team colour identifier.

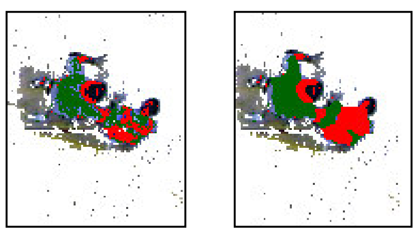


**Figure 4.20 - Image before applying the pseudo-mode filter (left) and after (right)**

With this filter is possible to reduce the noise and get a better colour blob definition. A colour blob corresponds to a portion of the image belonging to a given team colour. Usually a colour blob defines a player.

## 4.4.5 – Blob Aggregation and Characterization

At this point there is only information if a pixel belongs to a give team colour or no team it is still necessary to establish a relationship between pixels belonging to the same colour blob.

The algorithm responsible for establishing this relationship comprises two steps. The first step is based on a per line scan detection and the information is stored in a way similar to that of a run-length encoding using three parameters to define a blob of pixels belonging to a specific team colour: y, $x_{min}$ and $x_{max}$. Whenever a pixel belonging to a team colour is reached its x ($x_{min}$) and y values are stored and the subsequent pixels of the same line are checked to see if they also belong to the same team colour. Once reached the last pixel of the same colour or the end of the line the $x_{max}$ value is stored.

If the end of the line was not reached the scan continues in that line and if meanwhile another pixel of the same team colour is found and its distance to the $x_{max}$ pixel is inferior to a given value these new pixels are considered belonging to the first part. Diagram of Figure 4.21 exemplifies the algorithm.

As a result from the previous procedure a series of single lines are identified as belonging to a specific team colour and they still need to be joined to form a single blob, which represents the second step.

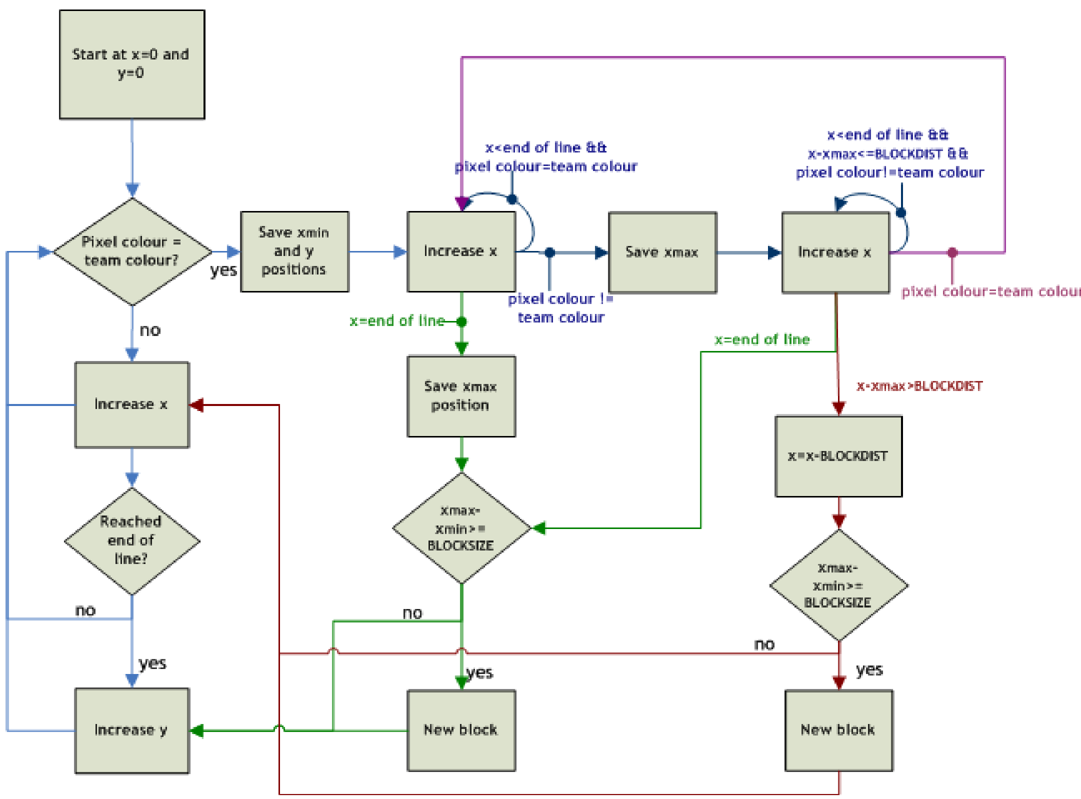



**Figure 4.21 - Detecting pixels belonging to a team colour**

If the distance between two of these lines is small and they belong to the same team colour then they are considered as being part of the same blob and are connected together.

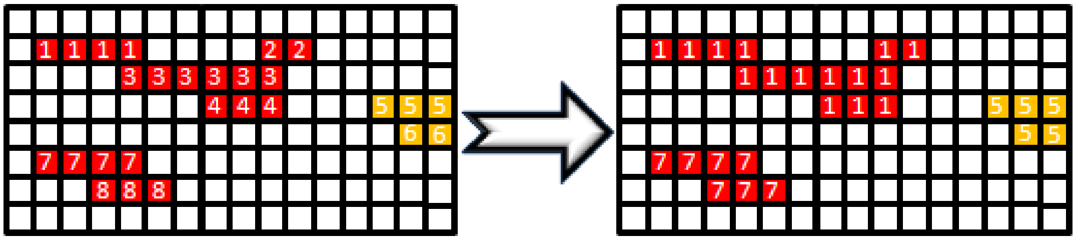



**Figure 4.22 - Single line aggregation to form a blob**

Once the blob aggregation is performed it is possible to characterize each blob, namely determine:

- minimum and maximum x and y
- blob area that corresponds to the number of pixels inside the blob that belong to the specific team colour
- rectangle that best fits the blob and centre of mass of the blob

Figure 4.23 show the results of the processing so far:

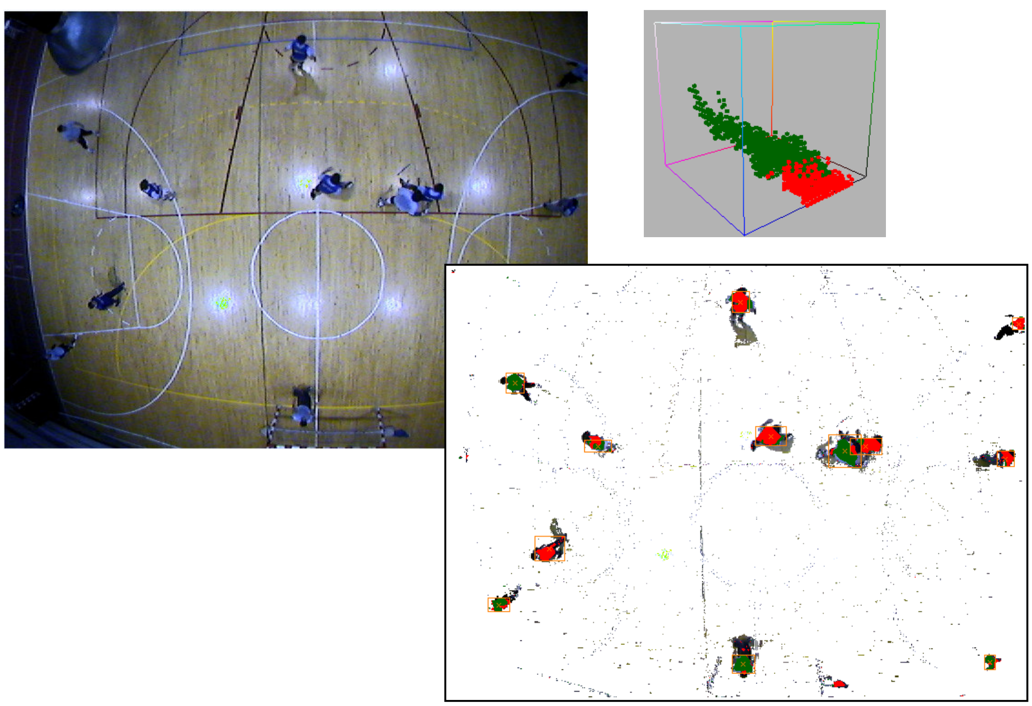


**Figure 4.23 - Result from the image processing system**

## 4.4.6 – Real World Transformation

So far the players are completely identified and the blobs characterization is performed, however it is still needed to perform the conversion into the real world coordinates.

