# Basketball Player Identification by Jersey and Number Recognition


by

Zubaer Ahammed

16141028


A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering

Inspiring Excellence


DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BRAC UNIVERSITY


July 2018

"I hereby declare that the work in this thesis is my own work except for quotations and summaries which have been duly acknowledged."

Signature : ___________________________

Date : ___________________________

Name : Zubaer Ahammed

ID : 16141028

Supervised by: ______________________

: Professor Md. Haider Ali

# Contents

# Image Contents

# Abstract

We are interested in the problem of automatically tracking and identifying players in sports video. While there are many automatic multi-target tracking methods, in sports video, it is difficult to track multiple players due to frequent occlusions, quick motion of players and camera, and camera position. We propose tracking method that associates tracklets of a same player using results of player number recognition. To deal with frequent occlusions, we detect human region by level set method and then estimates if it is occluded group region or unconcluded individual one. This Thesis paper gives a brief description about each paper/technology in the field of ―Player Detection and Mapping Techniques in Sport Videos.

Recognition of players in pictures of sporting events is an approachable but tough task. In the case of an NBA game, this task can be accomplished by compartmentalizing the job. By characterizing jersey color with MAP detection, isolating the jerseys and numbers, and using template matching, we can make a max-effort algorithm that identify as many players as possible.


# Proposed Method

In this Thesis, authors addressed the problem of robust multi-target tracking using hybrid information from both the image and field domains. When compared to other methods of object/player tracking-by-detection methods which rely on resolution of image/target for training a object detector, this method works effectively. Proposed method uses field domain and player trajectories to detect and predict the player motion. Particle filter framework [1] is used to guide the tracking process. In particle filter frame work the predicting distribution is approximated by a finite sample N. For observation likelihood multi-color observation model based on HSV color and a gradient based shape model (HOG) histograms.

# Keywords

1. Jersey color
2. in game picture
3. RGB color space
4. Template matching



# Acknowledgement

I would like to express my sincere gratitude to my supervisor, Professor Md. Haider Ali for his support, guidance and invaluable instructions which helped me as well as encouraged me to complete the thesis.

# Chapter 1

## Introduction

Image Processing is a valuable tool that is applied on media presenting sports frequently. One well known system that is used frequently is the 1st & ten system used to generate yellow lines symbolizing the 1st down barrier in football in real time during a game. The system is also utilized for applications like advertisements in unused sub-windows, and depicting race car position for NASCAR races.

Player Identification is an application of image processing that is a topic of interest. The idea to find and recognize players' numbers was explored in [1]. The authors manipulated the HSV characteristics of the team jersey to isolate them. Then the characteristics of the image and sport are manipulated to isolate the number for image processing and identification.

Figure 1. Result of MAP detection

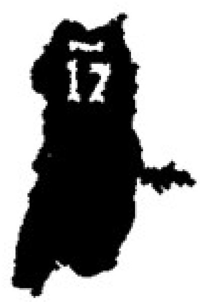


Figure 2. Result after additional processing

In [2]. We see a different application trying to identify digits in an image: the authors try to identify cars by reading the license plates. Morphological Image Processing is incorporated to isolate the numbers from the rest of the image, and the statistical method of correlation is used to detect the digits with an increased recognition success rate over [1].

Previous documentation on color properties provides motivation for the framework of this paper. In the following thesis, [3], I investigates the viability of using color detection in MATLAB to aid the visually impaired. He mentions the idea of recognizing certain colors, which inspired the idea to recognize jersey color by RGB value.

In this paper, we investigate the viability of a player detection system that has knowledge of the visual characteristics of an in-game image. We will use MAP detection to isolate jersey regions in the image, image segmentation to further process the image, and an OCR based number detection method to guess the number of a player.

The input to our algorithm is an image and knowledge of the team and jersey type, and the output will be the input image with the names of the players.

## 1.1 Motivation of Automated Sports Video Analysis

The Thesis on tracking systems in sports videos provides a tool for understanding intentional activities and can make a direct impact on sports, business and society by concretely an acting applications that enhance training methods as well as viewing practices for the general public. Applications for tracking identities in sports videos are manifold, as they prove advantageous for all participants of sports events, namely coaches, judges, scientists and spectators. We will point out possible applications and their likely impact, categorized according to the di errant bene carries.

## 1.2 Related Work

Existing approaches for automatic player identification in broadcast soccer videos can be categorized in two groups: One performing face recognition on close-up shots (not overview shots) in various types of sports videos, while other approaches rely on jersey number recognition. For the latter group, no approach is known to operate on soccer overview shots. They either operate on other sports where the resolution per player is higher (e.g. in basketball, or they perform on close-up shots, where jersey numbers are better readable and face recognition is feasible.

## 1.3 Challenges

Every tracking system for sports video analysis faces a number of technical challenges inherent in the problem and the domain of interest. Multiple interacting targets must be tracked concurrently, while occlusions of single protagonists occur frequently and on purpose, as interactions are part of the game. The motion of human players is complex and hitherto unknown for real competition scenarios (one aim of the tracking system is to investigate the typical motions of these). Hence, the position of unseen athletes can be predicted only for a limited time horizon, which hampers the processing of cut broadcasted material. Despite good visual discrimination between different teams, athletes of the same club are hardly distinguishable, which exacerbates their re-identification after an intermission of the video stream. Although identifiers like jersey numbers are attached to the players, their usage is unreliable as they are mostly facing away from the camera or appear covered or distorted in the video image.

