

Image and Object Recognition System Apps for Kids Learning

Nor Muhammad Wafiz Hamka Noruddin¹, Muhammad Anas Razali^{1*}

¹Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, 86400, Batu Pahat, Johor, MALAYSIA

*Corresponding Author Designation

DOI: <https://doi.org/10.30880/eeee.2021.02.02.033>

Received 04 July 2021; Accepted 14 August 2021; Available online 30 October 2021

Abstract: Deep learning is an artificial intelligence branch utilizing deep neural network systems that are highly specialized in machine vision, voice recognition, natural language, and other areas. Google has launched TensorFlow, a deep learning software library for open source definition, training, and implementation of Models in November 2015. In recent years, deep learning technology has developed significantly. Deep learning and object recognition is the use of artificial intelligence (AI) on recognizing obstacles, object identifying, and even traffic lights to improve quality in the processing system. The use of AI eliminates the need for human work in the sectors. This project is developed using AI to acts as a helper for parents and teachers to develop kids' growth. Tensorflow is implemented into this project to detect, training and process the object that it sees. Furthermore, Google Collab is used as a processing cloud because of its free cloud GPU usage. Finally, Android Studio is used to build the Android Application Package (APK) for the phone.

Keywords: Deep Learning, Object Recognition, Artificial Intelligence

1. Introduction

According to GSMA's real-time intelligence information, there are 5.22 billion mobile devices in the world today. This means that 66.83% of the world's population has mobile devices. Back in 2017, the number of people using mobile devices was just 53, breaching the 5 billion barriers. Statista estimates that this range of mobile device users will rise to 7.33 billion by 2023.

The new survey showed that 56% of children between 8 and 12 years of age had mobile phones. It is also more interesting to realize that twenty-one percent of children eight or younger use a mobile. Families of children aged eight or younger age have sixty-nine percent of the time a mobile.

This showed that most of the kids in 2020 use mobile phones in their daily day. Studies said the average screen time for kids under 9 is 48 minutes. They usually used their mobile phone for gaming

*Corresponding author: anas@uthm.edu.my

2021 UTHM Publisher. All rights reserved.

publisher.uthm.edu.my/periodicals/index.php/eeee

youtube, and social media. This showed that with the average screen time educational department can manipulate this data to pique the kids' interest to improve and develop their memory and imagination.

When children have free time at home, often they only play games or watch videos from the internet, but with the product, they enjoy playing and improving their memories and creativity at the same time. Especially during this pandemic, children are forced to stay at home to giving them a lot of free time. This product will aid to help parents improve the memories of their children [1].

The main objective of this project is to design a educational system of object detection that can assist parents and teachers on educating their children. Next is to develop a system that smart enough to recognize various type of objects and its classes such as kitchen and, animal or pet classes. The third objective is to design a real-time image and object recognition. Lastly, to develop an object recognition application so that the user can carry it around, for example Android Application (APK).

This project aims to be able to detect various kinds of objects and automatically recognize the object and show how they spell. Next, this project is an alternative way for kids to recognize and learn the objects.

There are several scopes of the research in this project such as, to create a simple platform for android so that it can be accessed more easily. The second scope is the system can be easily accessed by the user through their own phone. Next, to ensure the system to have at least 90% accuracy when recognizing the object. The final scope is to used the Google Collab as a cloud server to make deep learning run smoother and does not restricted by the device.

The research contribution for this project is it can implement into home and school education. It helps parents and teachers to develop kids' memory and imagination. It also serves as an alternative way for kids to learn.

2. Materials and Methods

This system is proposed to acts as an education helper that assist parents and teachers. This system is based on a phone application designed to introduce kids to recognize objects. A phone camera is used to detect the object and feeds it into the algorithm process. Android phone is focused on this project. There was a few software is used which are Tensorflow, Google Collab, Coco, Android Studio, and, camera. Tensorflow is used as a processing and recognition image. Before TensorFlow, neural developers used other software such as Opencv, Torch, and Scikit. Now developers tend to use Tensorflow because of the various benefits such as its main language program in python and it is included in Google Cloud Platform. The Android Studio is used as a final step which is to covert this project to an APK file for the android phone.