This real world transformation comprises two parameters one, as seen from the previous images is the barrel distortion induced by the lens, that in this case is quite severe and the other corresponds to a scaling factor between the coordinates of the image (that are in pixels) into the coordinates of the real world (in centimetres).

In order to compensate for these two factors Equation (4.2) was considered.

$$\begin{bmatrix} x_R \\ y_R \end{bmatrix} = \left(1 + kr^2\right)\begin{bmatrix} S_x \times x_D \\ S_y \times y_D \end{bmatrix} \qquad \textbf{(4.2)}$$

Where the $\left(1 + kr^2\right)$ factor denotes the barrel effect distortion, which has a radial effect and its severity increases with the distance to the centre of the lens. k is the barrel distortion coefficient, r is the distance of the pixel to the centre of the lens, $(S_x,S_y)$ the scale factor, $(x_R,y_R)$ are the real world coordinates and $(x_D,y_D)$ are the distorted coordinates that correspond to the image coordinates.

In this equation the parameters that need to be found are the k and the $(S_x,S_y)$, so that the real world coordinates for any given pixel can be determined.

The $(S_x,S_y)$ can be easily calculated aided by the lines of the field and using the ratio between the real world distance and the pixels distance. For example, measuring the pixels used to define the inner area of the 6 meters line.

Using this method both on the x axis as well as on the y gives: S=(2.94cm,2.80cm).

For determining the k factor an approach based on the least square method was used. Let's ignore for now the scale factor (S) and using the y component of Equation (4.2) a line like the one in Equation (4.3) would be obtained:

$$\begin{aligned} &y_D = -y_D r^2 k + y_R \Leftrightarrow y = mx + b, \\ &where \;\; y = y_D, \;\; x = -y_D r^2, \;\; m = k \;\; and \;\; b = y_R \end{aligned} \qquad \textbf{(4.3)}$$

Applying the least square method results in:

$$\begin{aligned} m &= \frac{n\sum_{i=1}^{n}\left(-y_D r^2 \times y_D\right) - \left(\sum_{i=1}^{n} - y_D r^2\right)\times\left(\sum_{i=1}^{n} y_D\right)}{n\sum_{i=1}^{n}\left(-y_D r^2\right)^2 - \left(\sum_{i=1}^{n} - y_D r^2\right)^2} = k \\ b &= \frac{\left(\sum_{i=1}^{n} y_D\right)\times\left(\sum_{i=1}^{n}\left(-y_D r^2\right)^2\right) - \left(\sum_{i=1}^{n} - y_D r^2\right)\times\left(\sum_{i=1}^{n} - y_D r^2 \times y_D\right)}{n\sum_{i=1}^{n}\left(-y_D r^2\right)^2 - \left(\sum_{i=1}^{n} - y_D r^2\right)^2} = y_R \end{aligned} \qquad \textbf{(4.4)}$$

A closer look into the images collected from the camera shows lines that are good candidates to use for the barrel distortion coefficient calculation, namely the white line from the basketball field on the top of the image that goes from one side of the image until the other.

The k factor is calculated on the fly by clicking with the mouse along the chosen line, so even if the lenses are switched there is an easy way to find the value.

The following picture shows the results of applying the barrel "undistortion" equation.

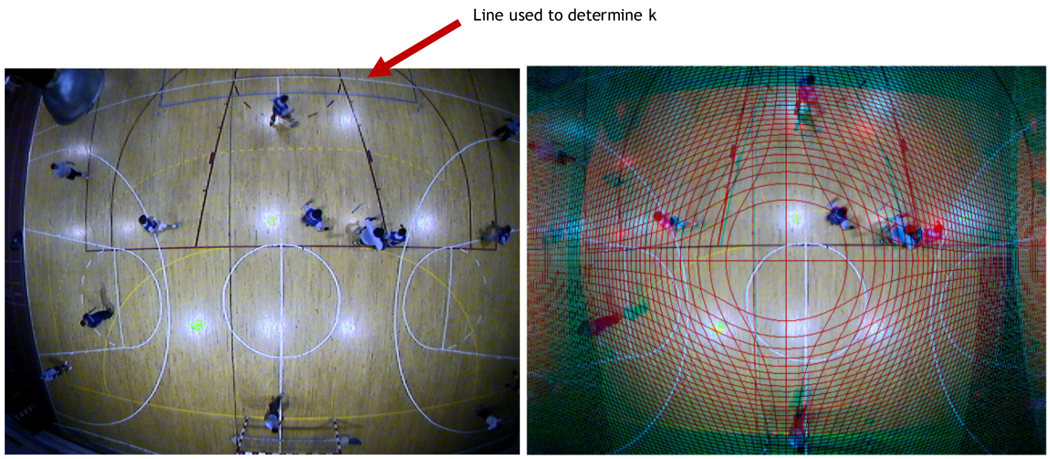



**Figure 4.24 - Barrel distortion**

For this "undistortion" 32 points from the referred line were considered and a $k=1.03\text{x}10^{-5}$ cm was obtained.

The transformation between image coordinates into real world coordinates is given by the following transformation:

$$\begin{bmatrix} x_R \\ y_R \end{bmatrix} = \left(1 + 1.03 \times 10^{-5} r^2\right) \begin{bmatrix} 2.94 \times x_D \\ 2.80 \times y_D \end{bmatrix} \tag{4.5}$$

In practice the entire image doesn't pass through an "undistortion" procedure, only the points of interest suffer this transformation.

## 4.4.7 – Player tracking

Once the players are detected it is possible to perform their tracking. As stated before the main parameters that characterize each player are the area and the centre of mass.

The method adopted is based on the past information and on defining a probable area around each player that defines his/her next position. This probable area takes into account the position and the maximum velocity a player can achieve. Figure 4.25 exemplifies this.

Once a new frame is picked, the processing system starts by identifying all the colour blobs and those that are identified as being a player are characterized.

The characterization parameters will be compared with the ones from the previously identified blobs and if they fit inside the probable area of one of those blobs then the new blob is assumed as being the sequence.

The player tracking algorithm keeps on memory a vector with the characteristics of every player for a limited period of time and makes use of this information to perform the comparison.

**Figure 4.25 - Player and next probable position area**

The theory behind this tracking algorithm is very simple but it proved to be functional and effective.

The final stage of the tracking algorithm consists on saving the positions and the identification of each player in a file (see Appendix B for an example). Each video has a specific associated file which would allow video tagging.

This file is a very useful source of information since it provides the bases to perform statistics such as the length and the time a player runs, the statistics of accelerations and velocities, among other things. This information is important for the sports teacher and the coach of the team in order to evaluate the level of physical intensity during the period under analysis (game or practice). Evaluation of these parameters may lead to conclusions like a player being season tired or other motives for abnormal performances that demand teacher/coach action. Such application to further help the teacher/coach is mentioned in the Future Work section.

To have a better idea of the importance of this file see Appendix C, where some straightforward statistics are performed using a simple excel sheet.

In future this file can also be the key to distinguish and recognize important team movements that often occur during offensive plays and further assist the teacher/coach with information from both teams.

