Color Detectıon For Interactıon Mobıle Game Usıng OpenCV

Nadia Amira Md Kibri[1] and Mohd Shahrızal Sunar[1]

[1,2]Faculty of Computing, Universiti Teknologi Malaysia (UTM),
81310, Johor Bharu, Johor, Malaysia
{nadiamdkibri@gmail.com,shahrizal@utm.my}

Abstract. The best way for children to play while learning is through interacting with toys that can be held by them and this will not only can stimulate their senses, but also develop healthy and strong nerve connections in their brains. Between then and now, the children back then were using all sorts of games using their imagination and involving the physical activities compared to the children nowadays; everything is centered on their tablets and iPod. This new generation is not interested in going outside and plays without some sort of device attached to their hands. Because of that, a mobile application game that can allow user to interact with real object in real time is developed through this project where it use the color detection technique to recognize the object in the real world.

Keywords: Color detection, image processing, tangible play.

1 Introduction

In this modern world, mobile phones are really essential for everyone's life no matter their age is, including the children. In contrast to the past, the mobile phone nowadays has become a necessity in human life and it became one of the best platforms to play the game as it is small and easy to carry anywhere. Therefore, the children have been exposed to mobile games in early age and it has led to passive form of learning or entertainment among children. Instead of using their own imaginations to learn about the world and create something, children are passive recipients of visual and auditory stimulation that may or may not require some response using fine motor skills.

Between then and now, the children back then were using all sorts of games using their imagination and involving the physical activities. They like to play with their friends at outside of their house and allow them to socialize more with the people around them. But the children today are so different; everything is centred on their tablets and iPod. This new generation is not interested in going outside and plays without some sort of device attached to their hands. Because of that, they tend to not playing an outdoor game which is very helpful in their growth. This leads to lack of social interaction in children and can make them to not being active.

As we know, the technologies back then are not advanced as today, so we cannot blame the people nowadays to give their children an early exposure on current technology such as the mobile device. Because of we already have the advanced technology, why not we develop it again to improve flaws in the present. The technology now can allow us to combine the virtual world with the real world such as the Oculus VR that allow user to see in the virtual life, Leap Motion that allow user to use gesture motion to interact with the computer, and many more. However, these high devices are too expensive for the common people especially with the current economic situation are not stable and worsening.

Guided by these cases, this project has been developed where it can allow the children to interact with real world toys and help them to use their imagination to solve the problem. The game also has some rules that need to be followed by the children; time limited, score, etc. Therefore, this project will be combining the physical and digital play which the children will not only just playing using their tablets, but also using the object in real world. Furthermore, instead of only playing alone, the children can play with their friends too and of course, this will solve the problem of the children who lack in socializing with friends and passive children. The objectives of this project is to analyze the requirements of developing a virtual game that can interact with real world object in real time, to design and develop a game that use color detection and concept of reflection and to test the application that can allow the user to use real objects to play. This project will use a mirror to reflect the front's camera, some building block's shapes; and mobile application that will be implement into an Android's tablet. This concept has been used by Osmo game where its co-founder, Pramod Sharma, has sparked the idea to revolutionize the way of how children interact with the mobile screen to play.

2 Literature

2.1 Reflective Artificial Intelligence

Reflective Artificial Intelligence is a brand new genre of technology in our industry where the idea of this technology is to use a mirror in order to see what user do in front of the screen in real time (Choi, 2014). This technology is impressively closing the gap between traditional play-based that used physical tools and modern technology in the screen. This phase has been created to explain about a process of a new game that has been created named Osmo game where it allows the iPad to make senses of what it sees. According to an article in Osmo's blog, there are two main components of reflective artificial intelligence; mirror and computer vision algorithm. Mirror is used as a reflector to reflect the object on the player's based into the front-camera of the iPad. The reflector flips downward the camera view and transforms the iPad into a tangible interaction platform that allows the interaction between physical objects and the iPad screen. A computer vision algorithm is needed in order to make this reflective artificial intelligence work. The algorithm needs to work perfectly in order to process the data that it gets from the camera. Not only it need to work accurately, but the algorithm also have to response to the user in real-time.