2.1 Project Implantation

Figure 1 shows the system flowchart. When launching the program, the user will be greeted with two-mode which are image and video. User can freely select whatever mode that suits them. Then, the system will initiate a recognition process that consists of multiples layers of the algorithm, neural, boundary, and prediction. If the system completes the recognition process it will produce an image with text output for image mode or the real-time video with boundary layout for video mode.

Figure 2 shows the mechanism of voice assist image and object recognition apps for kids learning works. First, The camera is used as an input (image and object detector). Pretrained data acts as a reference for the system to identified and recognize objects. When an object is present inside the range of the camera, it will send data to the processing and recognition system and it will start to recognize the object. Tensorflow will be used as a processing system that manages the input data and produces object detection. Python functions as TensorFlow language to run the system. Google speech is used as an audio output for the system [2].

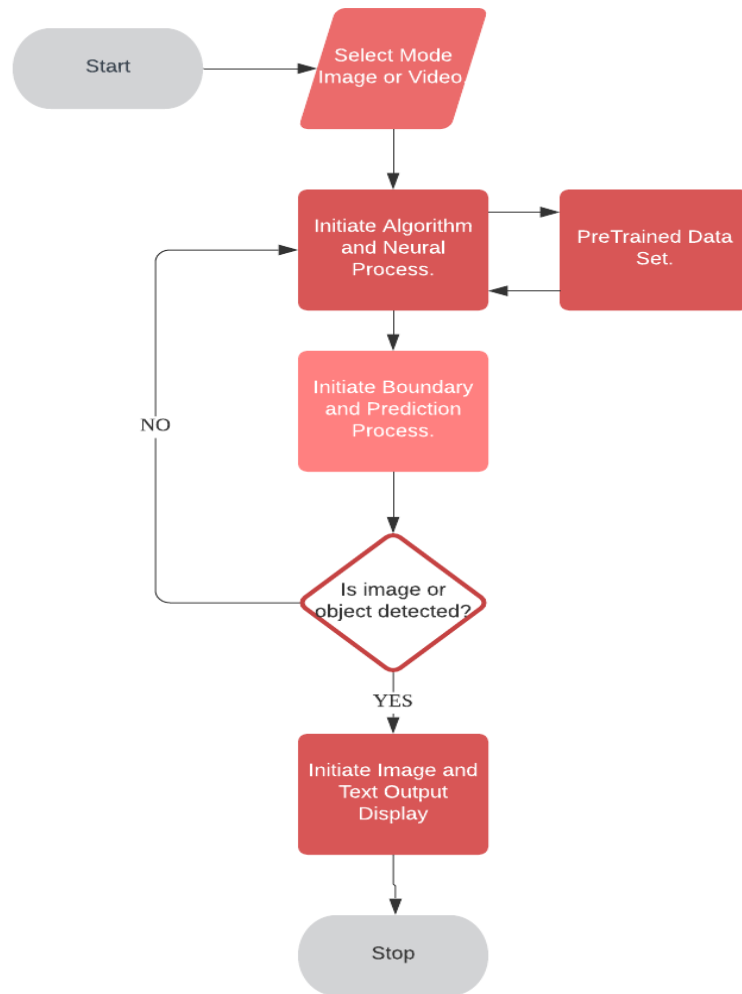


Figure 1: Process flowchart

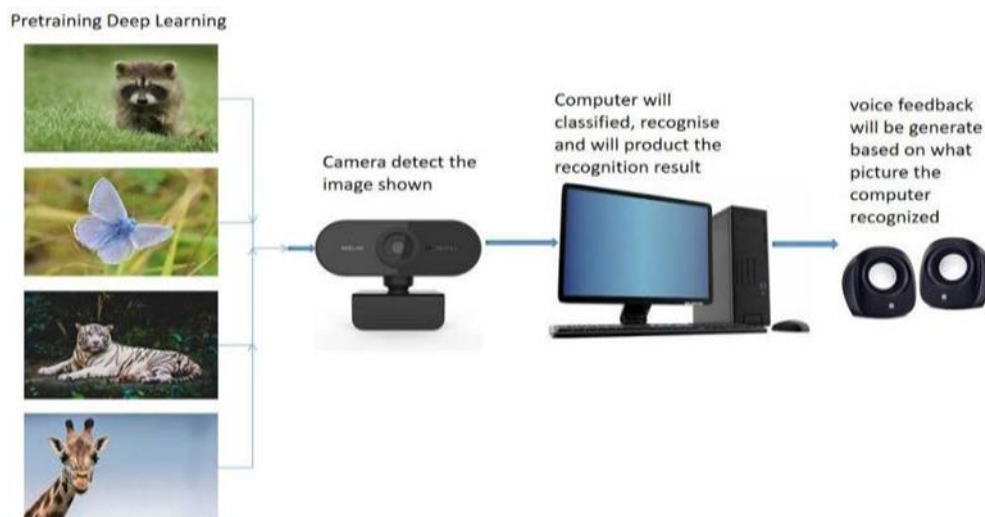


Figure 2: Mechanism of object recognition

3. Results and Discussion

This section is about the analysis of the functionality for each hardware and software used in this project. The experiment has been carried out to test the custom data set, the accuracy of the image detection and also to train the system to recognize at least 10 classes of objects.

3.1 Android Studio building

After a series of tests in the Google Collab, the next step is to implement it to the Android Studio, which then will be converted to an APK file (Android-based Application). This section will show all the processes from creating the commands or coding for the Android Studio until the building APK file process [3].

The first step is to set up the Android Studio before implementing all the coding that had been tested in Google Collab. The figure below shows on the step to set up the software. Figure 3 shows several options on how the Apps activity will be used and how it looks like.