In basketball players are detected using a deformable parts model (DPM), after which an exact localization of the jersey number is performed. Then, normalization, followed by thresholding and calculating the correlation between the digits and digit templates is applied. In, player identification is performed in overview shots by employing SIFT features for face recognition.

# Chapter 2

## Jersey Recognition

Since we will know the teams that will be playing in the picture, we will have up to 4 possible jersey colors present at the games (each team having a home and away jersey). In basketball, since the jerseys are typically one color without any stripes or advertisements, we can characterize them as a certain color.

## 2.1 RGB

This paper gives an approach to recognize colors in a two - dimensional image using color thresh - holding technique in MATLAB with the help of RGB color model to detect a selected color by a user in an image. The methods involved for the detection of color in images are conversion of three dimensional RGB image into gray scale image and then subtracting the two images to get two dimensional black and white image, using median filter to filter out noisy pixels, using connected components labeling to detect connected regions in binary digital images and use of bounding box and its properties for calculating the metrics of each labeled region. Further the color of the pixels is recognized by analyzing the RGB values for each pixel present in the image. The algorithm is implemented using image processing toolbox in MATLAB.

We attempted finding a loose range of RGB values that would characterize the jerseys in the pictures. The proposed algorithm would go through every pixel, and check whether its RGB values fell into the respective acceptable ranges. If all three of its RGB values did not fall in

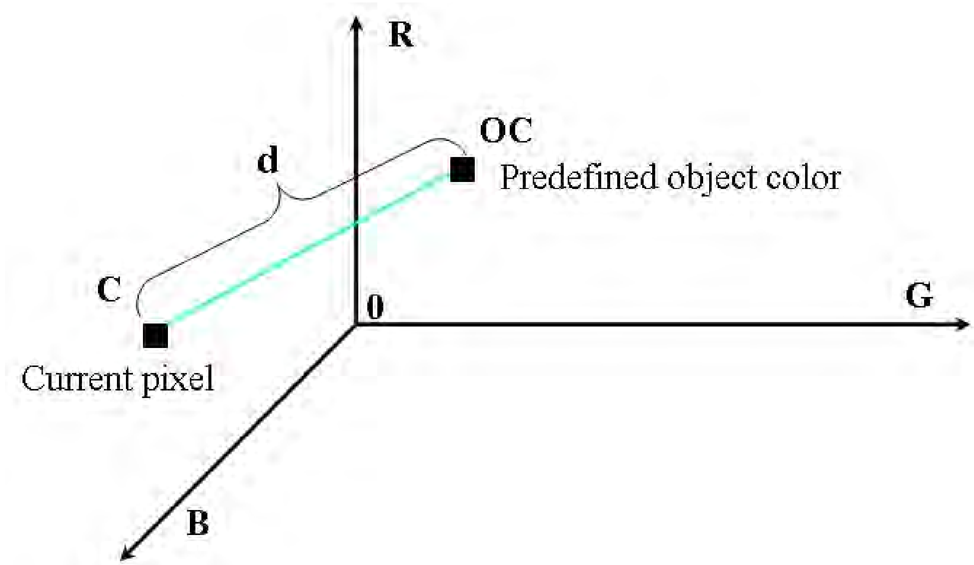



Figure 3. Direct comparing of RGB colors

The range of acceptable values, then the pixel got assigned RGB values of 0, to denote that it was not a jersey. Unfortunately, this algorithm was far too simplistic to succeed. Even with the removal of the crowd and its wide range of colors, there were still some false positives associated with this method. Additionally, jerseys did not always get identified by the algorithm. The algorithm was not robust for the different varieties of pictures that could possibly get taken, because of variables like angle, brightness, cropping, etc.

## 2.2 MAP Detector

Map detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, jersey, buildings, or cars) in digital images and videos.

MAP is the metric to measure the accuracy of object detectors like Faster R-CNN, SSD, etc. It is the average of the maximum precisions at different recall values.

In object detection, evaluation is non trivial, because there are two distinct tasks to measure:

1. Determining whether an object exists in the image (classification)
2. Determining the location of the object (localization, a regression task).

Furthermore, in a typical data set there will be many classes and their distribution is non-uniform .So a simple accuracy-based metric will introduce biases. It is also important to assess the risk of misclassifications. Thus, there is the need to associate a "confidence score" or model score with each bounding box detected and to assess the model at various level of confidence.

To circumnavigate that issue, we looked into designing MAP detectors for every jersey. We found five training images per jersey for our MAP detector. We attempted to find jerseys in different lighting conditions to fully capture a wide range of RGB values for the maximum chance of identifying a jersey. The regions with the color of interest were painstakingly cropped. Then, we train our MAP detector using our training images and derived masks. To isolate jerseys in an image, we will apply our MAP detector to the image. This algorithm ends up being much more robust than the previous simplistic one.

Ideally, this will produce a mask of the image, with the jerseys being white and the rest black, as shown in Figure 1. Realistically, false positives are impossible to avoid.