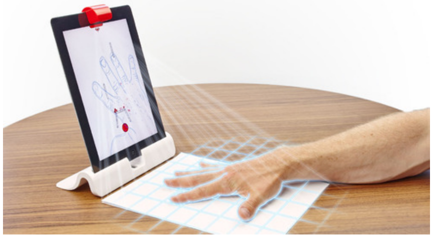


Figure 1 Osmo's gameplay and concept

2.2 OpenCV library

Based on Culjak et. al (2012), openCV is an open source library that is used to process the image and to analysis the video. OpenCV is actually comes from the word Open Source Computer Vision and was developed by Willow Garage, which is also the organization behind the famous Robot Operating System(ROS). Since it has been introduced, there are a numbers of programmers that contributed to improve this library. The openCV development has been funded by most of giant companies such as Intel, AMD and Google. OpenCV has been used extensively by people around the world whether they are a novice C++ programmer or a professional software developer. This library of programming functions is mainly aimed to allow user to create a real-time computer vision program or application. As it has been matured enough now, there are tons of collections of algorithms available in openCV dwarfs ready to be used by the developers. Most of the developers choose to use openCV because of openCV's focus performance is in the vision library, the application that is created using openCV can be embedded into the vision applications and mobile applications.

The CV from openCV is stand for computer vision where there are a lot of opinions on it and one of it is from Culjak et.al (2012) where they stated that computer vision is the combination of the science and the programming where it can process a computer to understand images and video. Thus, the computer can not only see the images, but they also can recognize it.

2.2.1 The way to use OpenCV

In order to use openCV library, user need to install the platform to use it first, such as Microsoft Visual Studio. Then, user need to download openCV library from its official website and run the installation setup. User also needs to setup the path in the environment variable in their computer/laptop. After all setup has done, the openCV library will be readily to use in their coding. In this library, all the classes and functions are defined within the name space cv. Therefore, the user can only access them by declaring using namespace cv; or prefix openCV class and function namespace specification cv:: and the main object is of class Mat. For an example, cv::Mat image will be creating an image size of 0x0

2.3 Object Detection

Nowadays, developers have developed a lot of attractive and sophisticated applications to facilitate our daily lives. Furthermore, Hirano et.al (2006) said that there are some industries such as Intel Company, has created a variety of computer vision products and service from the traditional area of machine inspection to more recent applications such as video surveillance, or face recognition. Based on the example, both of them are using the concept of object detection. Object detection is a basic computer technology that is related to computer vision and image processing where it can be defined as the task to find and identify objects in the real world from the image or video (Alhamzi et.al, 2014). Even though this area has been research for decades, it still has a lot of weakness and there are a lot of spaces that need improvements.

The aim of object detection is to detect and recognize the object that can be classified to a known class, such as people, cars or faces in an image or video. There are plenty of applications that use the concept of object detection such as face detection, people counting, vehicle detection and many more. These applications imposes various requirements and constraints on the object detection task such as the evaluation time, the accuracy to determine the object position, the reliability of recognition to reduce rates of "false alarms", and invariance where the algorithm need to be insensitive to the variance of the object (Treiber, 2010). Until now, we can see there are a lot of applications that implement object recognition such as shape, face, people, and vehicle and color detection.

2.3.1 Shape detection

Shape is commonly defined in terms of the set of contours that describe the boundary of an object. In object detection, shape detection is one of the popular techniques that have been used by the developers in their application. There is a technique by Toshev et. al, where they proposed a recognition method called as Boundary Structure Segmentation (BoSS). This method will relates the object detection where it is based on a novel holistic shape detector to figure segmentation and performs them simultaneously.

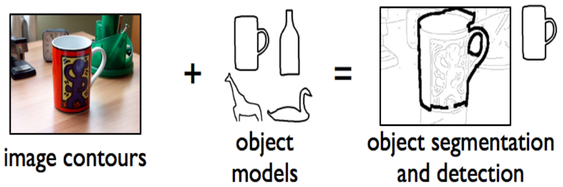
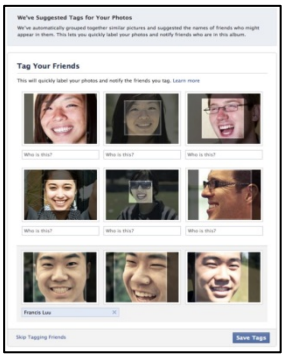


Figure 2. Example of the detection using BoSS to detect the shape of glass (Revised from Toshev, 2012).

