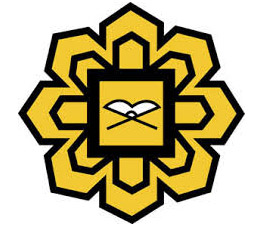
AUGMENTED REALITY SNOOKER PLAYING FOR SPECTATORS' VISUALIZATION

MOHD AMIRUL ASHRAFF BIN CHIK 1519017

ELECTRONIC – COMPUTER INFORMATION ENGINEERING

FINAL YEAR PROJECT

SUPERVISOR: DR MALIK ARMAN MORSHIDI



DEPARTMENT OF ELECTRONIC (ECE)

KULLIYYAH OF ENGINEERING

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

MAY 2019

# ABSTRACT

Augmented reality snooker playing for spectators’ visualization is a unique idea for bringing engineering to the well-known game of snooker. The project is to develop of an augmented reality system that provides the visualization of the estimated snooker balls trajectories for the spectators while the game resumes. This project has been proposed in order to meet the objective of improvement the previous development, mainly on ball detection and trajectories calculation. This project has been focused on identify the suitable method of algorithm for the ball detection and trajectories to overcome the problem of low accuracy on ball detection and low accuracy on calculation of trajectories. Basically, the improvement of the trajectories calculation needed the improvement on the whole system. Generally, for the system approach, the system has divided the task into four parts which is identify the boundary of the table, snooker ball detection, cue detection and lastly the calculation for the ball and cue trajectories. The boundary of the table has been drawn by identify its border coordinate. Next, using the Hough circle transform module which is the advantages when the system loads the top view of snooker video to detect the ball. The detection of snooker cue detected using the line transform module. This three information then be using to generate the formula to calculate the trajectories. Of course, as the player’s pool stick is repositioned around the table, the projected arrow will be updated in real-time to reflect the new trajectories. Keeping the objectives in mind, the researched has been done to the topic thoroughly to examine the potential difficulties and create a plan for implementation. The predicted trajectory that come up from this project show around at least degree error from the real trajectories.

# ACKNOWLEDGEMENTS

All praises to Allah SWT for His guidance, strength and blessing that I am able to

successfully accomplished my Final Year Project 2 report as undergraduate student of Computer and Information Engineering in International Islamic University Malaysia. I would like to thank Dr. Malik Arman Morshidi for giving me opportunity to doing this final year project under him, also for his guidance and assistance throughout this project. With the proper guidance and help, it was possible for me to finish my Final Year Project 2 successfully. Thank you to our Programme Coordinator, Madam Jannah Ibrahim for her assistant, a guidance and help throughout Final year project 2 period session. Finally, I would like to express my gratitude to my parents, family, lecturers and friends for countless motivation, advices as well as their help. I thanked them for being supportive in various aspect. Other than that, I would like to thank Kuliyyah of Engineering for the opportunity given to gain such a valuable experience throughout doing my FYP 2.

# TABLE OF CONTENT

ABSTRACT i

ACKNOWLEDGEMENTS ii

TABLE OF CONTENT iii

LIST OF TABLES i

LIST OF FIGURES v

LIST OF SYMBOLS vi

CHAPTER 1 1

1.1 OVERVIEW 1

1.2 PROBLEM STATEMENT 2

1.3 OBJECTIVE 2

1.4 METHODOLOGY 3

1.5 SCOPE 4

1.6 REPORT ORGANIZATION 4

CHAPTER 2 5

2.1 OVERVIEW 5

2.2 IMAGE PROCESSING 6

2.3 IMAGE PROCCESING MODULE 7

2.3.1 CANNY ALGORITHM 7

2.3.2 HOUGH TRANSFORM 8

2.3.3 HOUGH TRANSFORM FOR OTHER SHAPE 13

2.4 RELATED WORK 14

2.4.1 TRAJECTORIES CALCULATION 14

2.5 COMPARISON BETWEEN PREVIOUS WORKS 18

2.5.1 TABLE BOUNDARY DETECTION 18

2.5.2 BALL DETECTION 20

2.5.3 CUE DETECTION 22

2.5.4 TRAJECTORIES 23

2.6 SUMMARY 25

CHAPTER 3 26

3.1 OVERVIEW 26

3.2 IDENTIFY TABLES BOUNDARIES 27

3.3 BALL DETECTION 29

3.4 CUE DETECTION 32

3.5 BALL MOVEMENT DETECTION 33

3.6 BALL TREJECTORIES 35

3.7 REFLECTION OF THE CUE, BALL - TABLE 39

3.8 SUMMARY 43

CHAPTER 4 44

4.1 OVERVIEW 44

4.2 SETUP PROCESS 44

4.3 TRAJECTORY COMPARISON 46

4.3.1 BETWEEN BALL IN STATIC CONDITION AND MOVING CONDITION 46

4.3.2 BETWEEN CUE LINE AND BALL MOVEMENT LINE 47

4.3.2 BETWEEN PREDICTED TRAJECTORY AND TRUE TRAJECTORY 48

4.3.3 ANGLE OF REFLECTION 49

4.4 SUMMARY 49

CHAPTER 5 34

5.1 CONCLUSION 34

5.2 LIMITATION AND CHALLENGES 34

5.3 FUTURE WORK 35

BIBLIOGRAPHY 36

APPENDICES 38

# LIST OF TABLES

TABLE 2.1: Comparison of method to find table boundaries 18

TABLE 2.2: Comparison of method for ball detection 20

TABLE 2.3 : Comparison of method for cue detection 22

TABLE 2.4 : Comparison of method to calculate trajectories 23

# LIST OF FIGURES

FIGURE 1.1: Flow chart of methodology 3

FIGURE 2.1: Gaussian filter 7

FIGURE 2.2 : A depiction of the original image (left) after being applied with a gaussian blur 8

FIGURE 2.3 : Equation to find gradient strength and direction 8

FIGURE 2.3 : Detected edge point of a line 9

FIGURE 2.4 : Detecting all possible lines which go through a point 10

FIGURE 2.5 : Showing distance and angle of a line from the origin, that is passing through the edge point 11

FIGURE 2.6 : Showing the maximum diagonal distance of the picture. 11

FIGURE 2.7 : Maximum diagonal distance of the picture 12

FIGURE 2.8 : Output of the transformation 12

FIGURE 2.9 : Snooker stick labeled with two different colour pieces of tape 14

FIGURE 2.10 : Intersection 15

FIGURE 2.11 : One solution 15

FIGURE 2.12: No Solution 15

FIGURE 2.13 : The path of the cue ball with respect to the pool stick 16

FIGURE 2.14 : Trajectory of a ball bouncing off the boundary of pool table. 17

FIGURE 2.15 : Rotation coordinate 17

FIGURE 3.1 : Implementation flow 26

FIGURE 3.1: Distorted Table 28

FIGURE 3.2: Assign coordinate to every edge 28

FIGURE 3.3 : Passing image 29

FIGURE 3.4 : grayscale image 29

FIGURE 3.5 : Histogram Equalizer 30

FIGURE 3.6 : Blurry Image 30

FIGURE 3.7 : Hough circle transform 30

FIGURE 3.8 : Ball detection 31

FIGURE 3.9 : Edges detection. 32

FIGURE 3.10 : Hough line algorithm 32

FIGURE 3.11 : Before erosion and dilating process 33

FIGURE 3.12 : After erosion and dilating process 33

FIGURE 3.13 : Movement ball marking algorithm 34

FIGURE 3.14 : Ball movement 34

FIGURE 3.15 : White ball detection 35

FIGURE 3.16 : Idea to draw trajectory line 36

FIGURE 3.17 : Equation of circle line interception 36

FIGURE 3.18 : Graph of interception 37

FIGURE 3.19: Checking intercept 37

FIGURE 3.20: Calculation for quadratic equation 38

FIGURE 3.21 : Ball trajectory 39

FIGURE 3.22 : Interception cue line and table border 39

FIGURE 3.23 : Math calculation for line intersection 40

FIGURE 3.24 : Reflection algorithm 40

FIGURE 3.25 : Zero angle line 41

FIGURE 3.26 : Calculation of angle for reflection. 41

FIGURE 3.27 : Reflection angle on table 42

FIGURE 4.1 : Cue angle 44

FIGURE 4.2 : Line of ball movement 45

FIGURE 4.3 : Initial ball position 45

FIGURE 4.4 : Final ball position 45

FIGURE 4.5 : Theta between cue line and trajectory - ball in static position 46

FIGURE 4.6 : Theta between cue line and trajectory - ball in dynamic position 46

FIGURE 4.7 : Angle between cue line and movement line 47

FIGURE 4.8 : Predicted angle 47

FIGURE 4.9 : True Angle 47

FIGURE 4.10 : Angle of reflection 49

# LIST OF SYMBOLS

*r* Radius

*m*  Gradient

*Δ*  Discriminant

*θ* Theta

**CHAPTER 1**

**INTRODUCTION**

## OVERVIEW

Every Year, WSF (World Snooker Federation) host a lot of snooker championship tournament including senior snooker championship, female snooker championship and the greatest and absolutely the most waiting tournament, world snooker championship. Cue sport have their own supporters, fans and spectators all around the globe. From hard die fan that watch every snooker game for 23 years to the common ordinary fan that follow the sport. Snooker sport have huge amount of fan include the one who watch the game on television or internet (Rick, Maria, Marvin & Tucker,2011). Unfortunately, some article said that snooker sport won’t survive a decade because of many technical aspects. Snooker sport under attack from above and below. The golden era of lifestyle tie-in sponsorship has long since passed. Factor that cause all this problem probably happen when old generation that fanatic on snooker was gone, and then come new generation that saw this sport as old man game and feel outdated. Follow this new modern era, the idea is to bring up this sport to another level which is integrated the augmented reality into the game to provide the maximum enjoyment to the spectator when they watch the game. The project is about to develop an augmented reality system that provides the visualization of estimated snooker balls trajectories for the spectators while the game resumes. Basically, it is the improvement from the system that already exist. Previously, the system that has been develop by (Ricardo, Luis & Rodrigues,2013) faces the problem on the ball detection. Even there are a lot of improvement on continuation of that project, the system still can’t have the perfect detection on the ball (Ricardo, Luis & Rodrigues,2016).