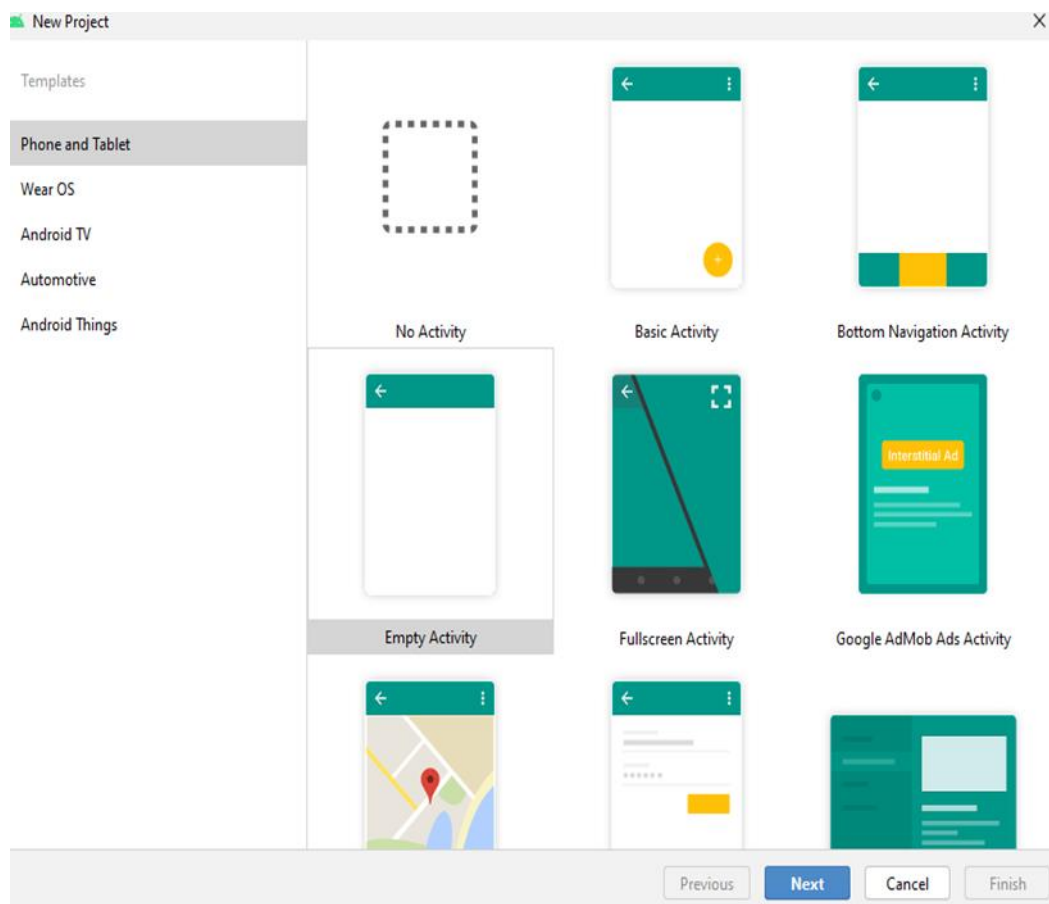


Figure 3: Activity set up for the recognition software

Figure 4 shows the section where the outlook for the software will be. In this section, the buttons, the output picture, and the result will be customized by the developer for this project itself.

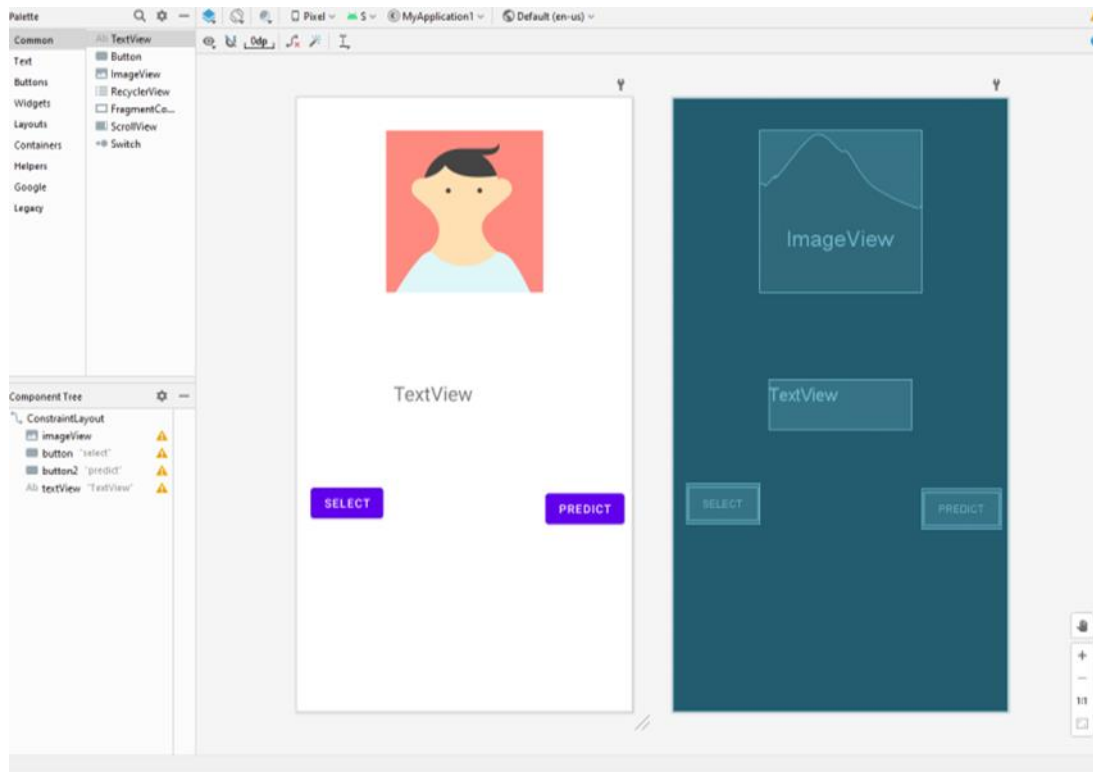


Figure 4: The outlook for the software

3.2 Set up for the implementation for Android phone

Accompanying discussions that further explain observations of the results are usually placed immediately below the results paragraph.

Several things need to be done before the test for the android phone. The first thing is to make sure that the mainframe does not have any errors or red lines in it. This is because any red lines could potentially be trouble during the APK building process. There also, the phone needs to be in developer mode. This is because the USB debugging option is only available if the phone in that mode. When all of this is done and checked, the test run for the system can be implemented into the phone.