BoSS will selects a foreground region while it is matching the input image with the object model, with some properties; similarity in shape and perceptual saliency. The properties of similarity in shape will be captured by a top-down process to exploit object-specific knowledge. While for the perceptual saliency, it will be captured by a bottom-up process based on general grouping principles, which apply to wide range of objects.

2.3.2 Face detection

As proposed from Ponce, et.al (2006), human face detection can be considered as a mature tool for the users, even though to get efficient face recognition, the progress must be made for full-profile view detection and accurate facial feature detection. Some of simple application of object detection that we can see in our daily life is in Facebook's new Tag suggestion. This new function can allow Facebook to suggest which friends are in their photo. Facebook are currently use facial recognition software that uses an algorithm to calculate a unique number based on user's facial features, for an example the distance between the eyes, nose, and ears. All of this information is gained based on their profile pictures and photos that they have been tagged to.



We've Suggested Tags for Your Photos

We've automatically grouped together similar pictures and suggested the names of friends who might appear in them. This lets you quickly label your photos and notify friends who are in this album.

Tag Your Friends

This will quickly label your photos and notify the friends you tag. Learn more

Francis Luu

Figure 3. Facebook Tag's suggestion that use facial recognition system

2.3.3 Color detection

Marini, A., et al (2015), has used color detection technique to classify the birds species based on its color features that is extracted from unconstrained images where the birds may appear in various scenarios with different poses, sizes and angles of view as illustrated in Figure 2.4. In addition, the images also present a solid variation in illuminations and some portions of the birds that is occluded by other elements of the scenario.

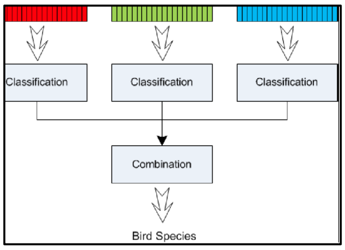


Figure 4. An overview of the proposed approach by Marini, et al (2015) for bird classification based on images

The first proposed step is to apply a color segmentation algorithm in order to remove the background elements and to limit the regions where the bird is presented as the image as illustrated in Figure 2.5. After that, the image is divided into component planes where the normalized color histograms are computed from these candidate regions at each plane.

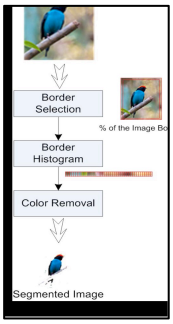


Figure 5. Details of the color segmentation approach

3 Implementation Of Propose Solution

The previous topic has been stated that the concept for this game application is color detection of the real world object. The concept is needed in order to make sure that the user will be able to play the color detection application through a game mechanic.
After that, as for the gameplay, it uses the technique of sorting the real world shape into the outline shape given in the application. As illustrated in the Figure 2.6, the framework of the game application described the whole flow of the game.

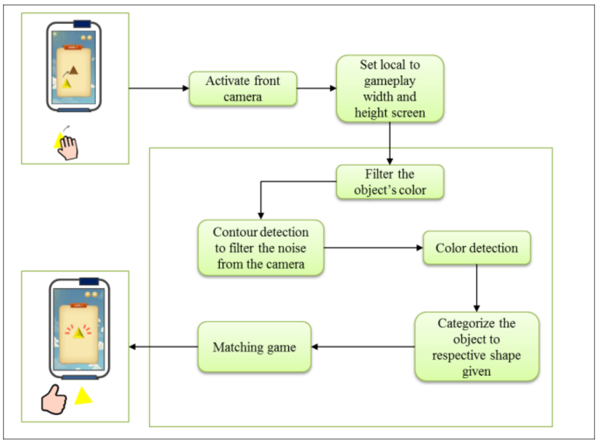
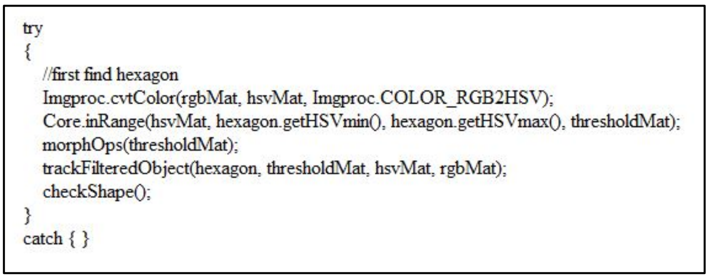