## PROBLEM STATEMENT

Knowing the tactical of the snooker player during the snooker game is quite interesting ability to have. However, people can’t visualize or predict the movement of the ball. So, the project basically to develop the augmented reality system on this game. Even there are the development happen in the past of this project, there are a lot of space on improvement such as the ball detection. Even the ball in static condition, the detection seem shaking or moving around the ball show the detection can improve on its accuracy. Moreover, the problem of the balls that were not detected were due to being too close to each other also the one that related with the accuracy. This project also focuses to improve the computation of trajectories. Basically, all previous system tried to improve the calculation of the trajectories with their own algorithm base on their source video.

## OBJECTIVE

The main objective of this project is:

* To develop an augmented reality system that projects the visualization of snooker balls estimated trajectories.
* To improve the accuracy of snooker balls detection by improve detection algorithm.
* To find the best way of calculation to improve the accuracy of snooker balls trajectory.
* To evaluate and compare the performance of the proposal system with existing approach.

## METHODOLOGY

The steps taken to achieve the objectives above are as follows:

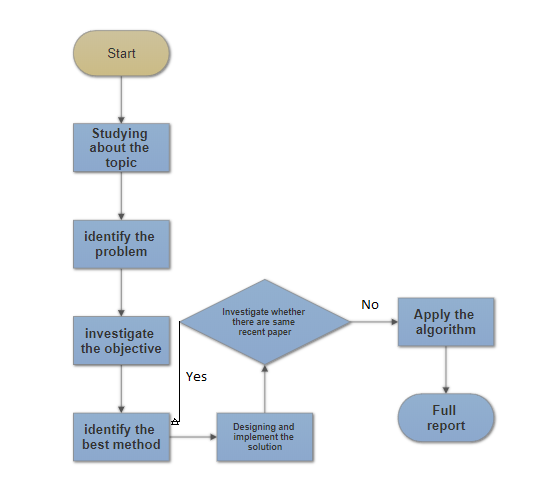
1. Literature Review: in depth research is performed relating to the topic of Augmented reality snooker playing for spectators’ visualization. The primary sources are conference publications and thesis papers, while the fundamental aspects are quoted from textbooks. A few sources cited that are from the Internet were thoroughly vetted for their authenticity.
2. Identify which method is the best on detection object on snooker table and calculate the trajectories.
3. Find out which algorithm works best with the method.
4. Programmed and test the algorithm by using visual studio software and C++ programming language, identified the effectiveness based on the accuracy of detection

Figure 1.1: Flow chart of methodology

## SCOPE

The scopes of this project are as follows:

1. The project involves with C++ language programming by using the visual studio software and knowledge about image processing.
2. The scope of the project only focusses on the balls and cue trajectories. As for table boundaries calculation, that part can divided into its own research work due to the distortion image cause by the lens. Of course, that not on our project scope.

## REPORT ORGANIZATION

This project consists of five chapters and the following are the outlines.

**Chapter 1: Introduction.** This chapter presents an overview of the project itself. Key items of the project such as objectives, problem statements, scope, methodology and structure of the report are discussed in this chapter.

**Chapter 2: Literature Review**. Review of past and current research made by other researchers is discussed. Key theories and key concepts that are related to the project are presented as well. The summary of findings is presented as well.

**Chapter 3: System Design**. This chapter focuses on the details of system design. Overall flow of the system’s operation will be presented in this chapter as well.

**Chapter 4: Result and Analysis**. Results and testing methodology are presented in this chapter. Test results are presented in this chapter.

**Chapter 5: Conclusion and Future Work**. This chapter concludes all the findings and relates its results with the objectives of the project. Suggestions for project improvement are presented in this chapter as well.

# CHAPTER 2

**LITERATURE REVIEW**



## OVERVIEW

In recent years, Sport technology developer has created and develop a technology that give a big impact to the sport world. They came out with the augmented reality technologies that bring the viewer of network and cable broadcast sports events to the next level of enjoyment. This AR technology are being blend together with the content of broadcast video to show something that will help the viewer to understand more about the game. Some common example of element that always used in any live game broadcast is the scores. We take the example of the most popular sport which is football game. If the viewers realize, the logos and the score between two team will be shown at the middle of the field area (circle’s center) when the game is about to start and when the game end. Some effects enable sponsorship commercials and logos to be integrated into the on-air video in a creative way that make the video look professional and do not distract the viewers. More example of visuals includes graphical guide that help the audience easily locate fast moving object like ball. As we realize or not, this kind of technologies also improve the quality of the game. The combination of AR technologies and the slow-motion video solve the problem of the referee to decide the position of the ball for score counting. This has been the big issues before in football game which referee can’t decide the status of the ball position that bouncing on the goal line. FIFA has introduced the forth referee just to in charge for that particular problem, but still it’s not the effective way. So, for the sake to improve the quality of the game, FIFA decide to fully implement this technology at FIFA World Cup 2018 (Rick, Maria, Marvin & Tucker, 2011).

Back to the main purpose for the project, this system actually the improvement of the previous development which is something similar consist of augmented reality in the snooker sport. The main function is to predict the trajectories of moving balls so that the viewers can understand the strategy of the player. This introduce to some knowledge about image processing which is contain the techniques to do the detection on the object and also the introduction to some math equation that involve in calculating trajectories.

## IMAGE PROCESSING

One of the important knowledge and the most crucial part to develop this project is to understand the image processing. Generally, image processing is used for two different purposed which is visual appearance of image to a human viewer improvement and preparing image for measurement of the structure present and features (Parker, 1993).

For each different task, there are not always using the same technique to solve it, but there is considerable overlap, or we can say it has a same pattern. To produce the high-quality work which is efficient and systematic, it is very important to know about the uses to which the processed image will be used and will be put. For visual improvement, this implies having some commonality with the human process, and an appreciation of what cues the viewer respond to in image.

The necessity measurement of the image features be well defining on edges or the colours. The type of measurement that will be performed on the whole scene are very important in determine the appropriate processing step. It may be easier to think the image processing like word processing or the food processing, it just simply rearranges it and do not reduce the amount of the data in that image. Some arrangement may be more appealing to sense, and some way convey more meaning, but these criteria may not be indistinguishable nor use identical method (Russ, 1995).

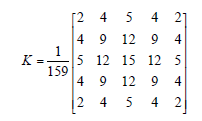
## IMAGE PROCCESING MODULE

Here is some image processing module that involve for object detection in this project

### CANNY ALGORITHM

The canny edges algorithm commonly used in image processing. As its name called, this module used to detect the edges of the image. It was developed by John F. Canny in 1986 (Malak, Muzammil, Matthew& Ali, 2012).

The algorithm works by removes noise present in the image using a gaussian filter such as the one show in figure 2.1.



[1]

Figure 2.1 : Gaussian filter (Malak, Muzammil, Matthew& Ali, 2012)

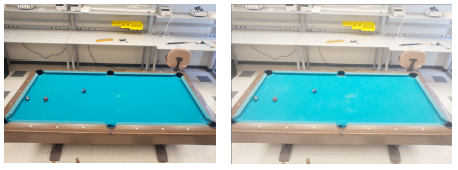
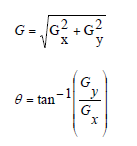


Figure 2.2: A depiction of the original image (left) after being applied with a gaussian blur (Malak, Muzammil, Matthew& Ali, 2012)

This filter must be convoluted by the original image to reduce noise. The output image will be slightly blurred as shown in Figure 2.2.

Next, the edge gradient (G) and the direction are determined from the first derivative in the horizontal direction () and the first derivative in the vertical direction () using the following equations in figure 2.3.

[2]

[3]

Figure 2.3: Equation to find gradient strength and direction (Malak, Muzammil, Matthew& Ali, 2012)

### HOUGH TRANSFORM

The Hough transform is a unique tool that is used in image processing to detect shapes such as lines, curves, circle, ellipse, and nonsymmetrical polygons. The transform is advance or complex algorithm that constructed in 1972 by Richard Duda and Peter Hart (Malak, Muzammil, Matthew& Ali, 2012). Fortunately, now it’s a built-in function in a library called OpenCV.

“The algorithm works by detecting the edge points of an object. Onces the edges point of the object has identified, the algorithm will then analyse each one of edge point by calculating lines, circles, or parabolas (depending on what shape) that pass through that point. An array is then created followed by a virtual Hough space. A Hough space is a grid that will be filled with functions. The algorithm then checks to see which functions are used by using a voting method. The voting method works by analysing the array and seeing which lines, curves have been used for each edge point (Malak, Muzammil, Matthew& Ali, 2012). How this work will be more clearly in the math section.””

“Each shape has a different Hough transform. As example, circles would have a different Hough transform than lines. Line detection by the Hough transform and the math behind it will be explain in this section. Say we had a picture that contained detected edge points of a line where one of the points were detected at point (4, 5) as shown in Figure 2.3.”

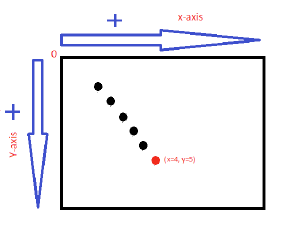


Figure 2.3: Detected edge point of a line (Malak, Muzammil, Matthew& Ali, 2012)

When the Algorithm detects the red point, the following operations are performed:

• Step 1:

The mathematical formula of a line is used:

y= mx + b [4]

Where y and x are coordinates of the detected edge point. m and b value are changing

depending on what line is passing through the point. For example, the transform will

detect lines passing through the detected edge point from Figure 2.4 (Only put 3 lines for simplicity. The transform will compute all possible lines):

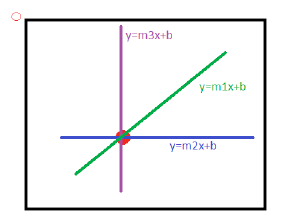


Figure 2.4: Detecting all possible lines which go through a point (Malak, Muzammil, Matthew& Ali, 2012)

Step 2:

The line equations are converted into sinusoidal equations because the slope m can reach

infinity thus a solution would not be impossible to compute. The sinusoidal equation is

expressed by:

ρ = x cos θ + y sin θ [5]

“Where now x and y are the coordinates of the detected line. The angle is the angle from the

origin where the line passing through the detected point is present. is the distance from

the origin to the line. So, the line y = mx + b shown above would now be represented like

figure 2.5.”

