Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus

BY

Lee Mun Hong

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (Honours)

Faculty of Information and Communication Technology
(Kampar Campus)

JANUARY 2021

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Mobile Tour Guide Application with Attraction Recognition

for UTAR Kampar Campus

Academic Session: 1/2021

I LEE MUN HONG

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:

B,5,2 One Imperial,
Lintang Sungai Ara 14,
11900 Bayan Lepas, P.Pinang

Liew Soung Yue

Supervisor's name

Date: 14.4.2021

Date: 16/4/2021

Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus

BY

Lee Mun Hong

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (Honours)

Faculty of Information and Communication Technology

(Kampar Campus)

JANUARY 2021

DECLARATION OF ORIGINALITY

I declare that this report entitled “**Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ___________________________

Name : Lee Mun Hong______

Date : 14.04.2021_________________

ACKNOWLEDGEMENTS

I would like to express my thanks and show my appreciation to my supervisor, Dr Liew Soung Yue who had given me the opportunity to take his title which is Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus and made me learn more about training a model and developing an Android app. Dr Liew also guided me through this project so that I can complete it successfully.

ABSTRACT

UTAR Kampar Campus has a lot of unique and beautiful buildings and structures but many people seldom get the chance to know the histories and stories behind these buildings and structures. This is such a waste because most of the buildings in UTAR Kampar Campus is built with meanings and the designs are based on some unique ideas such as the Ling Liong Sik Hall, which resembles the Forbidden City Palace in China. This is because most of the people, even if they are interested, do not want to go through the hassle of searching for the web and digging for the information about the place. So, this project aims to resolve the issue. This project is to create an app that allows user to either take photo from their phone or select photo from gallery of a UTAR building or structure and the app will identify the building in the photo taken. Then, a brief history and description of the place will be shown. The app can accurately identify the buildings using a model trained by a machine learning algorithm, and GPS information to increase the accuracy. With this app, users can get to know the histories and stories of buildings and structures in UTAR Kampar Campus with ease without even having to search the Internet for information.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

GPS	Global Positioning System
ERD	Entity Relationship Diagram
CNN	Convolution Neural Network
SVM	Support Vector Machine
ORB	Oriented FAST and rotated BRIEF
API	Application Programming Interface
QR	Quick Response
UTAR	Universiti Tunku Abdul Rahman

CHAPTER 1: INTRODUCTION

1.1 Problem Statement

Nowadays, many people know UTAR has a very beautiful campus in Kampar but not many know about the stories behind it, especially the students. Actually, many thoughts and efforts had been put into designing and building the campus to make the buildings and structures look unique. However, people might not fully appreciate the uniqueness of it as they do not understand the stories behind those structures. This is not entirely the people's fault as not much effort had been put into publicizing the histories and stories of the buildings and the information are buried deep insight the internet.

1.2 Background and motivation

UTAR is a private university in Kampar that is launched on 13 August 2002. It has more than 22,000 students now, both Sungai Long and Kampar campus combined. Kampar is a town in Perak with beautiful scenery including beautiful mountains and lakes and as a small town, it has many old traditional shops that makes Kampar a very unique place. UTAR Kampar campus preserves most of the Kampar natural scenery in the campus too. The lakes and land are not over developed and along with the unique buildings in the campus such as Ling Liong Sik Hall makes UTAR Kampar one of the most beautiful campus in Malaysia.

Machine learning is a type of method that help software make decision or perform a task without explicit programming or rules. Machine learning rely more on data instead of coding. Data is fed into an algorithm and a model will be trained and be used in programs. For example, to train a model to identify images such as apple, a lot of apple pictures need to be gathered from different angle, lighting or even color to put into the algorithm. The algorithm will then extract the features on the pictures such as the round shape of apple along with other features and at this stage, the model is produced. When the model is put into an app, it will compare the features it learned from the image dataset with the image uploaded by user using the app. Using traditional rule-based coding, it is almost impossible to create an algorithm that identifies an image.

The motivation of this project is to create an app that delivers the histories and stories about UTAR buildings and structures to the users through an easier process. Previously, users need to dig through the internet to find the relevant information. With this app, users just need to take a photo of the building and the app can identify the building in UTAR Kampar campus and display the information about the building using a model trained by a machine learning algorithm coupled with GPS information for added accuracy. Through this app, more people can get to know about the stories of certain buildings by facilitating the delivery of the relevant information.

1.3 Objectives

The objectives of this project are to develop a mobile application that:

- Can accurately determine a place in UTAR Kampar Campus from a photo taken and provide sufficient information about the place to the user.
- Promote the histories and stories of UTAR buildings.
- Although this app utilizes GPS information, it does not provide the navigation around UTAR yet although it is possible to be implemented in the future work.

1.4 Proposed approach and study

This project consists of two parts. The first part is applying machine learning to train the model so that the app can identify the buildings in the photo. The second part is designing the app interface and integrating the model into the android app itself. For the first part which is training the model, Jupyter Notebook and TensorFlow is used. First, I need to gather the dataset by going to UTAR Kampar Campus to take the photo of the building by myself using my phone. As a result, around 80-100 photos were taken for each place. Then, the photos were input into the TensorFlow Lite Model Maker library and the model will be produced in the .tflite file format.