Therefore, we will have to do more processing to obtain our jerseys without any noise from falsely identified jersey pixels. We can use the regional properties of the noise to discriminate against them. Each basketball team is allowed five players on the court. With that logic, we will find the regions with the five biggest areas, and eliminate those that do not fit that criterion. This will leave us with some noise, and our jerseys of interest.

# Chapter 3

## Number Recognition

Due to the steps we have taken previously, at this point we have a mask of the jerseys and some additional noise. We want to make sure that the numbers are isolated from the rest of the image, with no connected edges. We apply an erosion with a small disk structuring element to create some separation between the number and other elements of the picture.

To recognize the player, we need to try to recognize the number on the jersey also. We will look at each player individually, and compute which player it has the greatest chance of being. To isolate one player, we loop through the centroids in order of decreasing area. We look at each "region," or jersey by itself, but setting the rest of the regions to 0, and arrive at a mask like that in Figure [2]. Then we look to isolate the number of the player. To do that, we invert the mask, and then remove the largest area, which will be the background. We will be left with the number of the player and extraneous details from the jersey, which is usually the Lakers logo or the player's name, depending on whether the player is facing forwards or backwards during the photo.

## 3.1 Template Matching

The current algorithm uses the following templates to compute features: These prototypes are scaled in vertical and horizontal distance.

Figure 4. Templates used to compute features by the face and detection algorithm

Template matching is a 'brute-force' algorithm for object recognition. Its working is simple: create a small template (sub-image) of object to be found, say a jersey. Now do a pixel by pixel matching of template with the image to be scanned for, placing (center of) the template at every possible pixel of the main image. Then, using a similarity metric, like normalized cross correlation, find the pixel giving maximum match. That is the place which has a pattern most similar to your template (jersey).

### 3.1.1 Convert to Binary Images

The process of converting the color image into black and white image is called a binary image. This method is based on various color transforms. According to the R, G, B value in the image, it calculates the values of grey scale and also obtains the grey image [3].Template matching technique can be easily performed on grey images or edge images.

### 3.1.2 Find Character Boundaries

This step finds the character boundaries by using template image. Template image is a small portion of an input image; it is used to find the template in the given search image. Template matching technique is used to find character boundaries. The work flow of the template matching is illustrated in figure [5].

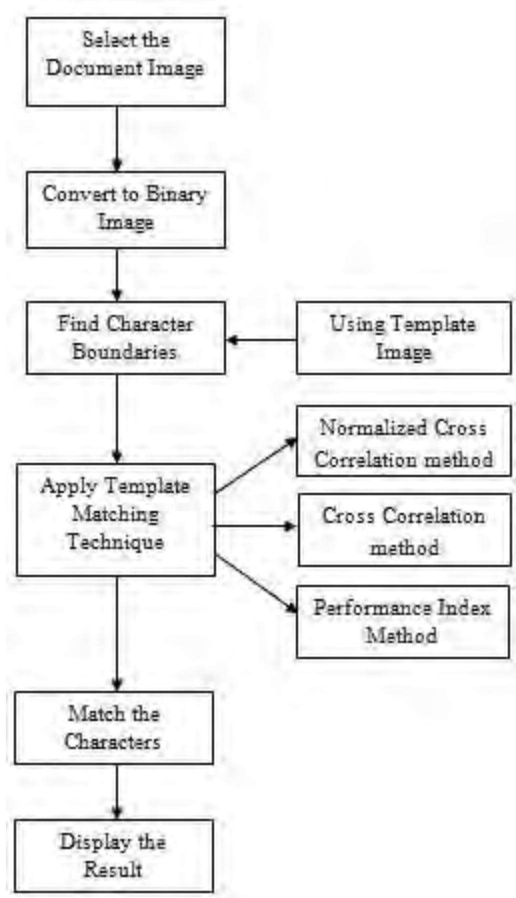



Figure 5. Workflow of the Template matching

This is just the brief description of template matching. You can find proper derivation of normalized cross correlation (ncc) in standard texts on Image processing.

## 3.1.3 Cross Correlation

The cross-correlation template matching is motivated by the distance measure (squared Euclidean distance) [17], [18]

$$d^2_{f,t}(u,v) = \sum_{x,y}[f(x,y) - t(x-u, y-v)]^2$$

Equ.1

Where f is the input image and t is the template image, the sum is over x, y under the window containing the feature t positioned at (u, v). In the expansion of $d^2$

$$d^2_{f,t}(u,v) = \sum_{x,y}[f^2(x,y) - 2f(x,y)t(x-u,y-v) + t^2(x-u,y-v)]$$

Equ.2

The term $\sum t^2(x-u, y-v)$ is constant. If the term $\sum f^2(x, y)$ is approximately constant, then the remaining cross-correlation term.

$$c(u,v) = \sum_{x,y} f(x,y)t(x-u,y-v)$$

Equ.3

is a measure of the similarity between the image and the feature.

There are some obvious flaws in template matching as a tool for object recognition. First of all, if you don't have any matching object in image, you will still get a match, corresponding to max of ncc. Also, this matching is affine variant: a change in shape/size/shear etc. of object w.r.t. template will give a false match. Thirdly, the calculation of ncc is highly inefficient computationally. Template matching is therefore rarely used. Some 'object descriptors' are used instead.