## 4.5 – Summary

This chapter enumerates the software requirements and introduces some open source libraries that can be used in this kind of application, namely OpenCV and OpenFrameWorks.

Afterwards, a detailed explanation of the image processing techniques applied is given. The image processing is based on the colour blob notion and includes a stage for defining the team colours.

Three different approaches were tested for defining the team colours: the physical range, the physical flood and the grow colour. The usage of the last two approaches shows better results.

The next stage is the background subtraction where a conditional pixel subtraction is performed. The colour pixel identification follows the background subtraction and allows to determine the pixels in the image that belong to a given team colour. After that the pixels are aggregated to form a single blob and characterized.

The final stage consists on performing the transformation into the real world coordinates. Finally the tracking algorithm is explained.

# Chapter 5

# Results

This chapter presents the results achieved and gives a description of the test platform.

All the images present in this document were recorded with the Sony SNCDM110 IP surveillance camera and tests were conducted in the FADEUP's sports hall during basketball and handball training sessions of the Futebol Clube do Porto junior teams with training vests.

This camera was lent by Sony Portugal for a short period of time which only allowed for a few videos to be captured and not always under the best conditions.

## 5.1 – Image Quality

The camera used for the tests provides different configurations and one of the first tasks consisted on evaluating the more promising configurations and detecting the one that best suits the problem. A series of tests were conducted during a basketball training session.

All the images were captured using the MJPEG format because it provides higher video quality since it is independent of the motion in the image as stated in (32).

Of all the possible configurations the camera provides, the following were tested:

**Table 5.1 - Sony SNCDM110 configurations tested**

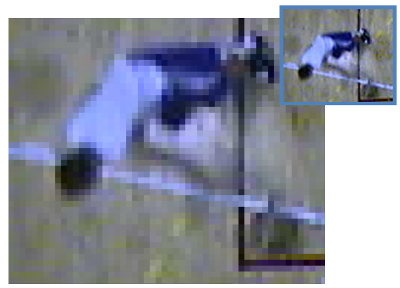
| *Quality (compression ratio)* | *Frame rate (fps)* | *Spatial Resolution (pixels)* | *Test #* |
|---|---|---|---|
| 1/30 | 15 | 1280x960 | 1 |
| | | 640x480 | 2 |
| | 30[4] | 640x480 | 3 |
| 1/6 | 15 | 1280x960 | 4 |
| | | 640x480 | 5 |
| | 30 | 640x480 | 6 |

[4] At 30fps the maximum resolution provided by the camera is 640x480 pixels

This evaluation consisted not only in analysing the image quality but also the rate of lost frames.

Table 5.2 summarizes the results obtained showing the number of captured frames, the lost frame rate and a sample of the images acquired for each configuration.

**Table 5.2 – SNC DM110 image quality test**

| *Test #* | *# total frames* | *% lost frame* | *Image quality* |
|---|---|---|---|
| 1 | 110 | 4.3 |  |
| 2 | 139 | 0 |  |
| 3 | 269 | 0.74 | |
| 4 | 52 | 60 |  |
| 5 | 134 | 1.5 |  |
| 6 | 186 | 34.7 | |

From the previous table it is possible to see that a higher resolution as well as a lower compression ratio results in a better quality image at the pixel level, however the better image quality implies an increment in the number of frames lost.

Taking into consideration the characteristics analysed it is also possible to see that this camera can only be used without loss of frames with a spatial resolution of 640x480 pixels, compression ratio of 1/30 and a frame rate of 15. A compression ratio of 1/6 for the same spatial resolution and image frequency implies a much better image quality with only 1.5% of frame loss.

## 5.2 – Player Detection and Tracking

The sample footage used to evaluate the algorithms and techniques developed is based on a video recorded during a handball training session of the Futebol Clube do Porto junior team, filmed with permission at the sports hall of the Faculty of Sports of the University of Porto on the 25th of April 2009.

As stated before the image capture not always occurred under the best conditions because during the short period the camera was available the sports hall was mostly occupied with students that don't wear specific equipment during classes. This handball training session was the best opportunity to shoot offensive/defensive game situations. However it still has some drawbacks namely the differences between the two teams equipments consist only in a simple vest.

According to the results obtained in the previous section the video has the following characteristics: spatial resolution of 640x480 pixels, frequency of 15 frames per second and despite an increase in the loss of frames a compression ratio of 1/6 was chosen because the gains on the side of image quality compensate the few frames lost.

A typical image in this video is shown below.

**Figure 5.1 - Example of a typical test image**

### 5.2.1 – Player Detection

Robust player detection plays a very important role since it establishes solid foundations where the tracking algorithm can build upon. Therefore the first tests were oriented to determine how robust the player detection is.

The tests consisted on analysing 900 frames (1 minute) of the captured video and determining how often the players are detected.

The identification of players from both teams is a very hard task due to the similarity of equipments, the differences between the two teams' equipments is only the vest the dark blue team wears above the same equipment of the light blue team.

In real game situation players have very different colour equipments and therefore each team is well differentiated in terms of colour.

Figure 5.2 shows the team colour definition used in this batch of tests (recall the theory behind the team colour definition stated in 4.4.1 – Team Definition). The green colour identifies the light blue team and the red colour the dark blue team. Due to the equipment similarity these two team colours have an interlaced region which makes it harder to identify players whose equipment colour is on this frontier.

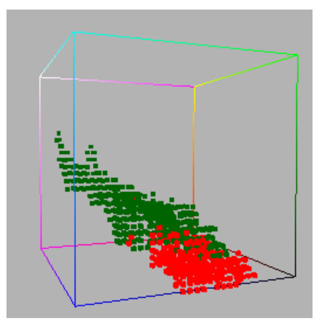


**Figure 5.2 - Team colour definition**

Table 5.3 presents the percentage of right detections for the highlighted players in Figure 5.3.

**Table 5.3 – Player detection**

| *Player* | *Detection (%)* |
|---|---|
| **A (goalkeeper)** | 76.2 |
| **B (dark blue team)** | 72.2 |
| **C (dark blue team)** | 88.0 |
| **D (light blue team)** | 93.3 |
| **E (dark blue team)** | 75.2 |

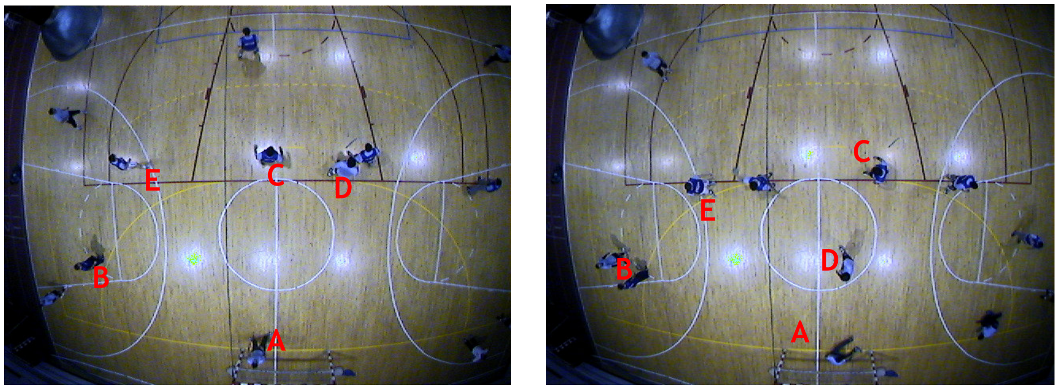



**Figure 5.3 – Player detection: beginning of the sequence (left) and end (right)**

Players B, C, D and E during this test sample occupy the following positions:

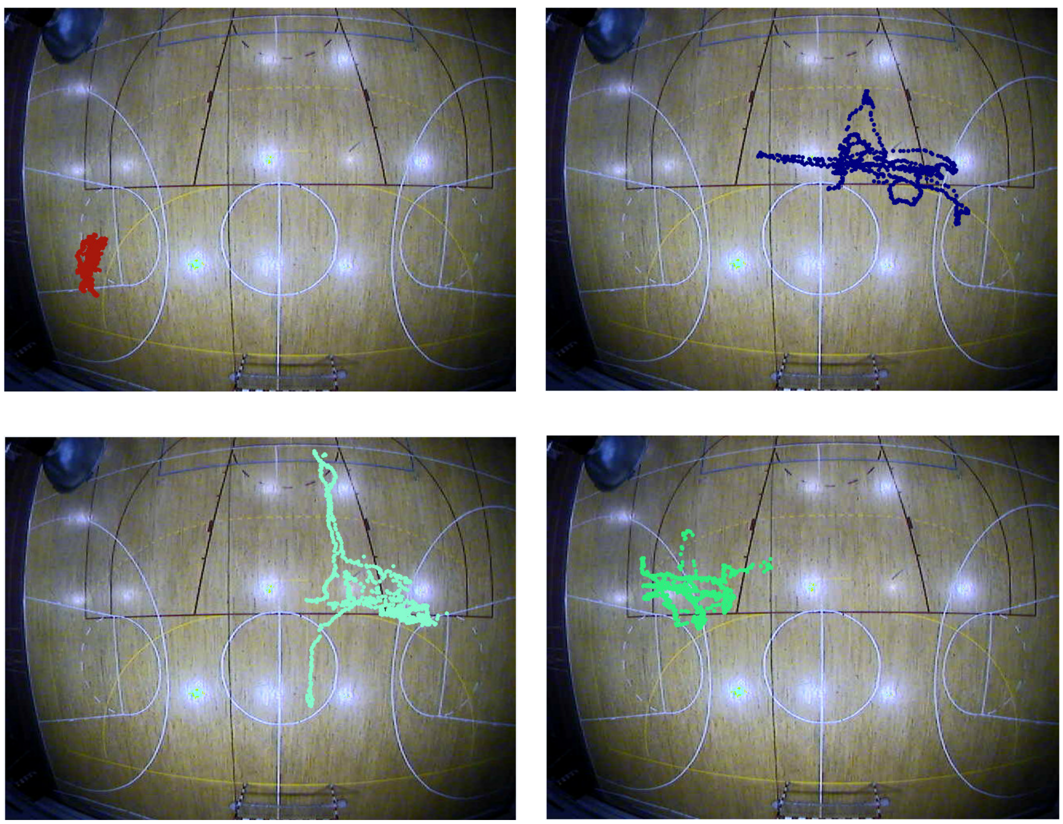


**Figure 5.4 – Detect player B (up - left), C (up - right), D (down – left) and E (down – right)**

The detection percentage of players is much better at the centre of the image because in this area the lenses produces better images and also due to a better illumination (there are some missing light bulbs on the sides of the field mainly near the goal).

Also if a player from the dark blue team occupies a flank position towards the camera then the most visible area of the equipment is the sleeve (of light blue colour) and therefore the player can be detected as belonging to the other team.

The lower detection of player B is explained because he stays most of the time in a very low light zone and in the extremities of the image where the lens doesn't produce such a good image. Section 3.4.4 – Lens Choice refers that a lens with higher dimensions than the sensor is an advantage and from these images it seems a good option.

Player C consistently stays in the centre of the image and with his backs towards the camera which allows good player detection.

A player belonging to the light blue team has good chances of being well detected because his entire equipment is of the same colour and that is what happens to player D.

Player E suffers from being of the dark blue team and passing the great majority of time with his flank towards the camera.

Even during player merging situations it is possible to identify the players as being different individuals. Figure 5.5 exemplifies two situations where the players are too close together and even so they can be properly differentiated.

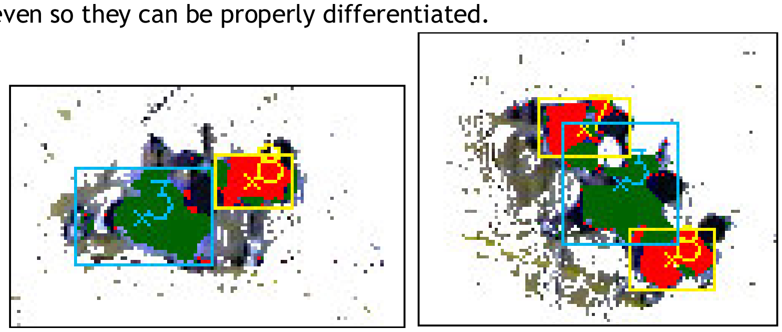


**Figure 5.5 – Correct identification during players' merging**

Here players' merging doesn't pose a problem because on the first case it happens between players of different teams and on the second despite having two players of the same team there is a player from the opposite team in the middle.

However when players of the same team get close enough to occur merging they are considered as being only one as shown in the following example.

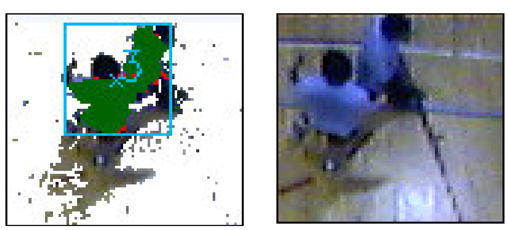


**Figure 5.6 - Merging between players of the same team**

The image on the right side shows clearly that the camera sees the sleeves from both players touching.

The most severe case of merging corresponds to occlusion and in this case if a player is completely hiding another the detection algorithm will only perceive the one that is visible.

## 5.2.2 – Player Tracking

The performance of the tracking algorithm is evaluated by the time it can keep the track of a player.

As stated before a good tracking algorithm is based on a solid player detection, therefore a lower detection rate implies a worse player tracking.

The following figures and tables show the performance of the tracking algorithm for players B, C, D and E from the previous section.

In the tables is possible to see for how long a player is followed. Each section corresponds to the period of time in which the tracking is performed correctly. A new section implies that the player track was lost. In the figures each colour identifies a section tracked properly. Although in reality the set of sections belongs to the same player the tracking algorithm sees each portion as belonging to a different player.

**Table 5.4 - Player B tracking**

| *Section* | *Duration (s)* | *Distance (cm)* |
|---|---|---|
| **Red** | 14.2 | 12.3 |
| **Green** | 14.4 | 8.1 |
| **Blue** | 16.2 | 9.7 |
| **Pink** | 12.2 | 8.3 |

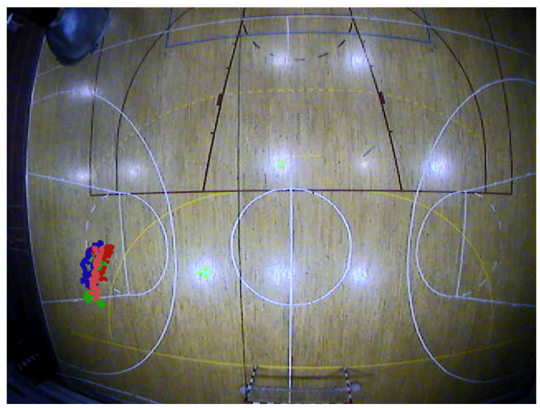


**Figure 5.7 - Tracking player B**

Table 5.4 and Figure 5.7 show the results of the tracking algorithm for player B. Initially the player is tracked in section red, but 14.2 seconds latter the player is not detected for a long enough period and his track is lost. During this sample footage the tracking algorithm lost this player 3 times.