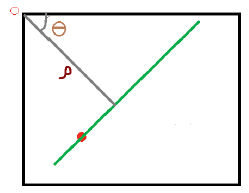


Figure 2.5: Showing distance and angle of a line from the origin, that is passing through the edge point. (Malak, Muzammil, Matthew& Ali, 2012)

Where θ varies from -90 or 89 degrees, and ρ can vary depending on how big your

image is. However, the maximum is the distance from the origin to the diagonal end of the

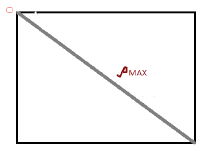
picture like show in figure 2.6.

Figure 2.6: Showing the maximum diagonal distance of the picture.(Malak, Muzammil, Matthew& Ali, 2012)

Step 3:

“An array and Hough space are created with elements θ and ρ.Each detected edge point will form a sinusoidal function in the Hough space. For example, if we had a picture that contained three detected edge points then there will be three sinusoidal functions in the Hough space:”

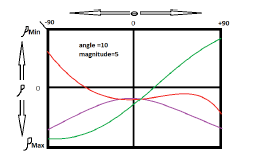


Figure 2.7: Maximum diagonal distance of the picture (Malak, Muzammil, Matthew& Ali, 2012)

“As can see from figure 2.7 there are 3 sinusoids in the Hough space. The three sinusoidal

functions created 2 intersections, however, all three functions share by one intersection.

The elements in that intersection would be voted to be the line elements that belongs to the edge

points. So, say the intersection that had all three functions pass through it in Figure 2.7 was located at θ = 10 and ρ **= 5.** Then the transform would decide that the line that belongs to the detected edge points is locate at θ = 10 and ρ **= 5** from the origin (Malak, Muzammil, Matthew& Ali, 2012). So, the result will get an output that will look something like Figure 2.8”

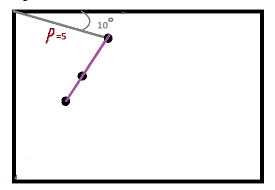


Figure 2.8: Output of the transformation (Malak, Muzammil, Matthew& Ali, 2012)

**2.3.3 HOUGH TRANFORM FOR OTHER SHAPE**

Detecting circles and parabolas becomes more complicated because instead of using the line

equation, the Hough transform will use the circle or parabola equation depending on what is

trying to be detected. These formulas will produce a Hough space that contain three dimensions

rather than two. For a circle, the algorithm would be almost identical to the line however the

equation used would be:

Circle equation

+ = [5]

Parabola equation

= a [6]

“Where y and x would be coordinates of the detected edge point. For the circle, y1, and x1 would be the midpoint of the circle and r in this case would be the radius of the circle. This equation will create a Hough space that contain dimensions (y1, x1, r). in the other hand for parabola, y1 and x1 are the vertex of the curve. The width and opening of the parabola would be determined by the variable a. This will also create a Hough that contains the dimensions (y1, x1, a) making it more complex than the line Hough transform due to the extra dimension added.”

**2.4 RELATED WORK**

**2.4.1 TRAJECTORIES CALCULATION**

Snooker table trajectories are essentially the path that snooker balls follow when they are struck by another ball or by cue stick. The concept of creating trajectories is to show to the people the direction that the balls will take. The first step in the process of calculating the trajectories is detecting the pool stick.



Figure 2.9: Snooker stick labeled with two different colour pieces of tape (Muzammil, 2012)

One of the methods to detect the cue is using the colour tape to mark two point of the stick. Figure 2.9 shown the pool stick consist of two pieces coloured tape. The group utilize the colour segmentation algorithm to detect the pieces of tape. After detecting the coordinate of that two point of sticker, the group will utilize the straight-line equation:

straight-line equation:

y - = m (x - ) [7]

“Next step will be determining the trajectory of the cue ball when the pool stick will point towards it. To build some algorithm to detect circle - line intersection is very important to make sure the program can do that thing which is calculate the point of intersection between the ball and the virtual snooker stick. Line and circle intersection point can be calculated by solving the systems formed by the equations of the line and the circle. The equation of the line has already been found earlier in point slope form. The equation of the circle can be calculated using the radius and the coordinates of the centre that is obtained from Hough Circles algorithm mentioned earlier, and by substituting them into the equation”

*+ =*  [8]

After obtaining a simplified equation by substituting the equation of the line into the equation of

the circle, a quadratic equation is obtained that will consist of a sign for the discriminant,

*Δ = – 4ac* [9]

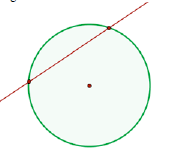


Figure 2.10 show the relationship for two intersection point

Secant Line if Δ > 0

Two solutions: There are two points of intersection.

Figure 2.10: intersection

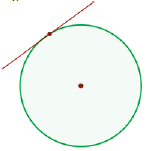


Figure 2.11 show the relationship for one intersection point

Figure 2.11: one solution

Tangent Line if Δ = 0

One solution: The line is tangent to the circle.

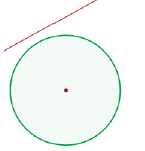


Figure 2.10 show the relationship for no intersection

If Δ < 0

No solution: There is no intersection between the line and the circle.

Figure 2.12: no solution

Once the cue stick line and cue ball circle intersection point is determined, the group will

draw a line that goes through the intersection point and the center of the cue ball circle. This will

essentially be the path of the cue ball, as shown in Figure 2.13.

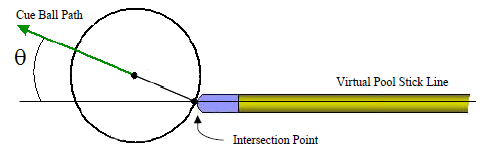


Figure 2.13: The path of the cue ball with respect to the pool stick. (Muzammil, 2012)

“Point slope form equation of the cue ball path can be determined using intersection point and the center of the cue ball. Next is to find the path between the cue ball and other balls or we called it as target ball, the ball that the cue ball come into contact with in its path. Using the same method which is circle line intersection algorithm to determine if there is a target ball along the path of the cue ball. Then the point of intersection will be calculated. A line will be drawn which goes through the intersection point (Contact Point) of the two balls and the center point of the target ball. That line will be the path of the target ball.

The trajectories of the balls and the pool stick will not be allowed to leave the perimeter of the pool table. When a trajectory reaches the perimeter of the pool table, it will be bounced off with an angle of reflection same as the angle of approach. Figure 2.14 demonstrates the theory that

Angle of Approach = Angle of Departure

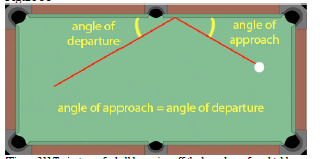


Figure 2.14: Trajectory of a ball bouncing off the boundary of pool table.

Basically, reflection also can be related to rotation method. Rotation can be interpreted as multiplying (rotating in anticlockwise direction) or dividing (rotating in clockwise direction) every point of the coordinate system by a constant vector (Geekforgeek, 2016). Figure 2.15 show the point rotate to perform the reflection.

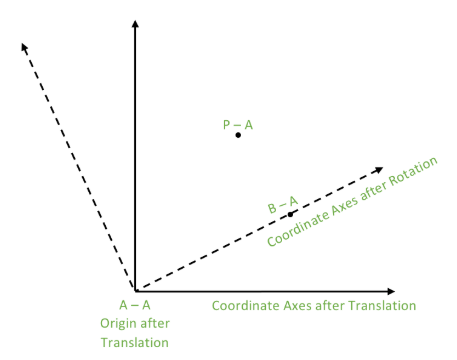


Figure 2.15: Rotation coordinate (Geekforgeek, 2016)

**2.5 COMPARISON BETWEEN PREVIOUS WORKS**

A snooker game can be very challenging and tricky. Even for the audience can’t predict the strategy and tactic use by the player. Every shot made by the player has it own aim to make sure their control the game. This project basically the improvement of the system that actually can visualize the snooker balls trajectories made by the player. There are numerous of research paper that specifically related about the topic of snooker balls detection, calculation on table boundaries and ball trajectories. There are many paper publish which cover different type of approaches, technique, implementing variety of algorithms, just to get the best result on accuracy.

### 2.5.1 TABLE BOUNDARY DETECTION

Table boundaries is the one of the most important aspect in this system. The table boundaries are the most crucial part because it will affect the calculation of the trajectories. Table 2.1 show the comparison of method to find table boundaries from others related work.

|  |  |  |  |
| --- | --- | --- | --- |
| Author/Year | Method | Advantages | Disadvantages |
| Shen, W., & Wu, L. (2010) | Using the color space conversion between HSV and RGB and histogram equalization in digital image processing, the billiard areas of accurate detection are derived from the characteristics of the candidates’ area. | High accuracy on colour detection. | - when video capture decreases the image definition, that will affect the abject shape and hue characteristic, which increase the difficulty of the object segmentation  - unsuitable method to find the table boundaries.  - No calculation and test on table edges. |
| Ling, Y., Li, S., Xu, P., & Zhou, B. (2012) | Combine colour segmentation marker with reflective point marker and detect green ball and non-green ball objectives respectively. | - easy to segment multiple target by label and marker number  -proposed algorithm can effectively detect multi-objectives in snooker game video. | -Difficulties of detecting green object directly. |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2013) | Table boundary extracted using the Canny Edges and Hough line transform.  Then the result represents the candidates line for table boundaries. the system checked the angle of the candidate line. | - when the video get well lighting, the table boundaries were always automatically detected.  - Angle test algorithm increases the chance to get accurate table line | -as boundaries are usually the same colour as the table, there is almost no contrast between them, making it difficult to detect if the light in the surrounding is poor |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2016) | The improvement of (Ricardo, Luis & Rodrigues, 2013) basically using same method on table detection. The improvement is the system use Kinect 2 sensor. Kinect 2 sensor responsibility for capturing the game, which is then processed by the standard computer, enabling detection of game elements including table’s rail ( border) | -Basically, this is the continuation project (Ricardo, Luis & Rodrigues, 2013) so this improve the robustness against changes in lighting condition and noise on video.  -Table boundaries were always automatically detected with less 3-pixels error |  |