To unlock the developer mode or dev mode, every android phone has the same way. This process starting with the setting menu, then go to the system option and tap at the about phone. By tapping the build number at the menu 7 to 10 times, a message will pop up that says “You are already a developer”. Lastly at the system menu, a developer option will show and tap at USB debugging [4].

After the checking red lines in the coding and unlocking the developer modes, the next step is to connect the phone using USB with USB debugging enable to the computer, Then, the Android Studio will detect the phone and show the phone model at the top right corner in the Android Studio software.

Figure 5 shows the phone is connected to the computer. This is because to test whether the system builds in the Android Studio works or not. The phone must be in developer mode so that the USB Debugger function will be enabled.

Figure 6 shows that the phone had been recognize by the Android Studio and the phone is ready to run the project. When clicking the green arrow will command the software to launch the project into the phone an notification will pop up when the launching or implementing is successful.

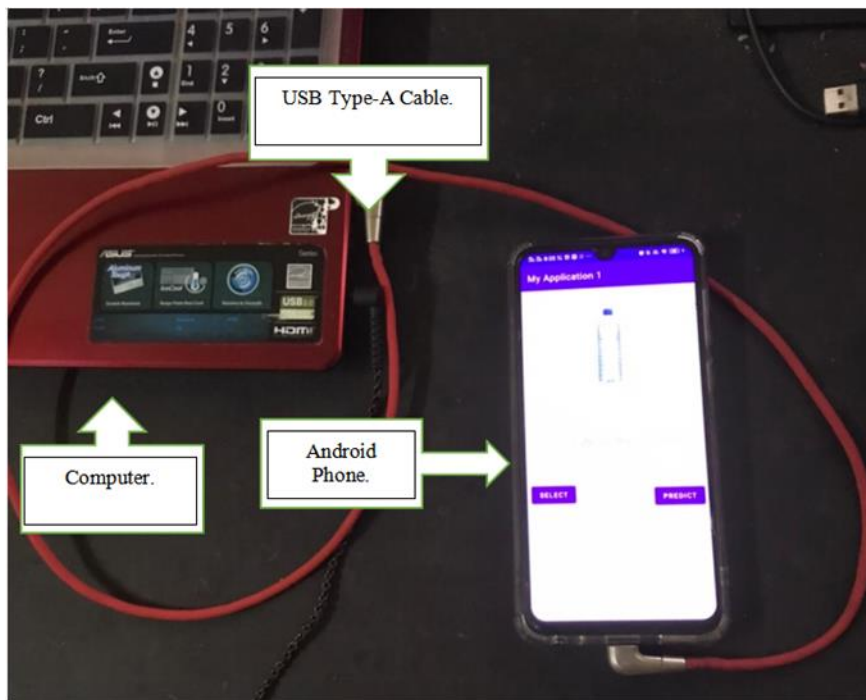


Figure 5: Phone connected to the system



Figure 6: The android studio recognise the phone

3.3 Building the APK file

After the successful of implementing the system to the phone via USB Debugger and Android Studio, the next step is to create the APK file for the Android phone so that it can be install and use without any USB cable. For the Android Studio to build the APK file is simple.

Figure 7 shows the options for Android Studio to build the APK file for Android phones. This process will take time depending on the computer processing power horse. This is the easiest part because the software will do everything and the user doesn't have to tweak anything.

Figure 8 shows the notification pop-up soon after the Android Studio successfully building the project system into the phone and the APK file will be saved at the Android Studio folder.