**Table 5.5 - Player C tracking**

| *Section* | *Duration (s)* | *Distance (cm)* |
|---|---|---|
| **Red** | 16.6 | 23.2 |
| **Green** | 30.0 | 31.3 |
| **Blue** | 13.1 | 21.8 |

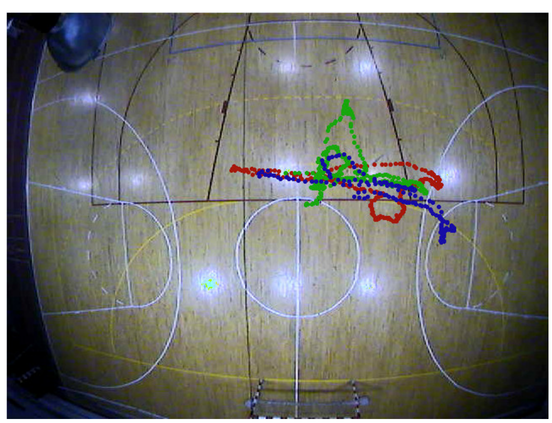


**Figure 5.8 - Tracking player C**

Player C had a very good detection rate and therefore the tracking algorithm behaves better only loosing the player 2 times and on the second section it is able to follow the player during 30 seconds.

**Table 5.6 - Player D tracking**

| *Section* | *Duration (s)* | *Distance (cm)* |
|---|---|---|
| **Red** | 27.5 | 44.5 |
| **Green** | 23.0 | 29.8 |
| **Blue** | 8.7 | 10.4 |

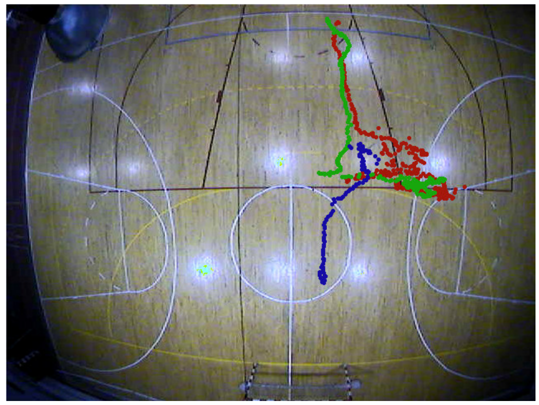


**Figure 5.9 - Tracking player D**

Player D can be tracked for a long distance in the first section 44.5 meters and like in the previous case the player is lost 2 times.

Player E was the one who showed worse results in terms of detection and that is reflected in the tracking algorithms that looses the player 5 times as can be seen by the following table and image.

**Table 5.7 - Player E tracking**

| *Section* | *Duration (s)* | *Distance (cm)* |
|---|---|---|
| **Red** | 2.5 | 3.5 |
| **Green** | 15.0 | 16.0 |
| **Blue** | 2.4 | 2.0 |
| **Pink** | 24.2 | 21.6 |
| **Purple** | 6.7 | 7.0 |
| **Yellow** | 4.8 | 3.1 |

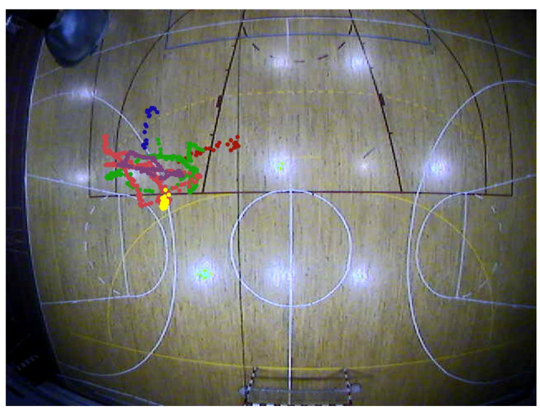


**Figure 5.10 - Tracking player E**

As stated before merging and occlusion can carry many troubles in a tracking algorithm, however when this happens between players from opposite teams the tracking algorithm has a good performance.

Figure 5.11 shows the result of the tracking algorithm during the right side merging of Figure 5.5. The images evidence that, despite a merging of several players, the algorithm is capable of identifying correctly the players afterwards.

Of course that during the merging some players are seen much bigger (example of player 3), but in the general sense it is able to detect all the players and position their centre of mass (cross inside the rectangle) in more or less the correct position.

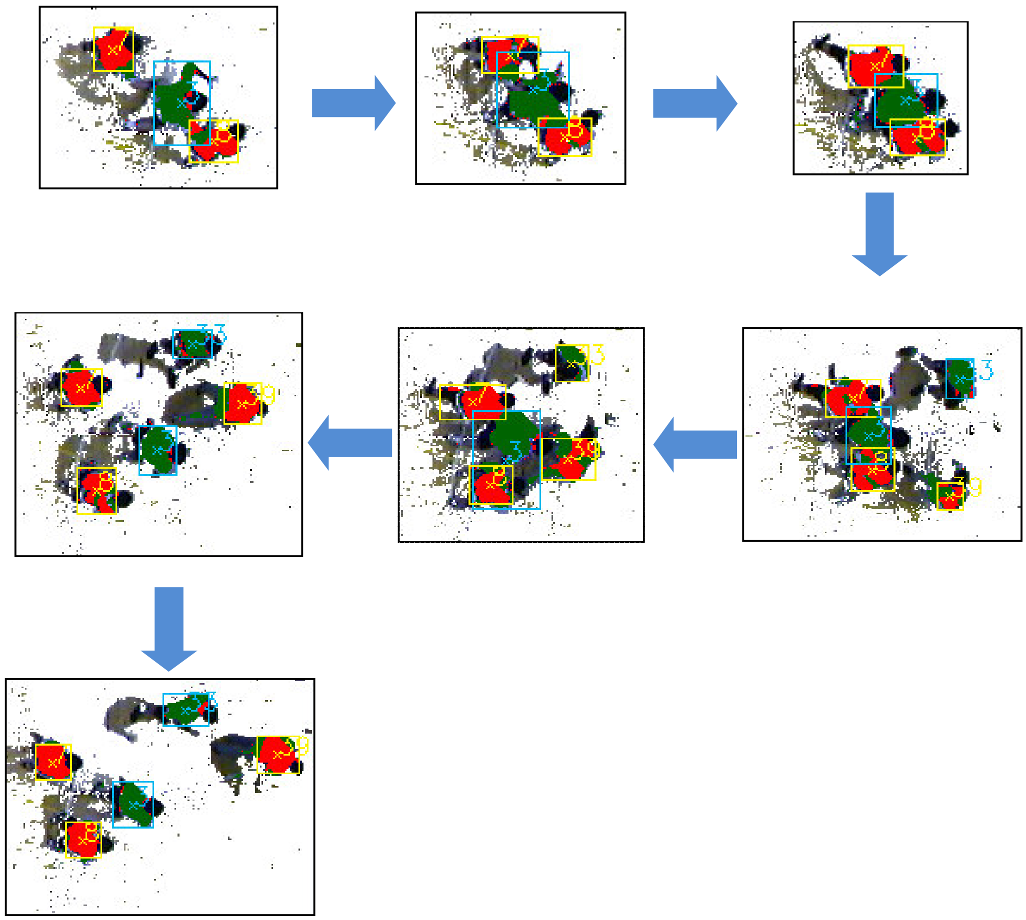


**Figure 5.11 - Tracking during a merging situation**

These tests were performed in a laptop computer with 1MB L2 cache and powered by an Intel T2130 processor running at 1.86GHz, under Windows Vista operative system and, in average, each frames takes around 65ms to be processed.

## 5.3 – Summary

This chapter refers to the results achieved and is divided in two main sections. The first section is intimately related to the camera and evaluates some possible configurations that include the following parameters: frame rate, spatial resolution and image quality that is translated into compression rate.