Table 2.1: Comparison of method to find table boundaries

### 2.5.2 BALL DETECTION

Table 2.2 show the comparison of method of ball detection from others related work.

|  |  |  |  |
| --- | --- | --- | --- |
| Author/Year | Method | Advantages | Disadvantages |
| Shen, W., & Wu, L. (2010) | Using an automatic segmentation method of local peak edges to extract the snooker table, this step mainly served for the image prepossessing, and then is the detection of video object. Through image preprocessing, morphological processing and clustering, the system can achieve the candidate areas of billiard balls | -An algorithm of billiard detection has been presented which can effectively detect all of billiard balls that appears in the snooker table from video and ensure high accuracy | - when video capture decreases the image definition, that will affect the object shape and hue characteristic, which increase the difficulty of the object segmentation |
| Ling, Y., Li, S., Xu, P., & Zhou, B. (2012) | - combine colour segmentation marker with reflective point marker and detect green ball and non-green ball objectives respectively. | - easy to segment multiple target by label and marker number  -algorithm can effectively detect multi-objectives in snooker game video. | -When the video image is fuzzy, extracting reflective point marker will affect by noise, thus affecting the detection of the green ball |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2013) | Every time movement stop, the ball’s detection starts, consisting in comparing the actual average frame with the initial average frame that contain nothing but the empty table. Binarization applied to the result. Using the contours finder, the system can know find various blobs which may, or not, be balls. Every blob detected will be considered to be a ball if it meets three following parameters:   1. Ratio between blob’s height and width is approximately 1. 2. Relation between blob’s area and circle area , both approximate equal. 3. Ratio between a circle and square area ≈ | - Ball detection work very well when balls are not close to each other.  - The white balls were successfully detected with zero false positive white balls. | - Tiny error on some balls that could not be measured.  - There are balls that were not detected were due to being too close to each other |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2016) | - The methodology for the ball detection similar with (Ricardo, Luis & Rodrigues, 2013) plus the system made the improvement by applied the erosion with the goal to remove remaining noise (small blob). | -all the balls were successfully detected but there were quite a number of false positive, due to failure to filter noise correctly. | - inconsistent detection of white ball. Several times fail to detect due to the white ball being potted and due to a stripped ball occurring with its white part facing upwards in an area with higher luminosity.  - inconsistent accuracy. balls were detected with a maximum of 10mm error of their true position, due to noise in the depth frames and distortion of video taken. |

Table 2.2: Comparison of method for ball detection

### 2.5.3 CUE DETECTION

Table 2.3 show the comparison of method of cue detection from others related work.

|  |  |  |  |
| --- | --- | --- | --- |
| Author/Year | Method | Advantages | Disadvantages |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2013) | The cue information is only needed when the cue is relatively close to the white ball. For cue detection, the current frame is subtracted to the reference frame, enabling the balls not to interfere with the cue detection. This difference only done in small square region of interest (ROI). From the line detection, the system compute the middle line(the line that split the line of right and left cue). This line considered to be the line of the cue, where the tips of cue is considered to be the point of the line closer to the white ball. | - The detection only when the cue enter the ROI save the CPU time. | -imperfection of the cue detection and increase the probability for having fake detection due to not applied the erosion and dilation module. |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2016) | Cue detection also have similar technique with (Ricardo, Luis & Rodrigues, 2013). It based on five steps. First, waiting for all the ball to stop moving. Second, defining the depth reference frame of the table. Third, computing the difference between the depth reference frame and the current depth frame. Forth, finding the largest blob, if any exists (removing all smaller ones). Lastly Detecting the centre line for the largest blob from the starting point near the white ball. | - Cue detection always worked for all shot | - Some error noticed in the cue detection outside this test, when a player did not behave in accordance with pool rules or other expectation. Example like putting two cues in the pool table area or there are several people present with their hands moving near the table border. |

Table 2.3 Comparison of method for cue detection

### 2.5.4 TRAJECTORIES

Trajectories prediction are related to the distance a ball can travel and how many bounces they have on table’s boundary. Table 2.4 show the comparison of method to calculate the trajectories from others related work.

|  |  |  |  |
| --- | --- | --- | --- |
| Author/Year | Method | Advantages | Disadvantages |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2013) | Ball trajectories can be computed after the cue and the centered white ball was detected. The system computes the correspondent line equation by the cue stick and the white ball by its radius and centre point. This two information will be used to calculate the trajectories of the ball. the calculation only be made after the cue struck the white ball.  -Reflection between the ball and table basically calculated using the reflection formula.  -ball interactions compute by considered the touch point between the ball. The calculation had been showed detail in methodology part because the idea for the propose method also inspired from this algorithm | -high rate of success. | Direct ball:  -error occur related to the imperfections on the cue detection and due to the distance that the white ball would travel being high.  Ball that bounce once:  -Unsuccessful trajectory due to the distance travelled, cue detection imperfection and spin gained or lost (Due to speed) in table boundary.  Ball that bounce twice:  -error happen due to what has been stated when the ball bounce one times. But the more the ball bounces to the table border, the lower the success of the prediction. |
| Sousa, L., Alves, R., & Rodrigues, J. M. (2016) | - Basically, the algorithm for ball trajectories is similar with (Ricardo, Luis & Rodrigues, 2013). | - increase the rate of success and improve the result on (Ricardo, Luis & Rodrigues, 2013). The result improved maybe by the affected of using the kinect 2 sensor not by the algorithm. | - Same error happen as (Ricardo, Luis & Rodrigues, 2013) |

Table 2.4 Comparison of method to calculate trajectories

## SUMMARY

From this chapter shows some basic idea how does the program or the system work. Basically, there are so many ways to do the object detection which is some from them have a very simple code and some of them contain a very complex algorithm. There are always the advantages and disadvantages for using these two. Simple coding makes the program run more faster and smoother. It’s also give the advantages to the beginner programmer to get the idea to develop this kind of project (Ricardo, Luis, Rodrigues, 2013). The disadvantages of the simple algorithm maybe it will not as robust as complex coding. Some object detection may not be so perfect and will be lacking in many aspect. On the other hand, complex algorithm can provide more effective and interactive gameplay. All unnecessary detection is removed to make the snooker playing more practical. Usually, the coding become longer and complex when there is factor that make the capture image unclear. By using the image processing, practically we can repair the image like adding the brightness, straighten the line and so on. By doing that, perfect detection of the object can be obtain (Ricardo, Luis, Rodrigues, 2016).

# CHAPTER 3

**METHODOLOGY**



## OVERVIEW

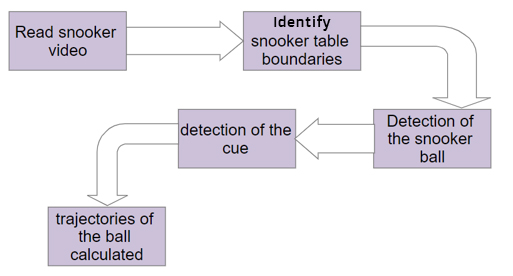
This Augmented reality snooker project is objectively to satisfy and maximize the audience’s enjoyment. Many developments out there also using same concept which is augmented reality technologies but in more advance way to assist inexperienced cue sport player. The concept is almost the same although we can vary the algorithm in our own way to make the coding more robust and the system more interactive. The big different between this project and the other projects that already exist is clearly the objective. To make system that project the trajectory in snooker table in real time playing is a lot more complicated because its need a good hardware like good camera and good projector. This project is different because all the trajectory drawing will be shown on the audience digital screen, and that our target, maximize the satisfaction of the audience watching the snooker game.

Figure 3.1: Implementation flow

Figure 3.1 show the implementation flow. The rough propose concept for this system is start with identify the table boundary. Every edges of the table must be marked first to make sure to get as nearest value and accurate as possible. The detection of the circle basically just using the Hough circle detection. Next is cue detection which is important. The movement direction of the white ball is depends on the cue direction. Movement detection can be obtaining by subtracting the actual frame with the previous frame. Two movement that must be considered here is ball movement and cue movement. The final one is to compute the ball trajectories which is must be include of ball interaction, reflection of the ball-table, how does it project the ball and trajectories to the table.

## IDENTIFY TABLES BOUNDARIES

The accuracy of trajectories of the ball is absolutely connected to the perfection of extraction of table boundary. Sometimes, this calculation is very crucial because even a few pixels wrong will lead to imperfection trajectories projection or in easy word the resultant trajectory will propagate the error. Basically, increasing the distance of this boundary will increase the error.

Unfortunately, the source of the video that available is not that good in term of quality and have small resolution. The distortion image cause by the camera lens make the table not in the condition of perfect rectangle. Fixing the image isn’t in the scope of the project and can have its own research work. It can be including in the future work.



Figure 3.1: Distorted Table

Figure 3.1 shows the distorted image of table compare with the straight red line. As has been advice, the boundary of the table has been made by drawing the straight line along the table boundary as precise as possible to decrease the percentage error of trajectories.

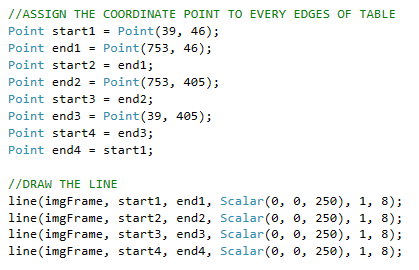


Figure 3.2: Assign coordinate to every edge

The method is to identify the edges of the table and appointed that point as a started point for every straight line. The first point in figure 6 which is identify as the nearest point of the table edge, start1, is pointed to top left edges of the table and it connected to the point end1 which on right top edges to make a line. Than the second point, start2, basically take the position of end1 and its connected to point end2 which is position on bottom right edges. The continuation happens as start3 take the position end2 and connected to end3, follow with start4 take the position of end3 and connected to the first position which is start1 to form a rectangular shape.