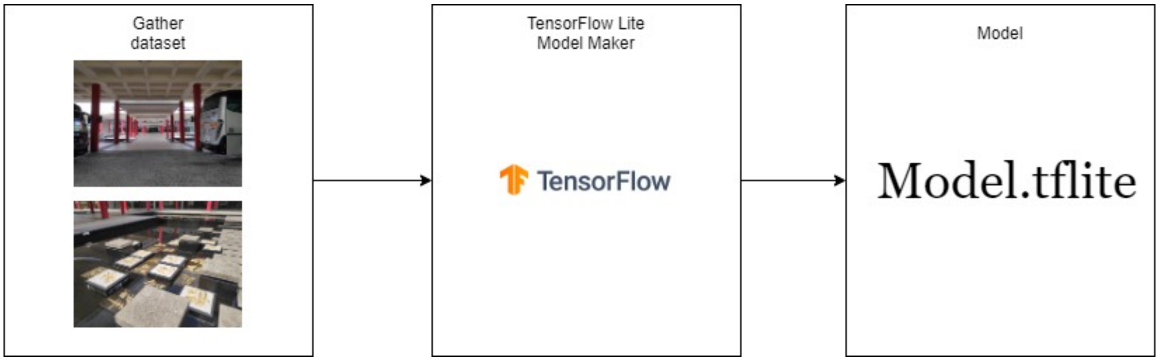


Figure 1.4.1 Machine learning flow

After the model have been produced, it will be imported to Android Studio where I developed the app. After importing, I can call the function provided by TensorFlow to identify the photo once user selected a photo. For this app, the programming language used is Java.

1.5 Highlight of what have been achieved

- An android app is developed for user using Java
- A model is produced using machine learning with TensorFlow

1.6 Report organization

This report consists of six chapters. The first chapter gives a brief introduction on the project, how the app works and the motivation behind developing this app. In the second chapter which is literature review, I review similar project that is done by other people on the internet and determine their strength and weakness. With that, I can refer their system and gather ideas from their system. Next is chapter 3 which I show the program flow and functions with flowcharts and use case diagrams. Gantt charts are also included to show the timeline of doing this project. In chapter 4 and 5, I talk about the design of the app, technologies used and development methodology used to develop the app. Lastly the last chapter, I talk about the challenges faced, contributions and future works.

CHAPTER 2: LITERATURE REVIEW

This chapter will be reviewing the works done by others such as articles and mobile apps that is related to my title. The features of the apps and the disadvantages and limits of them will be also discussed.

2.1 A scene recognition algorithm based on deep residual network

The work of (Mao Jiafa, et al., 2019) is to recognize a scene image based on deep residual network method. The model produced is then called "Scene-RecNet". The model produced is more efficient than other model because it can determine the image using less time and memory. The aim of this system is on extracting image features with strong representation with a good recognition rate even if the image is not clear, compressed or contain noise. During the image feature extraction process, the convolutional neural network is used to extract the parameters and features in the image and this process is repeat several times until the model learn most of the features.

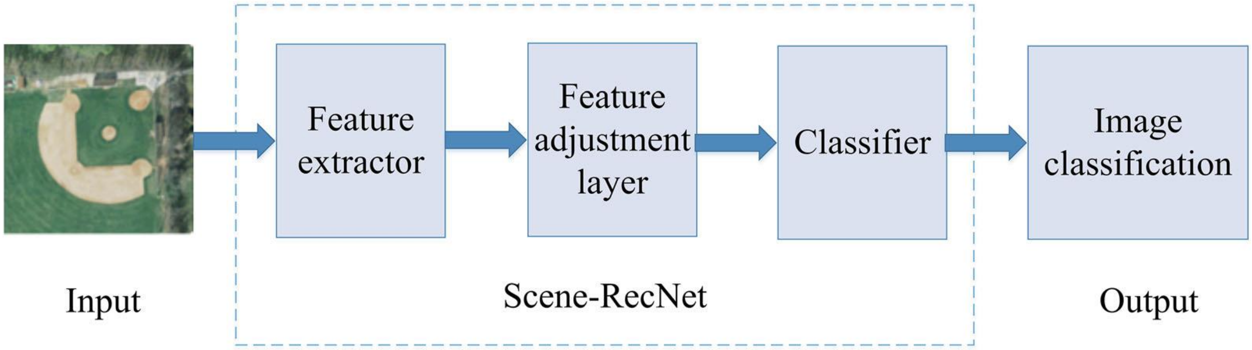


Figure 2.1.1: Steps of generating the model Scene-RecNet.

Two outputs are obtained from the model after the algorithm extract all the image features. One being the image features or vector and one is the labels of the image. The more the image is used to train the model, the more accurate the model is. The features extracted from the last image is more dominant than features from the other images in determining images. The image to be recognized is ran through the model and compared with the image features before adding a label to the image. Label with the highest probability will be the final label. The problem they wanted to solve with this project is to detect image accurately using less time and space as the image does not need to be high resolution in order to detect accurately.