Now we have what we need to recognize the player. Our first idea for doing so is to use template matching. We constructed five templates, one for every possible player. We then found the height of the number on the jersey. Using this, we resized each of the templates to match the height of the number. After resizing the templates, we applied a convolution on the mask of the number with each of the templates. We then observed the maxima of the convolution results to see which template had the highest chance of being a match. However, many factors deteriorate the viability of this strategy. One factor that seems to affect the accuracy of template matching is the angle of the number. The slight tilt causes a lack of alignment that will increase the chance of a false positive. Also, a jersey can fold over itself, which will cause distortion of the number.

## 3.2 OCR

Optical Character Recognition (OCR) technology got better and better over the past decades thanks to more elaborated algorithms, more CPU power and advanced machine learning methods. Getting to OCR accuracy levels of 99% or higher is however still rather the exception and definitely not trivial to achieve.

First, Let's Define OCR Accuracy

When it comes to OCR accuracy, there are two ways of measuring how reliable OCR is:

- Accuracy on a character level
- Accuracy on a word level

In most cases, the accuracy in OCR technology is judged upon character level. How accurate an OCR software is on a character level depends on how often a character is recognized correctly versus how often a character is recognized incorrectly. An accuracy of 99% means that 1 out of 100 characters is uncertain. While an accuracy of 99.9% means that 1 out of 1000 characters is uncertain.

Due to the inaccuracy of the template matching method, we decided to try OCR to detect the number. The OCR we used was designed to read a text file of numbers and letters. It analyzes each character and computes a correlation with every other template, and picks the best match. We edited the source code to only consider numbers as the possible results. However, testing revealed some flaws. Very frequently, characters would get confused for other characters. An example of a pair of numbers that would commonly get confused for each other is '1' and '4'. However, for this application, we do not need an OCR engine that can recognize any number. We only need it to distinguish between five possible numbers. Restructuring our identification method to only account for those five possibilities will cut down on errors during number recognition.

We train a different detector for each number from 1 to 22. We found that this approach is far more reliable that having classifiers for digits 0–9, because two digits numbers are not always well separated, and so they tend to cause missed detections. Moreover, detecting each digit separately would force us to impose constraints on spatial arrangement of detected digits which are not easy to verify in the cases where numbers are not perfectly horizontal.

Each detector acts as a dichotomize, allowing us to directly recognize which is the particular number that has been detected. Each classifier has been trained with 50 positive and 100 negative examples, the latter being randomly selected from images, while the former have been manually cropped. Other positive examples have been generated with graphic programs or obtained by small rotations of some selected images. Figure [6] shows examples from the training set used to build the detector for number 10.

Figure 6. Positive examples from the training set used to build detector for number 10.

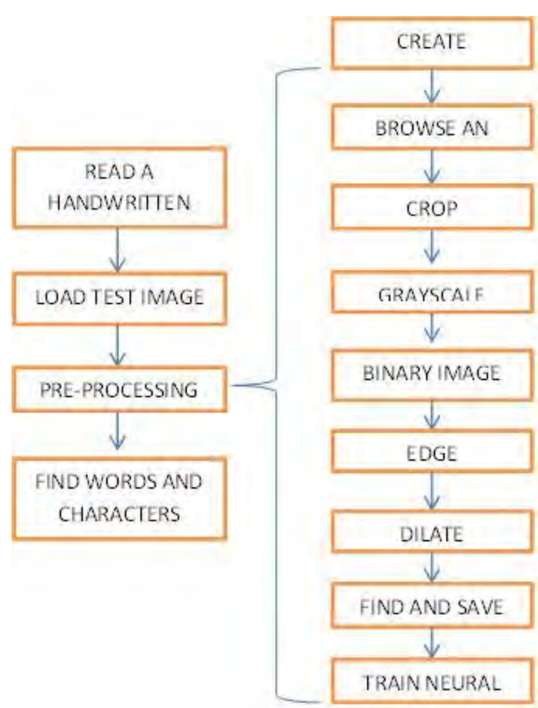
Therefore, for two digit numbers, we will use two parallel OCRs to detect the jersey number more efficiently. Each OCR will be customized for each digit. We know that the Lakers have five numbers on their starting roster: ‘7,’ ’15, ’16, ’17’, and ‘24’. Therefore, we know the first digit of the number will either be 1 or 2, and the second digit will be a 4, 5, 6, or 7. Using this fact, we can run two parallel OCRs. We will construct one OCR for the first digit (that only recognizes 1 and 2), and another OCR for the second digit (which only recognizes 4, 5, 6, and 7). We then take the results of both OCRs and combine the digits to complete the estimate of what the number is.

The analysis of a single digit number is a simple derivative of the previous analysis. We will just use the OCR for the second digit of the two digit numbers to find the number.

After getting the results of our OCR number recognition, we check for any exact matches. If we get a number that does not match, we will use some logic specific to the Lakers team. Since we know that there is only one player with a single digit number, we can claim that any single digit number corresponds to that player. The other four numbers are 24, 15, 16, and 17. Something we can exploit from these numbers is that their second digit is unique. Therefore, we can depend solely on the second digit to find what the number is.