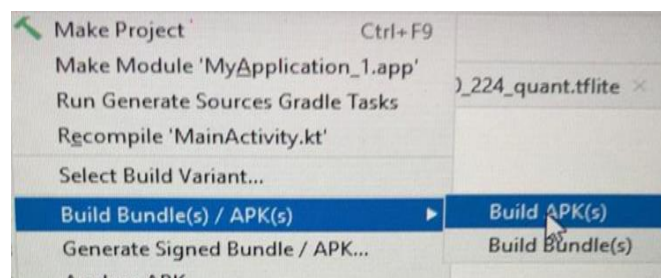


Figure 7: The android studio builds APK options

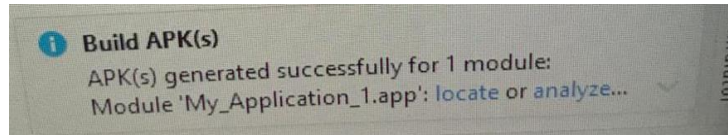


Figure 8: The notification that the APK is successful build

3.4 The output for the system in the phone

After the success of implement or launching the project system, the phone will automatically pop up the interface system which include the picture or video, the two button select or predict and the output result. Figure 9 and 10 show the result of run system with the picture and the text output [5].

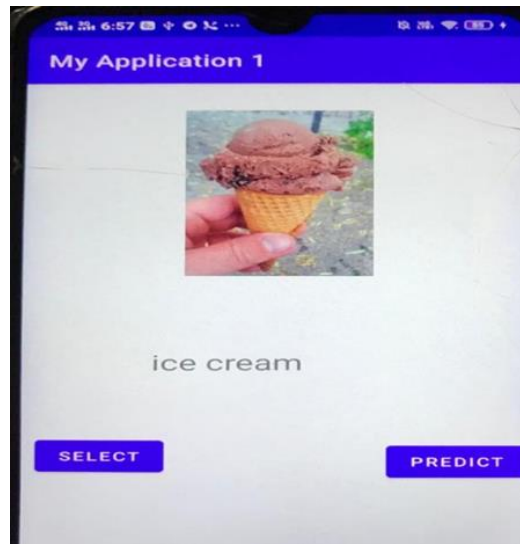


Figure 9: The interface shows up at the phone

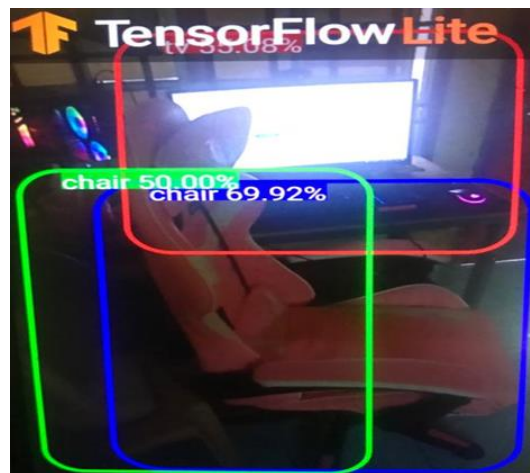


Figure 10: The video mode recognition process

After the success of implement or launching the project system, the phone will automatically pop up the interface system which include the picture or video, the two button select or predict and the output result. Figure 9 and 10 shows, the result of run system with the picture and the text output [5].

Figure 9 shows the interface pop-up soon after the Android Studio successfully implemented the project system into the phone. This interface was made to be simple so that even a child can understand and navigate thru the system.

There are 2 modes in this app which are the picture mode and the video mode. Picture mode is the user downloads a picture from the internet and asks the system to recognize it by pressing the select and predict button. This mode has more accuracy rate compare to video mode because the system only needs to process a standing still object and the pixel dimension and density are not too large.

Video mode is a mode that lets the system run a recognition process in real-time as shown at Figure 10. Due to this, the mode gave up a bit of accuracy and performance. This is because the system has to perform multiple processes at once. It has to continuously run an object recognition process like fetch the object data from the TensorFlow data set while maintaining accuracy. Thanks to the local data processing, the data transfer between input and output is less than 4 seconds for the image and video mode [6].

As shown in Figure 10, the accuracy score for the mode is a bit low. This is because there are few factors needed to take into count, which is the environment factor and the ambient light factor. Because the detection border or frame is already fixed at a position, the object environment such as how it was placed can be the cause for the accuracy scored to be a bit low. If the object is placed in an orthodox position the system will have difficulty recognizing it, thus it affects the score [7].