The second section is related to the player detection and tracking performance. Five players are taken as test subjects. Their detection rates are determined and range from

75.2% to 93.3%. These detection ratios would probably improve if the two teams were using official equipments that have very different colours.

Also the impact of merging and occlusion between players of the same and opponent teams is evaluated.

Finally the tracking algorithm is assessed. It is shown that the algorithm, depending on the performance on the detection phase, can track a player for as long as 30.0 seconds. It is also possible to verify that despite some merging situations the algorithm is able to track the players.

# Chapter 6

# Conclusions and Future Work

This is the final chapter of this document, where conclusions are drawn and the natural evolution steps that may be taken in future work are identified.

## 6.1 – Conclusions

This document presented the work developed towards the implementation of a system capable of analysing images from sports events. The main objectives were to study and develop methodologies for an engineering solution that could be able to deal with such a complex problem.

Taking into account the hardware requirements and the nature of the problem it was possible to find a solution based on a four GigEthernet digital camera system that can accomplish a good spatial resolution (deal with the size of the ball and lines and the high area under analysis) and maximize the number of sustained frames per second that, in turn, generates a huge amount of data transferred and processed.

In this architecture the cameras are fixed on the ceiling at strategic points that allow a good view of the field: avoid dead zones and provide an oblique view of the areas where most action happens.

The usage of GigEthernet cameras enables the cameras to be far away from the processor and therefore the spot chosen, the ceiling, doesn't pose cable length problems. These GigEthernet cameras were not available during the test phase.

All tests were performed using the Sony SNCDM110 IP surveillance camera lent, for a limited period of time, by Sony Portugal.

The software approach is based on two main systems:

- The Image Acquisition system that allows starting the image acquisition process and store the video.

- The Image Processing system responsible for analysing each frame of the video stream in order to detect and track the players.

Blob definition is on the base of the Image Processing system and each player is seen as a colour blob.

The first step towards the player identification consists on calibrating each team colour. Experience showed that a good colour calibration is a key factor to have a good players' detection. Three methods were tested and a combination of two techniques, physical flood and colour grow, produced the best results.

Afterwards a background subtraction based on a conditional pixel subtraction produced good results delimiting very clearly the areas of interest that correspond to the players.

Next to the pixel binning that determines if a pixel belongs to a given team, a pseudo-mode filter helped to reduce the noise and improve the player's identification.

After that each pixel is grouped to form a single colour blob to represent the respective player, ending the player identification phase.

Tests conducted using the mentioned IP surveillance camera to a sample footage of 60 seconds showed that most of the time players are correctly identified as being part of a team and that player detection based on pure colour blob notion achieves good results.

The best result achieved was 93.3% of correct identification and the worst 75.2%. Occlusion and merging between players of the same team seem to be the strongest factors penalizing the players' identification.

This is a good result taking into consideration the several artefacts the camera produces, the many light effects present (mirror of the lights, low illumination due to lack of lights, shadows of the players, and barrel effect among others) and the similarity between the equipments of both teams on the available sample footage.

Finally the last step, the player tracking, was based on defining a probable area around the previous position of the player using the centre of mass position and the maximum theoretical player velocity. The minimalistic presented tracking algorithm was able to correctly follow a player during 30 seconds even having some frames where the player was not detected.

Despite the test being conducted in a handball environment the application is generic and therefore should be able to analyse other sports images such as basketball or volleyball.

## 6.2 – Future Work

It is clear that there are many opportunities to improve and expand the work developed so far. Maybe one of the most important aspects to improve is the tracking algorithm so that its dependency on a good player detection is not so strong. Therefore Kalman filter techniques and artificial intelligence processes may be explored in order to improve the algorithm's performance.

It would also be interesting to implement the complete engineering solution and test it in a real game situation where the players have very distinct equipments (with official team colours).

Another aspect that would also deserve attention would be to develop an application that could provide the game statistics in a friendly format for the end user.

# References

1. *CMOS vs. CCD: Maturing Technologies, Maturing Markets.* **Litwiller, Dave.** Waterloo, Ontario, Canada : Photonics Spectra, 2005.

2. **Litwiller, Dave.** Photonics Spectra. *Lauring Publishing Co. Inc.* January 2001.

3. Trichromacy - Wikipedia, the free encyclopedia. [Online] Wikipedia. [Cited: 2 June 2009.] http://en.wikipedia.org/wiki/Trichromacy.

4. Color. [Online] MSDN Microsoft Developer Network. [Cited: 28 May 2009.] http://msdn.microsoft.com/en-us/library/aa511283.aspx.

5. hue - Definition from the Merrian-Webster Online Dictionary. [Online] Merriam Webster. [Cited: 2 June 2009.] http://www.merriam-webster.com/dictionary/hue.

6. HSV color space. [Online] Knowledgerush. [Citação: 15 de June de 2009.] http://knowledgerush.com/kr/encyclopedia/HSV_color_space/.

7. **Travis, David.** *Effective Colour Displays:Theory and Practice.* London, Uk : Academic Press, 1991.

8. **Zafrilla, Javier Marín.** PCE-RGB colour meter. [Online] [Cited: 28 May 2009.] http://www.industrial-needs.com/technical-data/colour-meter-pce-rgb.htm.

9. **Gonzalez, Rafael and Woods, Richard.** *Digital Image Processing 2nd Edition.* New Jersey, USA : Prentice Hall, Inc., 2002. 0-201-18075-8.

10. YUV/RGB conversion formulas: Information from Answers.com. [Online] Copyright © 2009 Answers Corporation. [Cited: 16 July 2009.] http://www.answers.com/topic/yuv-rgb-conversion-formulas.

11. **Benedetto, Franscesco, Giunta, Gaetano and Neri, Alessandro.** A new color space domain for digital watermarking in multimedia applications. *Proceedings of the IEEE International Conference on Image Processing, 2005.* 2005. Vol. 1, pp. 249-252. 0-7803-9134-9.

12. **Plataniotis, Konstantinos N. and Venetsanopoulos, Anastasios N.** *Colour Image Processing and Applications.* Germany : Springer, 2000. 3540669531.

13. **Liu, Jia, et al.** Automatic player detection, labeling and tracking in broadcast soccer video. *Pattern Recognition Letters.* New York, USA : Elsevier Science Inc., 2009. Vol. 20, 2, pp. 103-113. 0167-8655.

14. **Dearden, Anthony, Demiris, Yiannis and Grau, Oliver.** Tracking football players movement from a single moving camera using particle filter. *Proceedings of the 3rd European Conference on Visual Media Production(CVMP).* London, UK : IET Press, 2006. pp. 29-37.

15. **Xu, Ming, Orwell, James and Jones, Graeme A.** Tracking football players with multiple cameras. *in Proceedings of the International Conference on Image Processing 2004.* 2004. Vol. 5, pp. 2909-2912. 1522-4880.

16. ProZone. [Online] Prozone. [Cited: 15 June 2009.] http://www.prozonesports.com.

17. Sport Universal Process. [Online] SUP 2004. [Cited: 15 June 2009.] http://213.30.139.108/sport-universal/uk/amiscopro.htm.

18. SportsCode, leading edge sports Video Analysis Software. [Online] Sportstec. [Cited: 15 June 2009.] http://www.sportstec.com/Products_Sportscode.htm.

19. Orad TrackVision enhancement solution industry standard for live football productions. [Online] Shkalim Internet Innovations & Compix. [Cited: 15 June 2009.] http://www.orad.co.il/en/page.asp?id=283.

20. Trakus>>>Digital Sports Information. [Online] TKS Inc. [Cited: 15 June 2009.] http://www.trakus.com.

21. **Pers, Janez and Kovacic, Stanislav.** Computer vision system for tracking players in sports games. *Proceedings of the First International Workshop on Image and Signal Processing and Analysis.* Pula, Croatia : s.n., 2000. pp. 177-182. 953-96769-2-4.