## BALL DETECTION

Using the Hough circle algorithm may be very effective as the video is recorded on the top view, so, the image of the ball gets the perfect shape of its circle. Getting the perfect detection of the ball is not just straight forward method but the image needs through the pre-processing to improve accuracy.

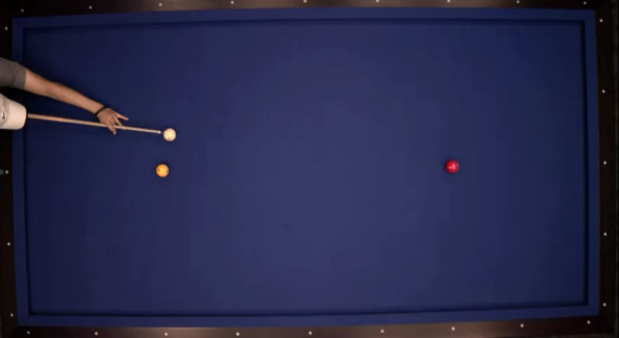


Figure 3.3: Passing image

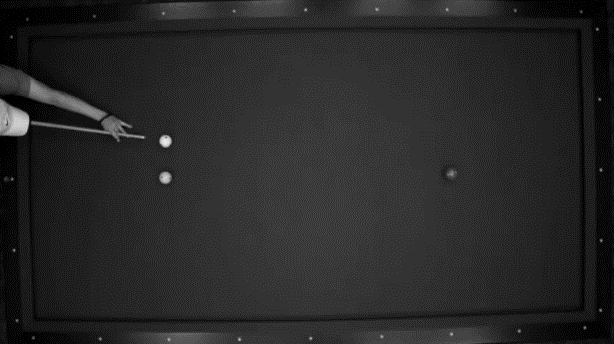


Figure 3.4: grayscale image

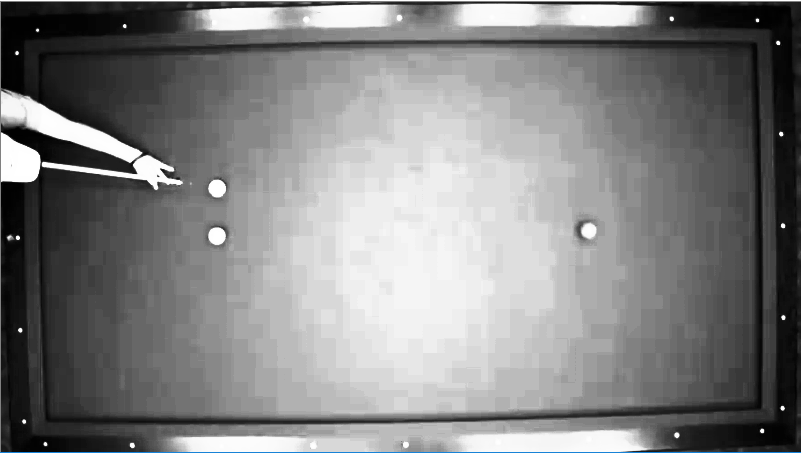


Figure 3.5: Histogram Equalizer

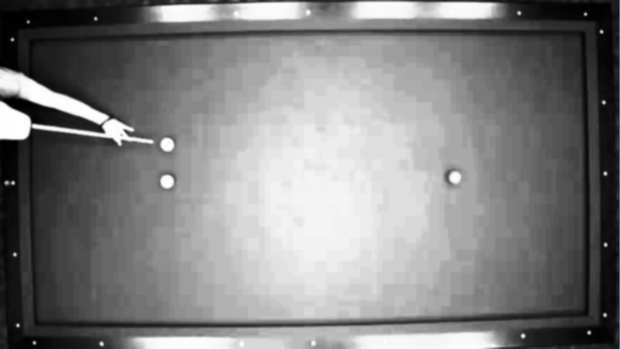


Figure 3.6: Blurry Image

When an Image is passed through the Hough Circle algorithm, three things happen in order. Figure 3.4,3.5,3.6 and 3.7 show the result of pre-processing image that occur after another in sequences. The grayscale image that already obtain from previous procedure will be blur using the Gaussian Blur module to get the smooth image.

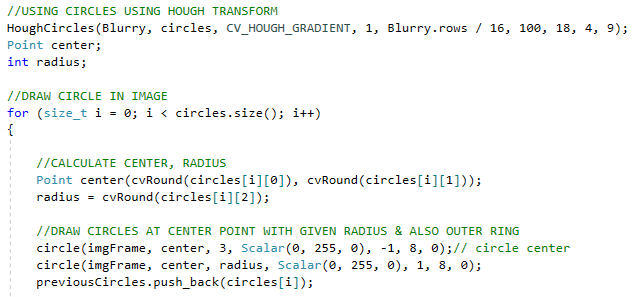


Figure 3.7: Hough circle transform

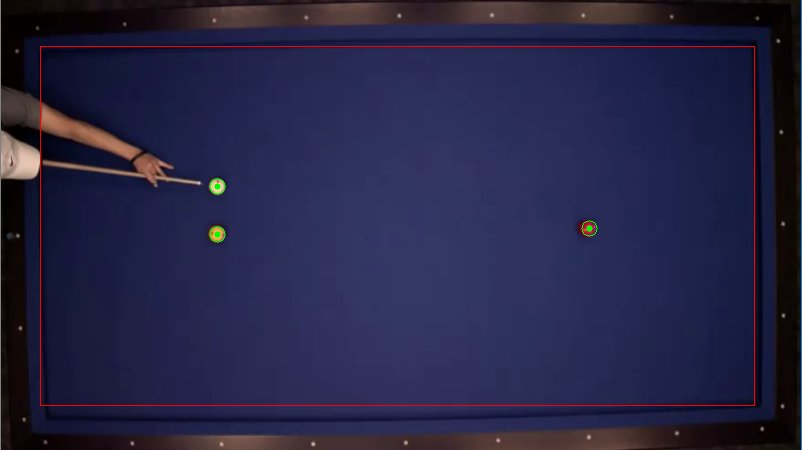


Figure 3.8: Ball detection

After applying the pre-processing process, the image can directly perform the circle detection (Hough circle algorithm) to detect the ball like show in figure 3.8. Some value in this module must be modified and through the process of try and error to get the perfect detection. Setting the maximum and minimum radius for the ball according to the size in the image is very important to make sure no fake detection happen. The perfect range of the ball radius for the image size of 800 x 450 in this specific image is about 4 for minimum and 9 for the maximum. Factor like the shadow maybe will affect the image cause the ball can’t get its perfect circle shape. Because of this factor, maybe some snooker balls can’t be detected by this detection because the detector can’t find any circle shape on that ball. The best way to avoid that happen is using histogram equalization module which is one of the pre-processing modules that shown in figure 3.5. This module can correct the colour of the image and make the colour of the ball brighter and clearly reveal its true shape. Applying this module before detecting the circle shape is necessary for this specific image.

## CUE DETECTION

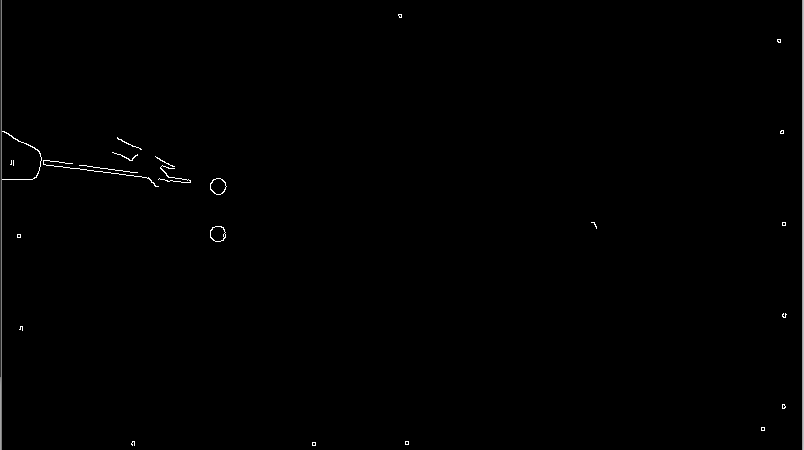
As for que detection, we convert the image into the grayscale shown in figure 3.4, so that we can work with single channel instead of three channels in colour. Working with single channel will be a lot easier, and for information, grayscale image is more stable compared to RGB (red, green, blue) and HSV (hue, saturation, value) image. Then, using the gaussian blur image processing module, that grayscale image will be smooth and easy for detection. The next procedure is to threshold the edges from the image using canny edges detector with kernel size of 3. The output image shown in figure 3.9.