Activate front camera
Set local to gameplay width and height screen
Filter the object's color
Contour detection to filter the noise from the camera
Color detection
Categorize the object to respective shape given
Matching game

Figure 6. 2D game application Framework

3.1 Implementing the color detection concept

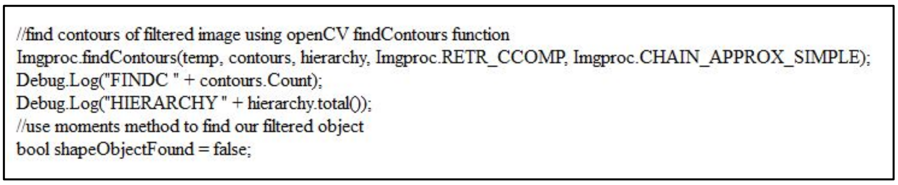
The application must allow the user to play the 2D game application using the real world object where to play the 2D game, user need to use several shapes to fit into the outline shapes given in the application. So, to make sure the application can recognize which color of the shape has been used by the user in the real world, the front camera of the tablet must be activated first. After that, while the camera is playing and the program in the Update() function, it filters the color of the shape regarding the shape that the player shows at the gameplay area in the real world. At this step, the program will try to find the threshold of the object/shape, and find the minimum and maximum Hue Saturate Value (HSV) of the shape's color to categorize it in the respective group of color.



```
try
{
   //first find hexagon
   Imgproc.cvtColor(rgbMat, hsvMat, Imgproc.COLOR_RGB2HSV);
   Core.inRange(hsvMat, hexagon.getHSVmin(), hexagon.getHSVmax(), thresholdMat);
   morphOps(thresholdMat);
   trackFilteredObject(hexagon, thresholdMat, hsvMat, rgbMat);
   checkShape();
}
catch { }
```

Figure 7. Example of the code to find the first shape's color which is Hexagon

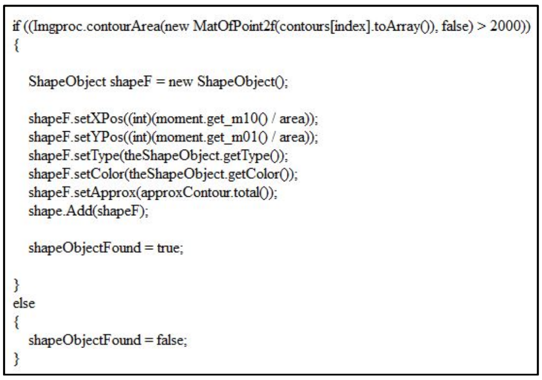
Next, the most important part of this program is to track the filtered object that is obtained from the previous steps. The Figure 8 below is the part of the trackFilteredObject() function from the program which is to find the contour of the filtered object. In order to find the object's contour, the program uses findContour() function from the OpenCV library and store it into the hierarchy variable.



```
//find contours of filtered image using openCV findContours function
Imgproc.findContours(temp, contours, hierarchy, Imgproc.RETR_CCOMP, Imgproc.CHAIN_APPROX_SIMPLE);
Debug.Log("FINDC " + contours.Count);
Debug.Log("HIERARCHY " + hierarchy.total());
//use moments method to find our filtered object
bool shapeObjectFound = false;
```

Figure 8. Part of the code to find the contour of the filtered object

After that, the total number of the object that is gained from the findContour() function is compared to the predetermined maximum number. If the number of the object is greater than the set maximum number, it will consider the object as the noise image, otherwise the program will continue to find the moment of the object and set the object found as true. The moment of object is needed to get the value of object's area where the area of the object will be used to determine whether the object is the right object or just a bad filter. Furthermore, using the moment of the object's area, the program also obtained the center point (x,y) of the object that is used for the shape's sprite in the application.



```
if ((Imgproc.contourArea(new MatOfPoint2f(contours[index].toArray()), false) > 2000))
{

   ShapeObject shapeF = new ShapeObject();

   shapeF.setXPos((int)(moment.get_m10() / area));
   shapeF.setYPos((int)(moment.get_m01() / area));
   shapeF.setType(theShapeObject.getType());
   shapeF.setColor(theShapeObject.getColor());
   shapeF.setApprox(approxContour.total());
   shape.Add(shapeF);

   shapeObjectFound = true;

}
else
{
   shapeObjectFound = false;
}
```