We run the risk of incorrectly identifying players when there are less than five on the court. Therefore, we must do some additional processing to remove extraneous areas. We can characterize jerseys as having regions inside that have a larger than unity height to width ratio, which are numbers. We also know numbers typically take up a roughly constant portion of a total image.

This is the main design layout of the OCR system. The video is processed and the jersey are detected using Kalman filter. The Kalman filter algorithm helps to track moving objects. The design layout of OCR is given below:



## 3.3 Noise Removal

Images are often degraded by noises. Noise can occur and obtained during image capture, transmission, etc. Noise removal is an important task in image processing. In general the results of the noise removal have a strong influence on the quality of the image processing techniques. Several techniques for noise removal are well established in color image processing. The nature of the noise removal problem depends on the type of the noise corrupting the image. In the field of image noise reduction several linear and nonlinear filtering methods have been proposed. Linear filters are not able to effectively eliminate impulse noise as they have a tendency to blur the edges of an image. On the other hand nonlinear filters are suited for dealing with impulse noise. Several nonlinear filters based on Classical and fuzzy techniques have emerged in the past few years. For example most classical filters that remove simultaneously blur the edges, while fuzzy filters have the ability to combine edge preservation and smoothing. Compared to other nonlinear techniques, fuzzy filters are able to represent knowledge in a comprehensible way. In this paper we present results for different filtering techniques and we compare the results for these techniques.

### 3.3.1 Fuzzy filter for Impulse noise

A color image can be represented via several color models such as RGB, CMY, CMYK, HSI, HSV and CIE L a* b*. The most well known of these is the RGB model which is based on Cartesian coordinate system. Images presented in the RGB color model consists of three component images, one for each primary color (Red, Green and Blue). Consider a color image represented in the x-y plane, then the third coordinate $z$ =1,2,3 will represent the color component of the image pixel at $(x, y)$. Let $f$ be the image function then $f(x\ y, ,1)$ will represent the Red component of pixel at $(x, y)$. Similarly, $f(x\ y, ,2)$ and $f(x\ y, ,3)$ represent the Green and Blue components respectively. This notation is followed through out this work.

Sorted vector (increasing order) of window elements shown above

Calculate the median ($M$) of the above vector

Calculate the difference between $M$ and each pixel value of window, here $d_i = |M - p_i|$, $i = 1,2,3,4,5,6,7,8,9$.

| | | |
|---|---|---|
| $d1$ | $d2$ | $d3$ |
| $d4$ | $d5$ | $d6$ |
| $d7$ | $d8$ | $d9$ |

Arrange all pixels of window that have $d_i \leq \delta_1$ in a new vector and calculate the median (*med*) of it.

Figure 7. A scheme for the computation of Median of noise-free pixel

The above median (*med*) is used to find the correction term for each pixel in the noisy image.

## 3.3.2 Structure of Impulse Filter

The proposed filter is designed for the reduction of impulse

| | | |
|---|---|---|
| $p7$ | $p8$ | $p9$ |

The current pixel ( $p_5$ ) with its neighborhood pixels ( $p_1$, $p_2$ , $p_3$ , $p_4$ , $p_6$ , $p_7$ , $p_8$ , and $p_9$ ).

noise in color images by treating each color component separately. Interactions among these color components are used to determine the similarity of the central pixel vis-à-vis the neighboring pixels. The nature of impulse noise is random in the sense that it corrupts some pixels while leaving others untouched. So our objective is to identify the noisy pixels along with the amount of noise present. It may be noted that the impulse noise bears similarity with the high frequency content of images like edges and fine details because both reflect sudden changes in pixel values. Three different membership functions, viz., Large, Unlike and Extreme are used to differentiate the noisy pixels from the high frequency contents. The proposed impulse filter consists of two sub filters in cascade.