Figure 3.9: Edges detection

Using the Hough line transform is necessary to detect all the straight line. In the code itself involving straight line formula: y= mx + b

Figure 3.10: Hough line algorithm

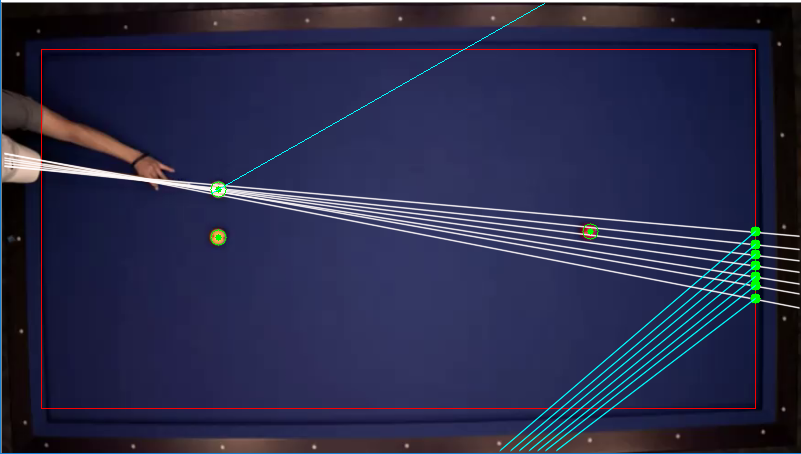
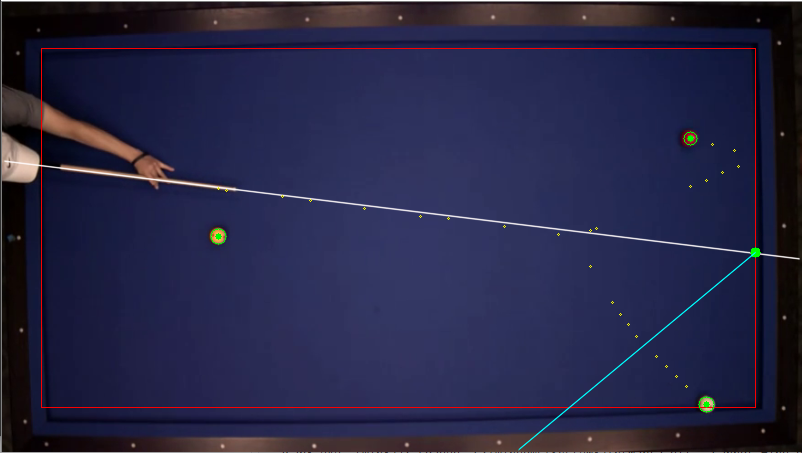


Figure 3.12: After erosion and dilating process

Figure 3.11: Before erosion and dilating process

Some problem may be faces during thresholding the cue line such as fake detection as shown in the figure 3.11. Another factor that make this situation happens also because of the low quality of image. That why the detection is quiet bit messy and form many directions of line. In order to filter out all the fake detection and make the line look more precise, the erosion and dilation module is applied before using the Hough line transform. Basically, these two modules very useful to remove blob. Figure 3.12 show the output image after using the modules.

## BALL MOVEMENT DETECTION

After we got the table boundary, the next procedure is to do the detection on the movement. The main objective to detect the movement of the ball is to compare it to the trajectory line. Every previous frame of the moving ball will be mark so that we can see the trail of the ball.

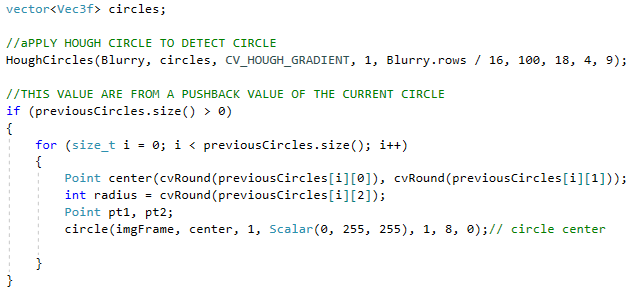


Figure 3.13: Movement ball marking algorithm

All the ball on top of the table will be detected by Hough circle transform detection and the center point of the ball will be mark. When the program iterates to the next frame, the algorithm will keep the center point of the ball and put it into another function. That will keep all the center point of the ball from all the previous frame. Technically, the center and radius value for ball movement has been pass from current position of the circle using push.back function.

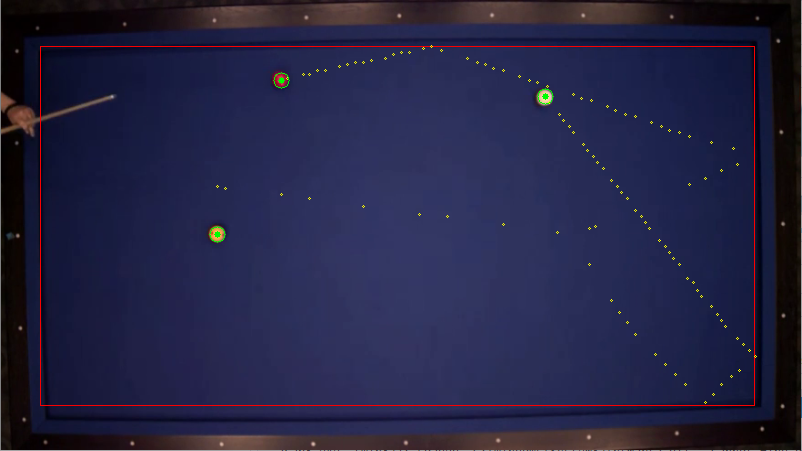


Figure 3.14: Ball movement

## BALL TRAJECTORIES

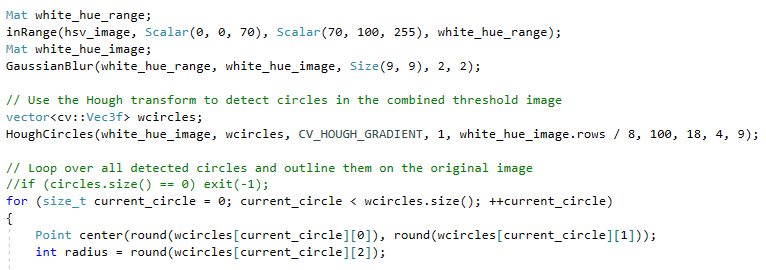
The idea to compute ball trajectories is only after the intersection between cue line and the contour of the white ball happen.

Figure 3.15: White ball detection

Figure 3.15 show the algorithm for white ball detection. The idea is to detect the ball by its colour to differentiate it from other balls. Using HSV colour module for colour detection is better than using RGB module because the image involves with the brightness and shadow.

Back to the trajectory, the cue stick is represented by two points, and the system can compute the correspondent line equation and the white ball by its colour and mark its radius and center point. The white ball is only going to be shot if the cue line intersects any point of the white ball contour. Thus, if any of those points are true, we can start calculating what would be the predictable trajectory, assuming the player will always try to hit its center. It can be easily calculated applying the cue vector to its center.

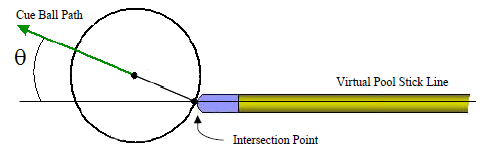


Figure 3.16: Idea to draw trajectory line

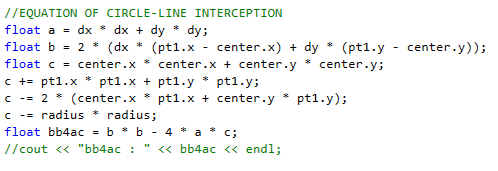


Figure 3.17: Equation of circle line interception

As shown in figure 3.17, pt1 and pt2 are two point of the straight line while center and radius value are come from detected circle. In order to identify the intersection point between the line and circle, the important value that must has to find is discriminant value (- 4ac):

Discriminant (graph is figure 3.18):

* When - 4ac is positive, we get two Real solutions (two or more intersection)
* When - 4ac is zero, we get just one Real solution (both answer same, one intersection)
* When - 4ac is negative, there is no real solution (no intersection)

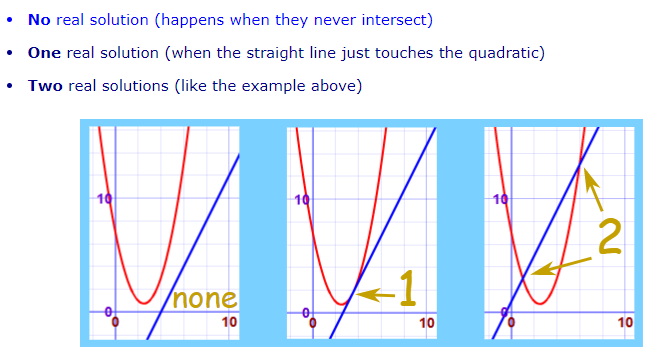


Figure 3.18: Graph of interception (MathIsFun, 2017)

The value of discriminant then will be tested on the if loop

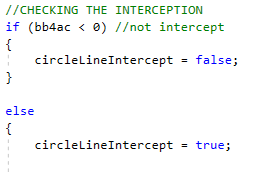


Figure 3.19: Checking intercept

Like has been mention, if the discriminant value is below zero, which mean a negative value, then there is no intersection between line and the circle. The intersection happens when discriminant value is larger than zero. As that happen, the calculation for the solution will be proceed using quadratic formula. Coding for the equation as shown is figure 3.20.

Quadratic equation:

X = [11]

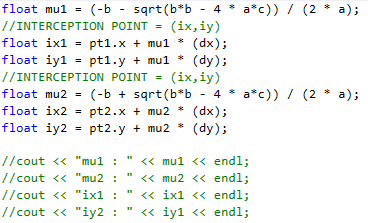


Figure 3.20: Calculation for quadratic equation

The case for two intersection point, there will two value of intersection coordinate which is (ix1, iy1), and (ix2, iy2). From that value, assign the first coordinate that intersect with the circle. Then, draw the line from that first intersection point through center of the circle to have trajectory line. Clear explanation can refer figure 3.16.

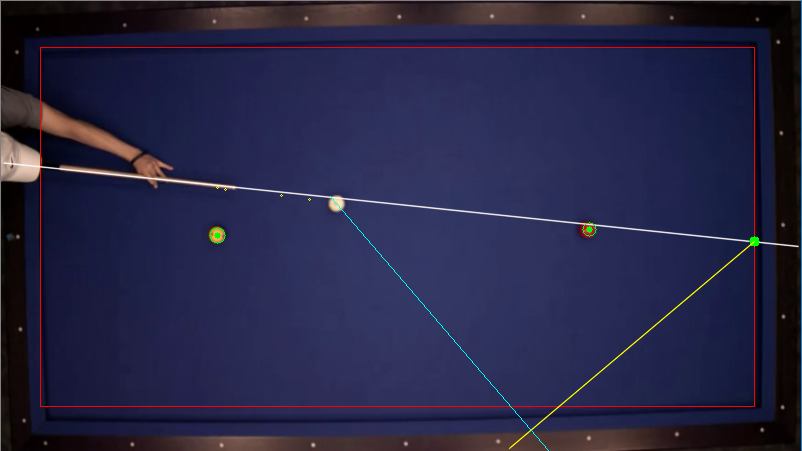


Figure 3.21: Ball trajectory

Figure 3.21 show the output for the drawing trajectory line. The blue colour line, which is the trajectory line are drawn from the first intersection point, all the way through the center of the ball.