Figure 9. Code to find the object's center using the moment of the object's area

3.2 Matching game

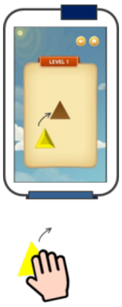
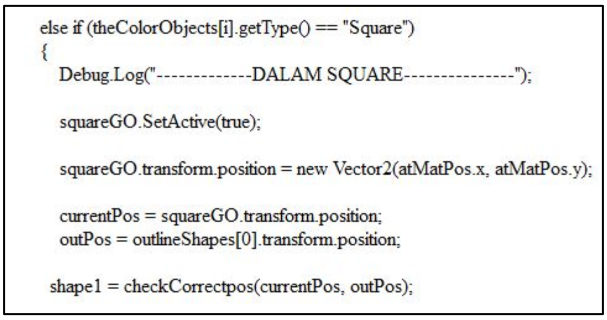


Figure 10. Example of the gameplay for the matching game.

Figure 10 is the example of the gameplay area for the matching game where the user needs to sort the shape into the outline shape. Then, the program get the type of the object based on the color that has been set before, in this case is the object type equals to “Square”. As shown in Figure 11 when the program is inside the Square condition, it will activate the Square shape’s game object inside the gameplay and use the position that is gained from the previous step as its current position. Next, it checks whether the object’s current position is equal to the outline position, if it is same, then the program display the win panel.



```
else if (theColorObjects[i].getType() == "Square")
{
  Debug.Log("-------------DALAM SQUARE---------------");

  squareGO.SetActive(true);

  squareGO.transform.position = new Vector2(atMatPos.x, atMatPos.y);

  currentPos = squareGO.transform.position;
  outPos = outlineShapes[0].transform.position;

 shape1 = checkCorrectpos(currentPos, outPos);
```

Figure 11. Match the square shape’s code

4 Result

4.1 Performance Testing

The performance testing has been conducted in order to know how good the application performance is in different devices. During the performance testing, the application has been integrated into several devices to see and compare the performance of the application through those devices.

Table 1. Table performance testing between different devices

No.	Task	Device	Final Result
1.	Able to recognize all the shapes	**Asus ZenPad** C7.0 (z170CG) Front camera: 0.3MP RAM: 1GB	Blue can be detected in any condition of light. When in low light, it detect yellow as orange, orange as red, green as blue
		Lenovo Tab 3 7 Front camera:2MP RAM: 2GB	Can detect Red, Blue and Green. However, sometime the application cannot differ the range color between green and blue. It detects orange as red.
		Samsung Galaxy Tab S2 Front camera: 2.1MP	Can detect all the colors

		RAM: 3GB	
2.	Time taken to change the location of the shape from first to second point	**Asus ZenPad** C7.0 (z170CG) Front camera: 0.3MP RAM: 1GB	Take <=2.73s
		Lenovo Tab 3 7 Front camera:2MP RAM: 2GB	Take <=3.03s
		Samsung Galaxy Tab S2 Front camera: 2.1MP RAM: 3GB	Take <=0.5s
3.	Condition of the light	**Asus ZenPad** C7.0 (z170CG) Front camera: 0.3MP RAM: 1GB	Low light
		Lenovo Tab 3 7 Front camera:2MP RAM: 2GB	Normal light
		Samsung Galaxy Tab S2 Front camera: 2.1MP RAM: 3GB	Normal light

4.2 Usability Acceptance Testing

The usability acceptance testing is also known as “Beta Testing” is commonly conducted in order to find bugs in the application. This type of testing is usually done near the end of the development process. Thus, the black box testing was conducted to obtain the user acceptance testing using Samsung Galaxy Tab S2 device.