# **A.** The First Sub filtering

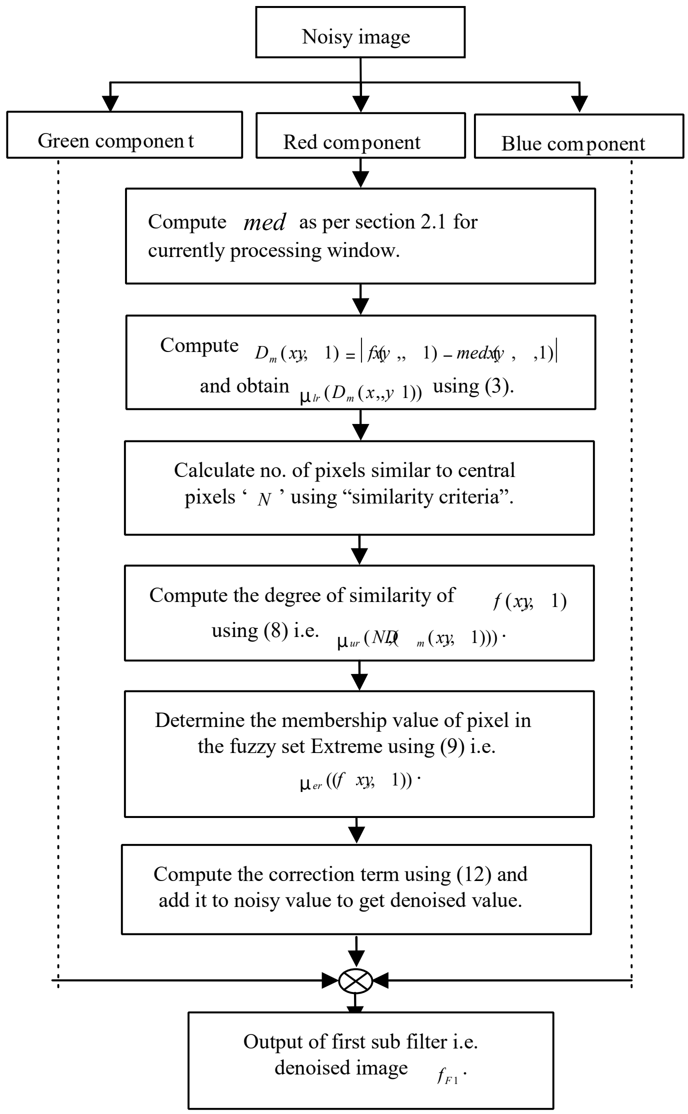



Figure 8. A scheme for the second sub filter

# B. The Second Sub filter

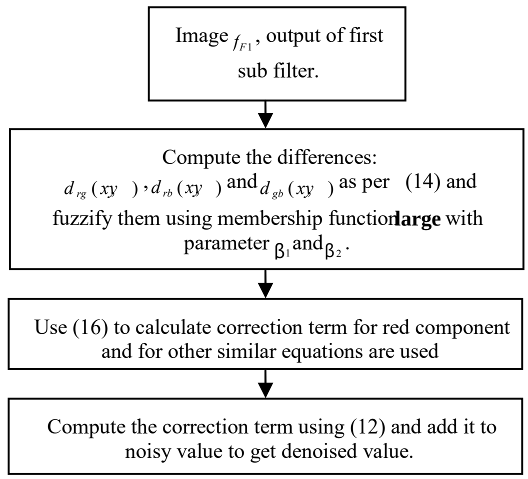



Figure 9. A scheme for the second sub filter

A scheme for the Impulse filter is illustrated below.

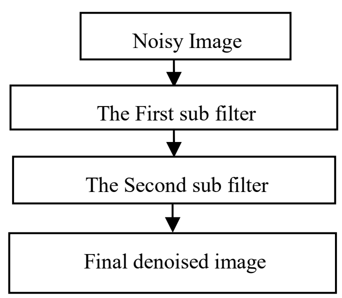



Figure 10. Block diagram of Impulse Filter

Therefore, we can first eliminate any noise inside the jerseys that are smaller than a certain percentage of the total image, such as .02% like this method [1]. Then we find the bounding boxes of all the remaining regions, and only keep regions that have a height to

Width ratio greater than the threshold, which will be something greater than 1. This will remove spotty noise. An additional result of this processing is that the logo of the team, which hovers above the number of the player, will be removed.

If all goes according to plan, only players will be processed by the algorithm. After we have our guesses for each player's identity, we can print the players' names right under their numbers to demonstrate to the user the results.

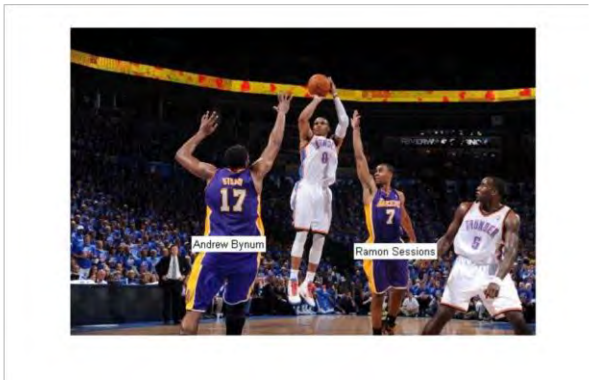



Figure 11. Result of Algorithm.


Figure 12. OCR Failure leads to wrong player getting recognized

# Chapter 4

## Results

We tested our algorithm using 30 in-game images with varying sizes and characteristics. Our figure of merit was the number of correctly identified players over the number of players total. We chose to not include pictures where the number was obstructed or altered, because of the algorithm's vulnerability to such features. Our testing produced an accuracy of 53.6%.


Figure 13. Result of jersey recognition

In the above picture, we can see a screen shot of a game between the Los Angeles Lakers and the Detroit Pistons. The Lakers are wearing the dark blue jersey and the Pistons are wearing the white jersey. In the above picture, I put a red square where we can potentially recognize the player's number, and therefore the player. Clearly, some numbers will be a lot easier to recognize than others, i.e. number 2 and 24 from the Los Angeles Lakers team. This is why we will make our algorithm a best effort algorithm, where it will attempt to recognize as many players correctly as possible.

# Chapter 5

## Analysis

Though there was great care in trying to make a robust algorithm to detect players, there were many factors that reduced the efficacy of the algorithm. One issue with the algorithm is that it relies on a faulty method of detecting and identifying text. Frequently, the OCR engine would misidentify a number, which would lead to a wrong identification of a player. The OCR engine is vulnerable to any imperfections in the depiction of the number in its relation to the template. For example, if the number being detected is rotated in any way, the results of the OCR detection will frequently be incorrect. Another problem with


Figure 14. Phantom Player recognized

OCR is that if the number being detected is too thick. This can cause the probability of false positives to increase significantly. A possible result of the mentioned vulnerability is shown in Figure 10.