Same as the ambient light factor. The system relies on the ambient light itself to see the object. If the contrast is too low such as at night, the system will try to recognize the object at the least pixel spot. This is because the object currently in low night light, thus make the object it sees have unbalanced contrast. This also can be a factor for the decreased accuracy score [8].

The application does not require any internet connection because the baked-in recognition process is already in the apps. This gives the app's advantage in terms of performance and availability. Users can access and use this app anywhere. The downside for picture mode is even the apps do not require an internet connection, the user still has to download their preference picture first. Unlike the video mode, the user can launch it has the system will start the continuous recognition process at once [9].

For the image mode, the user can download any image without concerning the size or the type of the image is. This is because the system already has a feature that automatically changes the picture type and resizes the picture so that it can run the recognition process with its preferred size. These features help optimize the system recognition speed thus maintaining a higher accuracy score [10].

Aside from that, these apps are an open program. Thanks to the android platform. this means that any other people besides the developer can make an additional data set. This gives the children something to tweak when they arrive at the next stage of growth.

Figure 11 shows, another output detection from household classes which is a water bottle. This proves that the data set for different classes is working properly.

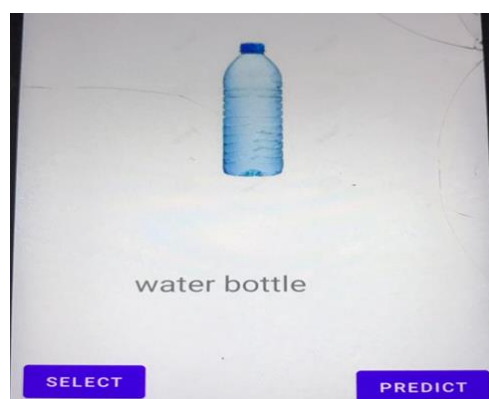


Figure 11: The output from another type of class

4. Conclusion

This project managed to achieve preliminary target that can work according to its general work flow which can detect image correctly with a high percentage of accuracy. Furthermore, this project work successfully on personal computer as well as android phone using Google Collab via Google search engine. Besides that, the system also successfully undergo a training state using the animal dataset from COCO. There are few issues that need to be solve to ensure that the system work properly such as the system detected too many object at the same time. Moreover, reading previous article, journals and project related to human, object and image detection help to complete this final year project.

Acknowledgement

The authors would like to thank the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia for its support.

References

- [1] "Text String Detection From Natural Scenes by Structure-Based Partition and Grouping," IEEE Transactions on Image Processing, Volume: 20 Issue: 9, 14 March 2011
- [2] "A recongnition system of real time paper currency", Proceedings of 2012 2nd International Conference on Computer Science and Network Technology, 10 June 2013
- [3] Build the Android App. In Android Continuous Integration (pp. 67-77). Apress, Berkeley, CA.
- [4] Wolfson & Felker, D. (2013). Android developer tools essentials: Android Studio to Zipalign. " O'Reilly Media, Inc."
- [5] Karaoglu, S., Van Gemert, J. C., & Gevers, T. (2012, October). Object reading: text recognition for object recognition. In European Conference on Computer Vision (pp. 456-465). Springer, Berlin, Heidelberg
- [6] Silberschatz, A., Stonebraker, M., & Ullman, J. (Eds.). (1991). Database systems: Achievements and opportunities. Communications of the ACM, 34(10), 110-120
- [7] Vogel-Ciernia, A., & Wood, M. A. (2014). Examining object location and object recognition memory in mice. Current protocols in neuroscience, 69(1), 8-31
- [8] Kalaiselvi, P., & Nithya, S. (2013). Face recognition system under varying lighting conditions. IOSR Journal of Computer Engineering (IOSRJCE), 14(3), 79-88
- [9] Choudhury, T., Borriello, G., Consolvo, S., Haehnel, D., Harrison, B., Hemingway & Wyatt, D. (2008). The mobile sensing platform: An embedded activity recognition system. IEEE Pervasive Computing, 7(2), 32-41
- [10] Song, J., & Lyu, M. R. (2005). A Hough transform based line recognition method utilizing both parameter space and image space. Pattern recognition, 38(4), 539-552