Table 2 Black Box Testing

No.	Acceptance Requirement	Expected Result	Final Result
1.	The application must navigate the button to the exact scene.	Able to show the right scene whenever the button is tapped.	Able to show the right scene.
2.	The application must able to recognize the shapes by identifying the color of the shapes.	Able to recognize all the declare shapes.	Certain device cannot be able to recognize some shapes. But using Samsung Tab S2, all the devices can be detected.
3.	The application must able to track the location of the shapes in less than 5fps	The application able to track the location within 5fps.	Certain device can track the location of the shapes in 3 seconds, some in 2 seconds and 0.5 seconds. Using the Samsung Tab S2, the time taken is only in 0.5 seconds.
4.	The application must able to count the time.	Able to count the time.	Able to count the time in unit seconds.

5.	The application must able to give the star based on the user's performance.	Able to give stars according to the condition. 3 stars within 15 seconds, 2 stars within 35 seconds and 1 star within 55 seconds.	Able to give stars according to the given condition.

4.3 Usability Testing

The usability testing is used to test whether there is any problem when the end user uses the application. Besides, this testing is conducted in order to know whether the application is attractive and convenient, which would fulfill the user's needs. The methodology that has been used in order to collect the data is using interview's method. This interview includes the total of 13 novice respondents aged five years old from Tadika Ihsan UTM. Before the interview starts, the respondents have been asked whether they have any experience of using or playing any mobile application. Based on the Figure 12 below, there are 4 respondents, 3 female and 1 male that never use or play the mobile application while the rest of them have the experience on using it.

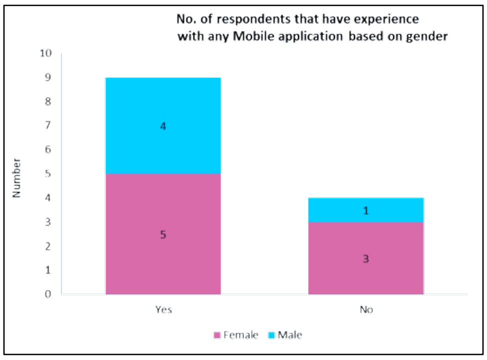


Figure 12. The bar chart to show the number of respondents that have experience using mobile application

Then, the respondents need to start the application on their own until the gameplay is shown. The testing is needed in order to know whether the graphical user interface (GUI) button is easy to follow by the focus user who aged 5 years old. Based on the Figure 13, there are seven respondents who strongly agree that the button is easy to use, three more respondents find it is normal where they can find all the button easier except to one button which is next button, and three more respondents disagree that the GUI button is easy to follow as it was their first time to use the mobile application. However, they manage to choose the right button to play after being asked.

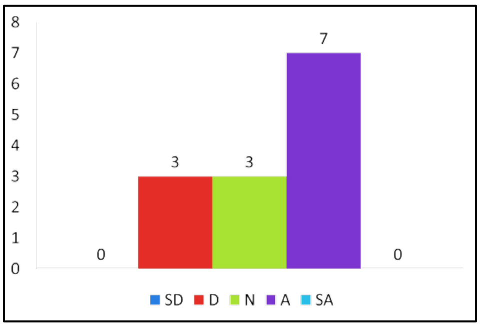


Figure 13. The bar chart depicts the respondents respond on the easiness of GUI button's navigation

Next, the respondents also play the application to know whether or not this application can be used by the focus group. Based on the testing, all of the respondents only know the basic shapes which are circle, square and triangle. However, the respondents have already known to match the same shapes even though they did not know the name of the shapes but it still affect the time to solve the problem. In order to know whether the problems given in the application can be solved by the user, three level of the game has been tested by the user individually where the number of shapes inside every level is different. After testing the game individually, the respondents need to solve the problem in a group, to compare the time solved of the problem in a group and individually. Figure 14 below shows the average of comparison between when the respondents play individually with when in a group.

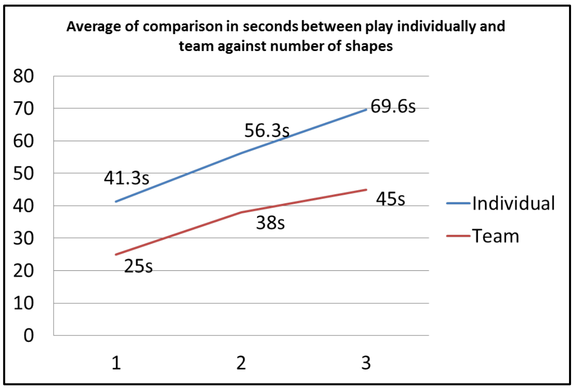


Figure 14. The graph shows the average comparison between time to complete (seconds) play individually and in team, to number of shapes

Based on the Figure 4.3, it shows that the time to complete the problem for every level tested when the respondents played in a group are shorter than when they play it individually with the average of reducing time equals to 19.7s. This is because, when they play the game in a group, they tend to discuss among each other on how to sort the shapes into the right location, while when they play the game individually, they still did not able to sort the shapes correctly on their own.