22. **Liu, GuoJun, et al.** Hierarchical model-based Human Motion Tracking Via Unscented Kalman Filter. *Computer Vision, 2007, IEEE 11th International Conference.* October 2007. pp. 1-8.

23. **Needham, Chris J. and Boyle, Roger D.** Tracking multiple sports players through occlusion, congestion and scale. *CiteSeerX - Scientific Literature Digital Library and Search Engine (United States).* 2001.

24. **Browning, Brett and Veloso, Manuela.** Real-time, adaptive color-based robot vision. *In Proceedings of IROS'05.* Canada : s.n., August 2005.

25. **Sousa, Armando.** Arquitecturas de Sistemas Robóticos e Localização em Tempo Real Através de Visão. *PhD thesis, Engineering Faculty of Oporto's University.* Portugal : s.n., July 2004.

26. **Moreira, António, Sousa, Armando and Costa, Paulo.** Vision Based Real-Time Localization of Multiple Mobile Robots. *3rd International Conference on Field and Service Robotics.* June 2001. pp. 103-106.

27. **D'Andrea, Raffaello, et al.** Big Red: The Cornell Small League Robot. *Lecture Notes in Computer Science*. Berlin, Germany : Springer Berlin / Heidelberg, January 2000. pp. 89-98. 978-3-540-41043-0.

28. http://en.wikibooks.org/wiki/Adventist_Youth_Honors_Answer_Book/Arts_and_Crafts/Digital_Photography. [Online]

29. Focal length - Wikipedia, the free encyclopedia. [Online] Wikipedia. [Cited: 2 June 2009.] http://en.wikipedia.org/wiki/Focal_length.

30. Welcome - OpenCV Wiki. [Online] Wikipedia. [Cited: 12 February 2009.] http://opencv.willowgarage.com/wiki/.

31. **Bradski, Gary and Kaebler, Adrian.** *Learning OpenCV Computer Vision with the OpenCV library*. s.l. : O'Reilly, September 2008.

32. **On-Net Surveillance System Inc.** *MJPEG vs MPEG4 Understanding the differences, advantages, disadvantages of each compression technique*. Suffern, NY : s.n.

33. *Automatic Detection and Tracking of Handball Players*. **Santiago, Catarina, et al.** Porto, Portugal : Taylor and Francis, 2009.

# Appendix A

# Application Interface

This appendix presents images of the software application developed. Each of the referred systems (Image and Acquisition) was placed in a specific tab.

The first image reports to the Image Acquisition system that corresponds to tab "Record Video" and includes fields to define the camera IP address and the name of the video file.

There are also two buttons available that allow start and stop recording a video stream.

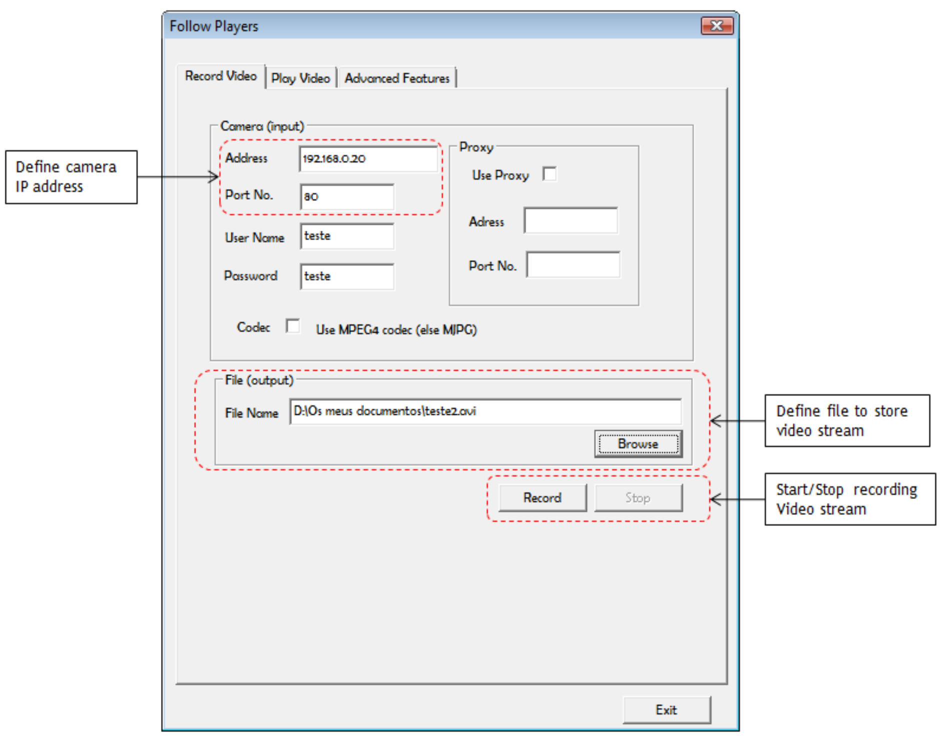



**Figure A.1 - Image acquisition graphical interface**

The Image Processing system is more complex and rests in the "Play Video" tab. This tab provides fields to define the video to be processed and the empty field image used for the background subtraction.

Several checkboxes and radio buttons allow calibrating each team colours according to the three methods explained in 4.4.1 – Team Definition.

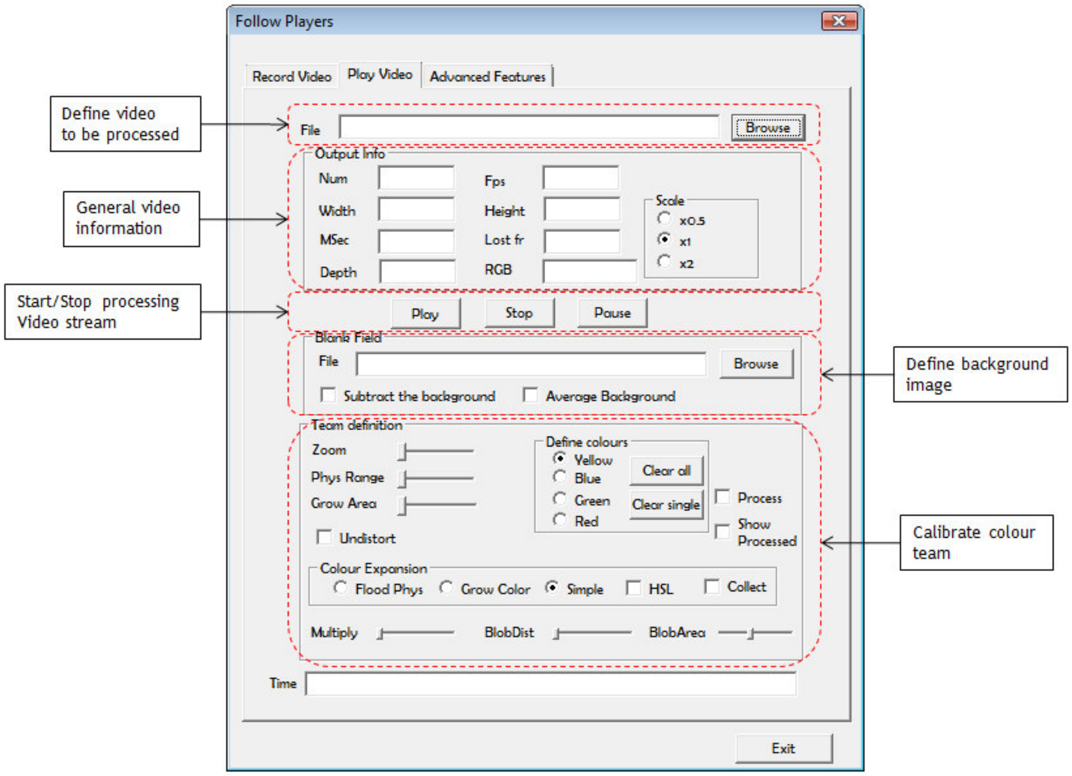



**Figure A.2 - Image processing graphical interface**

There is also an “Advanced Features” tab where it is possible to calculate the barrel distortion coefficient, define a region of interest and load or save a file with the team colour calibration.