### 3.7 REFLECTION OF THE CUE, BALL - TABLE

Figure 3.22: Interception cue line and table border

Figure 3.22 show the algorithm for the intersection of the cue line (also use for the ball) with the table boundaries. Before applying the reflection, the image needs to know the location of its intersect. Basically, what we do here is line intersection. From the figure 3.22, Pt1 and pt2 represent the cue line, while start2 and end2 is the line of the right border of table. Because these two lines is intersect, they share single value of x and y on both of this line. So, we just solve value x and y separately.

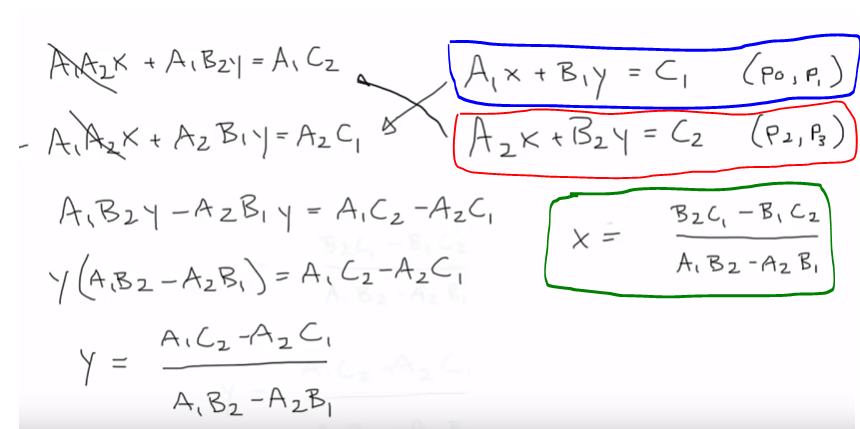


Figure 3.23: Math calculation for line intersection (CodingMath, 2014)

Figure 3.23 show how the line interception has been derived. Blue box is the first line equation and the red box is the second line equation. As you derive, you can see at the end, the denominator value of x and y will be the same.

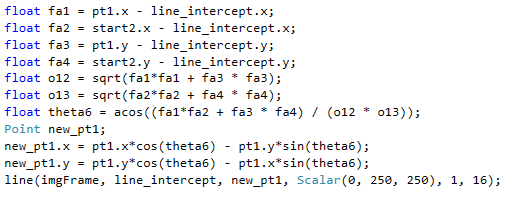
****

Figure 3.24: Reflection algorithm

Move to the last part which show in Figure 3.24, reflection line. The algorithm in figure 3.24 basically perform the calculation of acute angle. The calculation of the angle will be different as shown in figure 3.26 depend on the position of the table border.

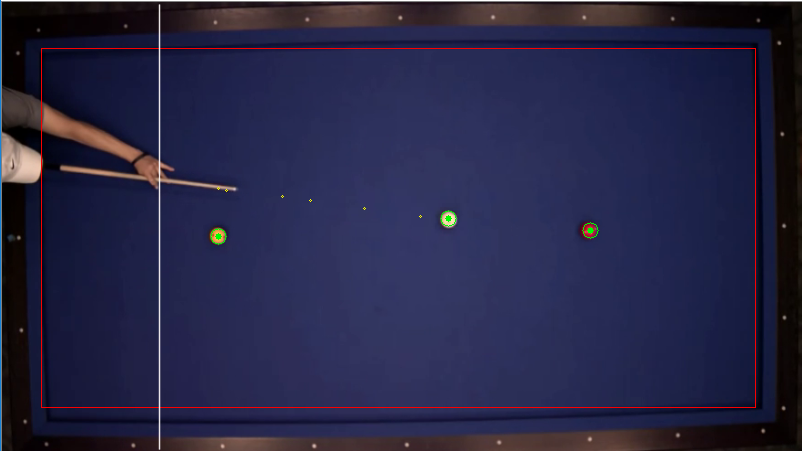


Figure 3.25: Zero angle line

Figure 3.25 show the zero angle line. As line for zero angle is known, now we can construct the formula of reflection for four side of table.

θ +θ

θ θ

-θ θ

θ -θ

Figure 3.26: Calculation of angle for reflection

* Right border angle for reflection: θ
* Left border angle for reflection: - θ
* Top border angle for reflection: + θ
* Bottom border angle for reflection: - θ

From that angle, assign new point as destination point. So, the reflection line will be draw from the intersection point to the new point which has the angle of reflection.

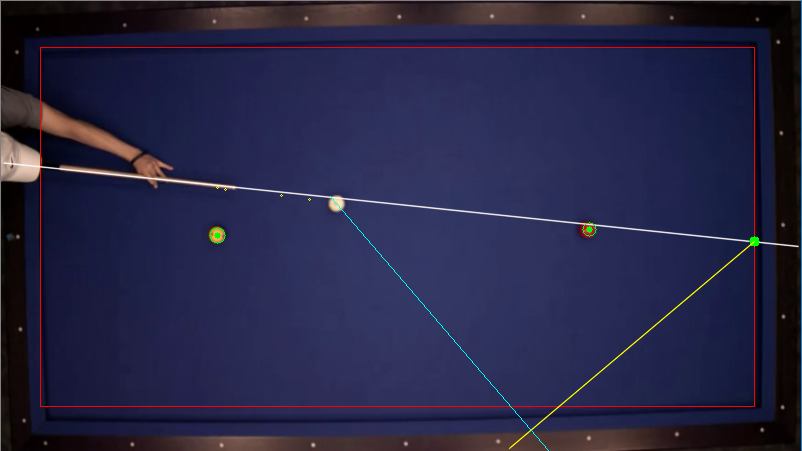


Figure 3.27: Reflection angle on table

Figure 3.26 show the reflection angle and its line which in yellow colour and also the intersection between the cue and the table border which mark with green.

## 3.8 SUMMARY

The detail of algorithms is explained also the math background behind code. Basically, this project need an expertise skill in math. Mathematic topic that has been covered in this including linear equation, Pythagoras theorem, quadratic equation and others. Some calculation like the reflection part didn’t take the solver in the benchmark paper because of some limitation, beside construct the reflection formula from its scratch. Overall system is presented. Some of the detection cant be continue because of some problem such as the quality of video itself. Overall, all the flow of the system’s operation be presented in this chapter. In chapter 4, the chapter will show the result and observation of this propose system.

# chapter 4

**RESULT AND ANALYSIS**



## OVERVIEW

This chapter basically show the result of the trajectory angle. Through this chapter, the result make the comparison between the trajectory angle which is the predicted one and the line of ball movement which is the true one and calculated the error between the angle.

## SETUP PROCESS

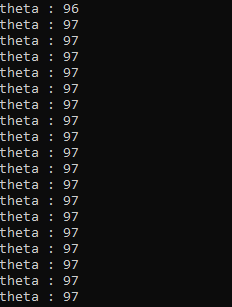


Figure 4.1: The cue angle

|  |  |
| --- | --- |
| **Angle** | **Θ(average)** |
| Between cue line and table border (obtuse) |  |

Figure 4.1 show the recorded angle of the cue from the baseline which is maintain at

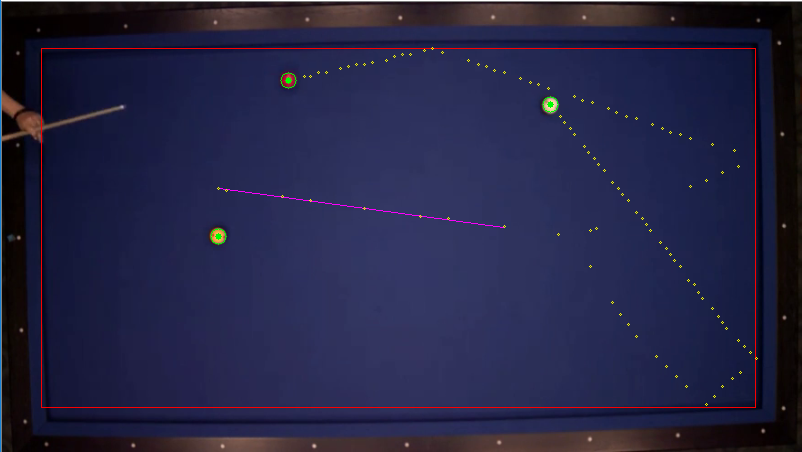
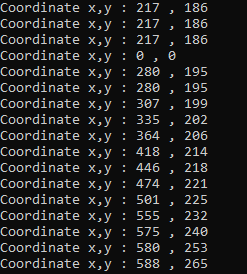


Figure 4.2: Line of ball movement

Figure 4.2 show the line of movement ball (in magenta colour). This line specially drawn to make a benchmark line and compare with the predicted line. To make sure the result stay authentic and accurate as possible, we draw the line by using the point of initial position of the ball and the last point of the ball before it change its direction.

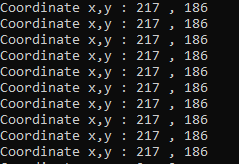


Figure 4.4: Final ball position

Figure 4.3: Initial ball position

Figure 4.3 and figure 4.4 show that the point has been taken to draw the tru straight line movement. The straight line using the point (217,186) for initial position and (501,225) as final position.