5 Conclusion

5.1 Achievements

The process of developing this application has followed the objectives of the project where the first objective is to analyze the requirements of developing a virtual game that can interact with the real world object. Guide by this objective, all the requirements to create the application that can be played using the real world object has been study. In order to make the application able to interact with the real world object, the application requires running inside the tablet as the tablet's height allows the camera to see the whole gameplay area compared with the normal mobile phone.Next objective of this project is to design and develop a game that use color detection and concept of reflection. During the development of the application, the knowledge that is needed in order to build an application that can interact with the real world object has known which is using the color detection technique to detect the object while using the mirror to reflect the gameplay area in the real world.The third objective is to test the application that can allow the user to use the real objects to play. During the user testing phase, 13 respondents who aged 5 years old has tested the application as the focus group for this application is kids from aged 5 years old. The user testing has been conducted to know whether the application is suitable to be used by the focus user.

5.2 Game Limitations

In this game development project, there are several game limitations that have been detected. Firstly, the application's performance does not constant in every device that has been tested. Although some of the shapes can be detected, there are several shapes that cannot or required low or high lighting condition. Furthermore, sometimes the application will detect user's skin as an orange color which cause the shape with orange color is displayed in the gameplay. This may be due to orange color with skin tones are within the same range of color. Then, the application is not capable to rotate the shapes when the user is rotating the shape in the real world. During the development phase, the rotation function has been tried to be embedded in this application using the edge detection technique but unfortunately, it does not suitable as the object has different shapes and the technique is not too stable to be used to detect the real world object.

5.4 Future Works

The result of this game development project provides potential insights where there are some suggestions for the future development that can improve this application. First of all, the application should be able to rotate the shapes according to what the user is doing to the shapes in the real world. If this function can be accomplished, the outline of shapes in the application can be varied and the level of difficulty can be increased, so that the target group of user can be wide-ranging. Next, the current application depends on the value of the tablet's front camera. So, this application should be improved to ensure that the application does not depend on the front camera's value so that every types of tablet can use this application. Furthermore, the application can also be applied into the edutainment game to teach kids about the type of the shapes. This is because during the user testing, the respondents; consists of 5 years old children, still does not know the types of the shapes except the basic shapes such as circle, triangle and square.

References

1. Alhamzi, K., Elmogy, M., and Barakat, S.: 3D Object Recognition Based on Image Features: A Survey. (2014)
2. Choi, J.: Give Your Ipad The Power Of Sight…For Fun, Forbes.com, `http://www.forbes.com/sites/jennchoi/2014/05/22/give-your-ipad-the-power-of-sight/#6b77607f15b1` (2014)
3. Culjak, I., Abram, D., Pribanic, T., Dzapo, H., & Cifrek, M.,: A brief introduction to OpenCV. In: MIPRO, 2012 Proceedings of the 35th International Convention (2012)
4. Hirano, Y., Garcia, C., Sukthankar, R., & Hoogs, A.: Industry and object recognition: Applications, applied research and challenges. In: Toward Category-Level Object Recognition, pp. 49 - 64. (2006)
5. Marini, A.,Turatti, A. J., Britto, A. S., Koerich., A. L.: Visual and acoustic identification of bird species. In: Acoustics Speech and Signal Processing (ICASSP) 2015 IEEE International Conference, pp. 2309-2313 (2015)
6. Ponce, J., Hebert, M., Schmid, C. And Zisserman, A.,: Toward Category-Level Object Recognition. 1st ed. Berlin [u.a.]: Springer (2006)
7. Toshev, A., Taskar, B., & Daniilidis, K.: Shape-based object detection via boundary structure segmentation. In: International journal of computer vision, 99(2), 123-146 (2012)
8. Treiber, M.: An Introduction To Object Recognition. 1st ed. London: Springer (2010)