Another issue that arose was that many pictures had a lot of noise with similar RGB properties to the Lakers' jerseys. The MAP detector would detect large regions of noise as Laker's players, and the processing between the MAP detection and OCR detection did not eliminate the noise. This sequence of events potentially leads up to something akin to Figure 6, where a "phantom" player is identified. The prevalence of this error can be prevented by more aggressive processing to remove noise. However, the tradeoff to more aggressive processing can lead to more errors in identifying the number, due to distorting the jersey or number. The best solution might be to design a more robust MAP detector, to exclude more noise and reduce the chance of a false positive jersey getting recognized.

Another issue that arises is the nature of the input picture. Many characteristics of the input images can impact the efficacy of the algorithm. The shading on the jersey can impact the output of the map detector, and cause issues if there is distortion of the MAP detector output. Also, if someone's arm cuts across the jersey or the number, it can cause issues by either ruining the isolation of the number or splitting the region of the jersey. Ripples in the jersey also can cause issues, because they can overlap the number on itself and cause errors in the OCR detection. All of these errors can lead to no player getting recognized in the image, as shown below in Figure [15].


Figure 15.  No Player recognized

# Chapter 6

## Conclusion

The goal of this thesis is to investigate the viability of constructing a detection method for identifying players from the NBA in an in-game image. Though our focus was scaled down to one team and one of their jerseys, this does not take away from the results of the paper. We found limited success with our algorithm such as,

Experimental results show the proposed method could correctly recognize the jersey numbers even if the size of a player was relatively small and when the jersey numbers were in a very low resolution as well. There are cases where jersey number detection process failed to detect the complete shape of the numbers and thus influenced the subsequent results, shown in Figure[16]. Proposed method triggered the jersey number recognition process on the correctly detected jersey number blobs the average recognition rate of the proposed approach was 83.74% for all the testing videos.


Figure 16. False recognition results

But there is a lot that can be done to increase the robustness of this algorithm. For example, the method implemented in [3] can be used to improve the text recognition for the jersey number. Overall, with more robust individual components to our algorithm, and more time to incorporate other teams into the system, there is great potential for a system that will identify players from any team.

# REFERENCES

[1] M Saric, H Dujmic, V Papic, N Rozic, "Player number localization and recognition in soccer video using HSV color space and internal contours," World Academy of Science, Engineering and Technology 43 2008

[2] S Ozbay, E. Ereelebi, "Automatic Vehicle Identification by Plate Recognition," World Academy of Science, Engineering and Technology 9 2005

[3] I. Z. B. A. Aziz, "Object's RGB Color Extraction", Nov 2007, unpublished

[4] L. Ballan, M. Bertini, A. D. Bimbo, and W. Nunziati. Soccer Players Identification Based on Visual Local Features.

[5] E. Andrade, E. Khan, J. Woods, and M. Ghanbari. Player identification in interactive sport scenes using region space analysis prior information and number recognition.

[6] F. Chollet. Keras: Theano-based deep learning library. https://github.com/fchollet/keras, 2015.

[7] D. Delannay, N. Danhier, and C. De Vleeschouwer. Detection and recognition of sports(wo)men from multiple views.

[8] S. Gerke, S. Singh, A. Linnemann, and P. Ndjiki-Nya. Unsupervised color classifier training for soccer player detection. In *Visual Communications and Image Processing (VCIP).*, 2013.

[9] C.-W. Lu, C.-Y. Lin, C.-Y. Hsu, M.-F. Weng, L.-W. Kang, and H.-Y. M. Liao. Identification and Tracking of Players in Sport Videos. In *Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service - ICIMCS '13*, page 113, New York, New York, USA, 2013. ACM Press.

[10] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, number 2, page 5. Granada, Spain, 2011.

[11] Michael Isard and Andrew Blake, ―Condensation - conditional density propagation for visual tracking,‖ *IJCV*, vol. 29, pp. 5–28, 1998.

[12] M. Li et al., ―Rapid and robust human detection and tracking based on omega-shape features,‖ *ICIP*, 2009.

[13] Y. Zhai and Shah, ―Visual attention detection in video sequences using spatiotemporal cues,‖ *ACM MM*, 2006.

[14] S. Jiang, Q. Ye, W. Gao, T. Huang, ―A new method to segment playfield and its applications in match analysis in sports video,‖ ACM Multimedia, pp. 292-295, 2004.

[15] V. Vapnik, The nature of statistical learning theory, Springer-Verlag, New York, 1995.

[16] G. Zhu, D. Liang, Y. Liu, Q. Huang, W. Gao, ―Improving particle filter with

support vector regression for efficient visual tracking,‖ IEEE International Conference on Image Processing, vol. 2, pp. 422-425, 2005.

[17] P. Felzenszwalb, D. McAllester, and D. Ramanan. A Discriminatively Trained, Multiscale, Deformable Part Model. In CVPR, 2008.