## trajectory COMPARISON

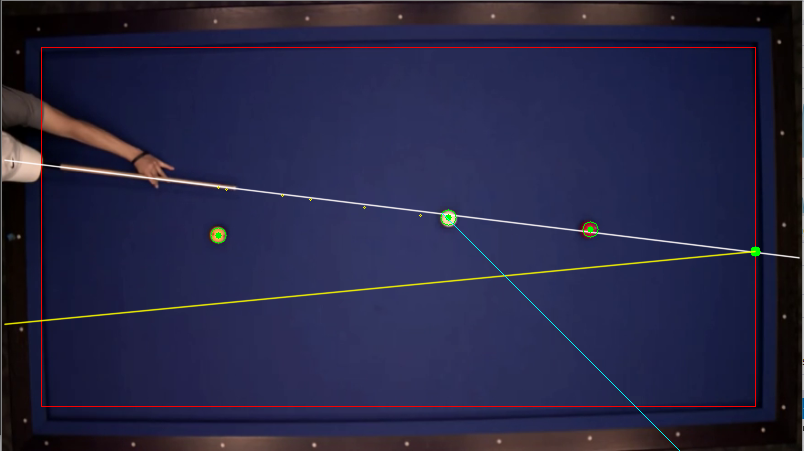
## 4.3.1 between ball in static condition and moving condition

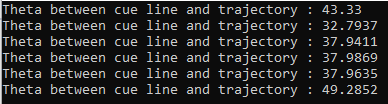




|  |  |
| --- | --- |
| **Angle** | **θ** |
| Between ball in static condition |  |

Figure 4.5: Theta between cue line and trajectory - ball in static position





|  |  |
| --- | --- |
| **Angle** | **Θ (average)** |
| Between ball in moving condition |  |

Figure 4.6: Theta between cue line and trajectory - ball in dynamic position

Figure 4.5 and figure 4.6 show the different result although the ball is moving in straight line and the player is in static position. The biggest factor that contributed to that result of course the quality of the video itself. Low quality of video affect detection process to the object. From the observation, moving ball also affect the result. The center of the ball like shaking and affects the predicted line.

* Theta between cue line and trajectory **-** ball in static position (average angle):
* Theta between cue line and trajectory - ball in dynamic position (average angle):
* Different between angle:

## 4.3.2 between cue line and ball movement line

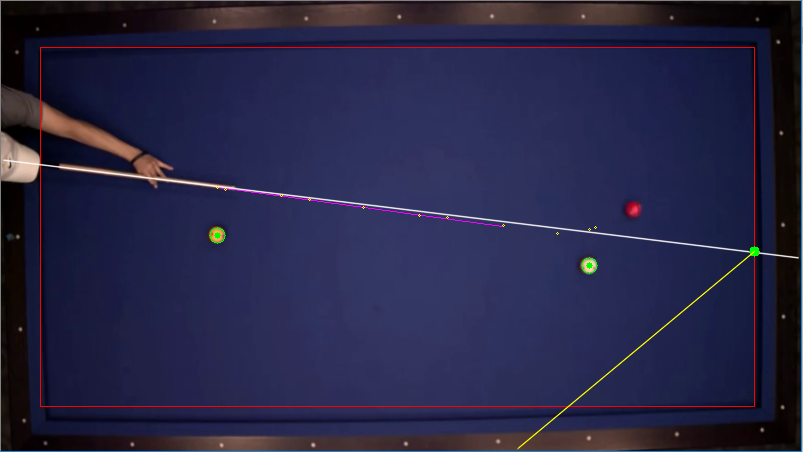
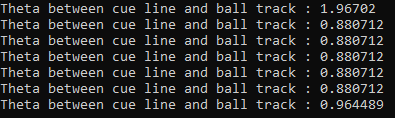


Figure 4.7: Angle between cue line and movement line

|  |  |
| --- | --- |
| **Angle** | **Θ (average)** |
| Between cue line and ball movement line |  |

Figure 4.7 show the angle gap between the cue line and the movement line. The expected result should be as the ball moving in straight line.

Average angle:

Error: | - | =

## 4.3.2 BETWEEN PREDICTED TRAJECTORY AND TRUE TRAJECTORY

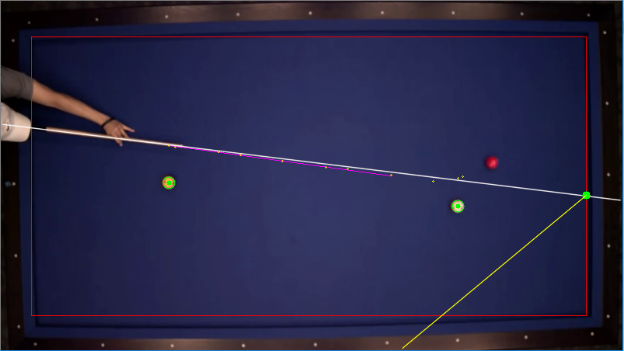
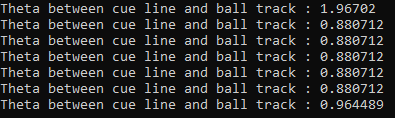


Figure 4.9: True angle

Figure 4.8: Predicted angle



|  |  |
| --- | --- |
| **Angle** | **Θ (average)** |
| Predicted trajectory |  |
| True trajectory |  |

Angle of predicted trajectory:

Angle of true trajectory:

Error:

Figure 4.8 and Figure 4.9 shows the main test of this project which is the comparison between the predicted and true trajectories. From the observation, the error for almost 9 degree is quite big.

## 4.3.3 ANGLE OF REFLECTION

# 

Figure 4.10: Angle of reflection

|  |  |
| --- | --- |
| **Angle** | **Θ (average) – Acute angle** |
| Source line (white) |  |
| Reflection line (yellow) |  |

Figure 4.10 showing angle of reflection. As for result, the acute angle between those two lines with the table border should be the same.

## 4.4 SUMMARY

As show in the results of the trajectories the same, it can be concluded that the result is not too impressive. One of the problems to pulled out the best result from this project is because of the quality of the video of itself which is not in high definition quality. That why some detection is quite disappointment. Another problem is because of the simplicity of the coding because of some limitation. At the first place, the coding has been set which is the hit will always to the center of the ball. So, Data of trajectories already lost some information about the rotation of the ball, and that make error to the calculation. The improvement of the trajectories can been make by considering of this two factors.

# chapter 5

**CONCLUSION AND FUTURE WORK**

# 

## CONCLUSION

As a conclusion, merging engineering skills with the sport entertainment is surely cool. As the project finish, I could say that there is many improvement that can be make especially in coding part. However, this project surely has its own values. In term of result, the value is not as perfect as its expected but the program is working as it should although there are many limitation and challenges.

## LIMITATION AND CHALLENGES

In completing this project, there are some limitations and challenges has been faces. The biggest challenges to handle is to make the object detection from the low quality of image. Firstly, it quite hard to perform the detection because the colour of the object is not solid which is need to much work to figure out the correct value. This situation also happen with different method of detection. Second, using C++ programming quiet limiting the project source. Availability of reference is more on python language than C++. Lastly, of course the level of programming skill. Lack of knowledge in programming skill make the objective of the program can’t be achieved or slow to it success. More or less it’s don’t give freedom in coding and to complete the task.

## FUTURE WORK

There are so much room for improvement in the project. The one that should have continuation in the project of course the table boundaries detection. It should have its own research work specifically on that part only because the project will involve with distortion image, image segmentation and many more. Generally, the existing coding surely has many flaws. That weak spot should be improve from time to time along the improvement of the result.

# BIBLIOGRAPHY

Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer Vision with the OpenCV Library. Sebastopol, CA: O'Reilly Media.

Cavallaro, R., Hybinette, M., White, M., & Balch, T. (2011). Augmenting Live Broadcast Sports with 3D Tracking Information. IEEE Multimedia, 18(4), 38-47. doi:10.1109/mmul.2011.61

M. A., M. M., M. W., & A. Y. (2012). Precision Pool-Aid. *Precision Pool-Aid,*1-51. Retrieved November 15, 2012.

Hong, K. (2016). [Photograph]. Retrieved from https://www.bogotobogo.com/python/OpenCV\_Python/python\_opencv3\_Image\_Canny\_Edge\_Detection.php

Jiang, R., Parry, M. L., Legg, P. A., Chung, D. H., & Griffiths, I. W. (2013). Automated 3-D Animation From Snooker Videos With Information-Theoretical Optimization. IEEE Transactions on Computational Intelligence and AI in Games, 5(4), 337-345. doi:10.1109/tciaig.2013.2275164

Li Dun, J. (2018). [Photograph]. Retrieved from https://stackoverflow.com/questions/48528754/what-are-recommended-color-spaces-for-detecting-orange-color-in-open-cv?noredirect=1&lq=1://

Ling, Y., Li, S., Xu, P., & Zhou, B. (2012). The detection of multi-objective billiards in snooker game video. 2012 Third International Conference on Intelligent Control and Information Processing. doi:10.1109/icicip.2012.6391406

Mallick, S. (2014). [photograph]. Retrieved from <http://opencvexamples.blogspot.com/2013/10/line-detection-by-hough-line-transform.html>

Reflection of a point about a line in C. (2017, July 06). Retrieved from <https://www.geeksforgeeks.org/reflection-point-line-c/>

Quadratic Equations. (n.d.). Retrieved from <https://www.mathsisfun.com/algebra/quadratic-equation.html>

Math, C. (2014, November 28). Coding Math: Episode 32 - Line Intersections Part I. Retrieved from https://www.youtube.com/watch?v=4bIsntTiKfM

Parker, J. R. (1993). Practical Computer Vision Using C. Hoboken, NJ: Wiley.

Parry, M. L., Legg, P. A., Chung, D. H., Griffiths, I. W., & Min Chen. (2011). Hierarchical Event Selection for Video Storyboards with a Case Study on Snooker Video Visualization. IEEE Transactions on Visualization and Computer Graphics, 17(12), 1747-1756. doi:10.1109/tvcg.2011.208

Russ, J. C. (1995). The image processing handheld (2nd ed.). Boca Raton, Ann Arbor, London, Tokyo, United State: CRC Press.

Shapiro, L. G., & Stockman, G. C. (2001). Computer vision. Upper Saddle River, New Jersey, United state: Prentice Hall.

Shen, W., & Wu, L. (2010). A method of billiard objects detection based on Snooker game video. 2010 2nd International Conference on Future Computer and Communication. doi:10.1109/icfcc.2010.5497393

Sousa, L., Alves, R., & Rodrigues, J. M. (2013). PoolLiveAid: Augmented reality pool table to assist inexperienced players. Computational Visual Media, 1(1).

Sousa, L., Alves, R., & Rodrigues, J. M. (2016). Augmented reality system to assist inexperienced pool players. Computational Visual Media, 2(2), 183-193. doi:10.1007/s41095-016-0047-3

# appendices