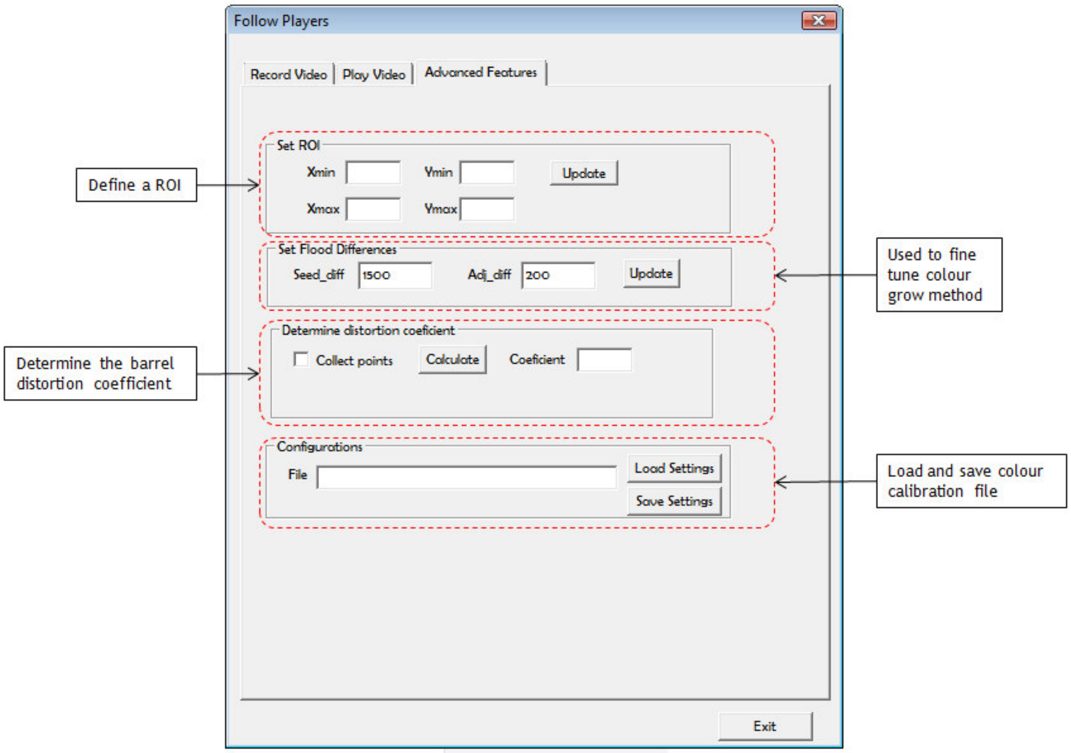



**Figure A.3 – Advanced Feature graphical interface**

# Appendix B

# Colour Calibration and Players' Position Files

The player position file contains the player number (BlbNumb), the team colour to which he belongs and the centre of mass position.

```
BlbNumb:0    colour:2    xcm:918     ycm:106
BlbNumb:1    colour:2    xcm:1729    ycm:110
BlbNumb:2    colour:2    xcm:165     ycm:311
BlbNumb:3    colour:2    xcm:1248    ycm:799
BlbNumb:4    colour:2    xcm:433     ycm:803
BlbNumb:5    colour:2    xcm:208     ycm:982
BlbNumb:6    colour:3    xcm:1781    ycm:722
BlbNumb:7    colour:3    xcm:969     ycm:836
BlbNumb:8    colour:3    xcm:1323    ycm:835
BlbNumb:9    colour:3    xcm:1812    ycm:1203
BlbNumb:10   colour:3    xcm:887     ycm:1245
---------------
BlbNumb:0    colour:2    xcm:920     ycm:106
BlbNumb:1    colour:2    xcm:1729    ycm:113
BlbNumb:2    colour:2    xcm:159     ycm:309
BlbNumb:3    colour:2    xcm:1236    ycm:784
BlbNumb:4    colour:2    xcm:442     ycm:807
BlbNumb:5    colour:2    xcm:208     ycm:983
BlbNumb:8    colour:3    xcm:1324    ycm:832
BlbNumb:9    colour:3    xcm:1815    ycm:1198
BlbNumb:10   colour:3    xcm:891     ycm:1244
---------------
```

**Figure B.1 - Excerpt of the players' position file content**

The colour calibration file consists on a sequence of numbers that define for each entry in the colour lookup table the team (-1 represents no team).

```
-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;3;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;3;3;3;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;3;-1;3;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;2;3;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
3;3;2;3;2;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;3;2;2;2;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;2;2;2;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;2;-1;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;-1;-1;2;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;-1;-1;-1;2;2;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;-1;
```

**Figure B.2 - Excerpt of the colour calibration file content**

# Appendix C

# Statistics

The players' position and velocity can be easily calculated using a simple excel sheet. Figure C.1 exemplifies and gives the formulas used to achieve the results of section 5.2.2 - Player Tracking.

| | |
|---|---|
| Time per frame | 0,07 |
| Lenght per frame (LPF) | $\sqrt{(xcm_n - xcm_{n-1})^2 + (ycm_n - ycm_{n-1})^2}$ |
| Velocity | $\frac{LPF_n - LPF_{n-1}}{Time\ per\ frame}$ |

| Blobnumber | Colour | xcm (cm) | ycm (cm) | LPF (cm) | Total Time(s) | Velocity(m/s) |
|---|---|---|---|---|---|---|
| BlbNumb:7 | colour:3 | 969 | 836 | | 0,07 | |
| BlbNumb:7 | colour:3 | 986 | 830 | 18,028 | 0,13 | 2,70 |
| BlbNumb:7 | colour:3 | 997 | 830 | 11,000 | 0,20 | 1,65 |
| BlbNumb:7 | colour:3 | 1026 | 835 | 29,428 | 0,27 | 4,41 |
| BlbNumb:7 | colour:3 | 1045 | 838 | 19,235 | 0,33 | 2,89 |
| BlbNumb:7 | colour:3 | 1072 | 840 | 27,074 | 0,40 | 4,06 |
| BlbNumb:7 | colour:3 | 1090 | 843 | 18,248 | 0,47 | 2,74 |
| BlbNumb:7 | colour:3 | 1104 | 849 | 15,232 | 0,53 | 2,28 |
| BlbNumb:7 | colour:3 | 1123 | 853 | 19,416 | 0,60 | 2,91 |
| BlbNumb:7 | colour:3 | 1136 | 856 | 13,342 | 0,67 | 2,00 |
| BlbNumb:7 | colour:3 | 1149 | 856 | 13,000 | 0,73 | 1,95 |
| BlbNumb:7 | colour:3 | 1169 | 861 | 20,616 | 0,80 | 3,09 |
| BlbNumb:7 | colour:3 | 1190 | 862 | 21,024 | 0,87 | 3,15 |
| BlbNumb:7 | colour:3 | 1206 | 861 | 16,031 | 0,93 | 2,40 |
| BlbNumb:7 | colour:3 | 1225 | 861 | 19,000 | 1,00 | 2,85 |
| BlbNumb:7 | colour:3 | 1251 | 865 | 26,306 | 1,07 | 3,95 |
| | | | | Total lenght (cm) | Total Time(s) | Max velocity |
| | | | | 286,979 | 9,07 | 4,41 |

**Figure C.1 - Calculate players statistics using an excel sheet**