[18] Felzenszwalb PF, Girshick RB, McAllester D, Ramanan (2009) Object Detection with Discriminatively Trained Part Based Models.

[19] Khan SM, Shah M (2006) A Multiview Approach to Tracking People in Crowded Scenes using a Planar Homography Constraints. In: European Conference on Computer Vision.

[20] Okuma K, Taleghani A, de Freitas N, Little JJ, Lowe DG (2004) A Boosted Particle Filter: Multitarget Detection and Tracking. In: European Conference on Computer Vision .

[21] J.A. Bilmes, ―A Gentle Tutorial of the EM Algorithm and Its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models,‖ TR-97-021, Berkeley, April 1998.

[22] J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, 2001.

[23] David J. Abel and David M. Mark. A comparative analysis of some twodimensional orderings. Int. Journal of Geographical Information Systems, 4:21 31, 1990. 126, 174

[24] Y.Demiris A.Dearden and O. Grau. Tracking football player movment from a single moving camera using particle lters. In The 3rd European Conf. on Visual Media Production (CVMP 2006), 2006. 38

[25] Impire AG. Vis.track. http://www.vistrack.de.

[26] E.L. Andrade, E. Khan, J.C. Woods, and M. Ghanbari. Player classication in interactive sport scenes using prior information region space analysis and number recognition. In Proc. of Int. Conf. on Image Processing (ICIP), volume 3, pages 129 132, 2003. 118

[27] Andreas Andreakis, Nicolai von Hoyningen-Huene, and Michael Beetz. Incremental unsupervised time series analysis using merge growing neural gas. In JosØ Carlos Pr ncipe and Risto Miikkulainen, editors, Proc. of Int. Workshop on Advances in Self-Organizing Maps (WSOM), volume 5629 of Lecture Notes in Computer Science, pages 10 18. Springer, 2009. 163

[28] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle lters for online nonlinear/non-gaussian bayesian tracking. IEEE Trans. on Signal Processing, 50(2), Feb 2002. 26, 50

[29] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, 2 edition.

[30] Noboru Babaguchi and Ramesh Jain. Event detection from continuous media. In Proc. of Int. Conf. on Pattern Recognition (ICPR), 1998.

[31] Noboru Babaguchi, Yoshihiko Kawai, and Tadahiro Kitahashi. Event Based Video Indexing by Intermodal Collaboration. IEEE Transactions on Multimedia, 9(2):68 75, 2002.

[32] Tucker Balch, Adam Feldman, and Wesley Wilson. Assessment of an r d system for animal tracking. Technical report, Georgia Institute of Technology, October 2004.

[33] Dana H. Ballard and Christopher M. Brown. Computer Vision. Prentice Hall, 1982.

[34] Yaakov Bar-Shalom. Extension of the probabilistic data association lter to multitarget environments. In Symp. on Nonlinear Estimation, San Diego, CA, September 1974. 55

[35] Yaakov Bar-Shalom and Thomas E. Fortmann. Tracking and Data Association,

volume 179 of Mathematics in Science and Engineering. Academic Press, 1988. 8, 26, 50, 55, 65.

[36] Yaakov Bar-Shalom, Thomas E. Fortmann, and M. Sche e. Joint probabilistic data association for multiple targets in clutter. In Proc. of Conf. on Information Sciences and Systems, Princeton, NJ, March 1980. 55, 141

[37] Yaakov Bar-Shalom and A. G. Ja er. Adaptive nonlinear ltering for tracking with measurements of uncertain origin. In IEEE Conf. on Decision & Control, pages 243 247, New Orleans, LA, December 1972. 52

[38] Yaakov Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. Automatic, 11:451 460, September 1975. 52

[39] Lluis Barcel , Xavier Binefa, and John R. Kender. Robust methods and representations for soccer player tracking and collision resolution. In Proc. of Int. Conf. on Image and Video Retrieval (CIVR), pages 237 246, 2005. 55

[40] Francois Bardet, Thierry Chateau, and Datta Ramadasan. Real-Time Multi-Object Tracking with Few Particles. In AlpeshKumar Ranchordas and Helder Araujo, editors, Int. Conf. on Computer Vision Theory and Applications (VISAPP), volume 1, pages 456 463, Lisboa, Portugal, Feb. 2009. INSTICC, INSTICC press. 57, 78

[41] Ricardo M. L. Barros, Milton S. Misuta, Rafael P. Menezes, Pascual J. Figueroa, Felipe A. Moura, Sergio A Cunha, Ricardo Anido, and Neucimar J. Leite. Analysis of the distances covered by rst division brazilian soccer players obtained with an automatic tracking method. Journal of Sports Science and Medicine, 6:233 242, 2007. 156

[42] Adrien Bartoli, Navneet Dalal, and Radu Horaud. Motion panoramas. In Journal of Computer Animation and Virtual Worlds, 15(5):501 517, nov 2004. 38

[43] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. SURF:

Speeded Up Robust Features. Computer Vision and Image Understanding (CVIU)

# Appendix

Here are the methods I used

- create_templates
- jerseyrecognition
- lines
- numberrecognition
- OCR
- OCR_1
- OCR1
- playerrecognition
- read_letter
- read_letter1
- removeArea
- removeBox