Weakness and limitation of this system

Although ResNet is very powerful, a lot of time is required to train the model which is not suitable for normal use. This is because it is using a convolution neural network method where most of them have huge memory and computation requirements, especially while training. So, problem arises when the model is needed to deploy on mobile where the computation ability is not that great. Other than that, there are also some problem such as easy of training and the ability to generalize well.

2.2 Recognition of Tourist Attractions

The work of (Yuanfang Li, et al., n.d.) is to train a model that can recognize the famous attractions in Beijing. They used 10000 images and 10 places in Beijing, with 1000 images per place. First, they used Places-Convolution Neural Network (CNN) that is pre-trained using 2.5 million images and feed the 10000 images through the model. Then, the image features are obtained and the image features are different for night and day pictures. So, they trained separate classifiers for day and night. Then, they proceed to using pixel ratio by computing the total number of pixels of each value and then dividing by the total number of pixels. They used softmax, which is an equation that outputs a value between 0 and 1 that can be interpreted as the probability of a sample z from class j.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Figure 2.2.1: Softmax equation

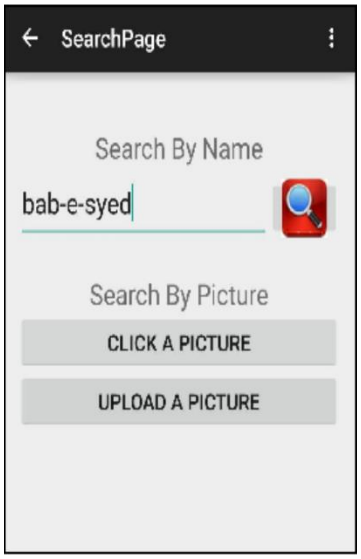
Other than softmax, Support Vector Machine (SVM) to classify the image to whether it is day or night. They reason they use SVM is because in some night photos, certain areas of the images are well lit and there will be some overlap in ratio features between day and night images. So, the used SVM instead of the simpler linear separation.

Weakness and limitation of this system

This system can only recognize the tourist attraction place from the outside. If a user has gotten into the building or place and wanted to recognize it, the system would not be able to tell which tourist attraction place it is. Moreover, in dim light condition, the SVM will have a hard time telling whether it is day or night which leads to inaccuracy in result.

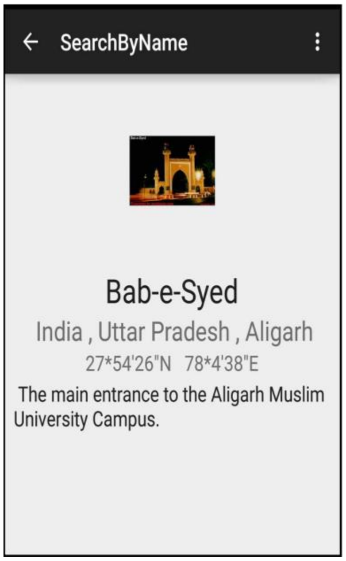
2.3 Mobile Travel Guide using Image Recognition and GPS/Geo Tagging

The work of (Ramsha Fatima, et al., n.d) is a mobile travel guide app that utilize image recognition. The image taken by the camera is compared with images in database using OpenCV along with ORB (Oriented FAST and rotated BRIEF) algorithm to detect image features and compare two images. The app has features for user to search the information about a tourist attraction using either name or picture. The main aim of this app is to help the user identify a place if they wanted to know the name of the place but could not find the sign board.



bab-e-syed

Figure 2.3.1: Tourist attraction search page using name



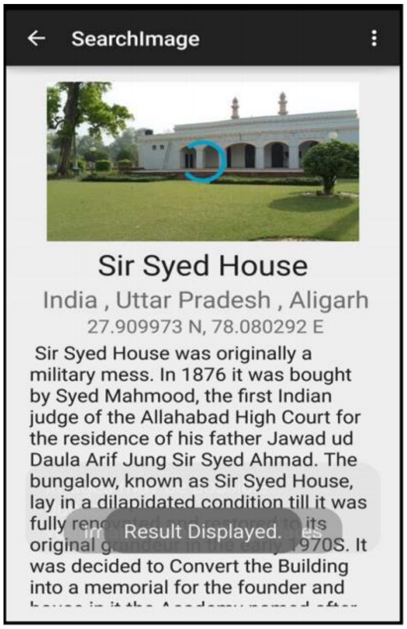
Bab-e-Syed

India , Uttar Pradesh , Aligarh

27*54'26"N 78*4'38"E

The main entrance to the Aligarh Muslim University Campus.

Figure 2.3.2: Search by name result



Sir Syed House

India , Uttar Pradesh , Aligarh

27.909973 N, 78.080292 E

Sir Syed House was originally a military mess. In 1876 it was bought by Syed Mahmood, the first Indian judge of the Allahabad High Court for the residence of his father Jawad ud Daula Arif Jung Sir Syed Ahmad. The bungalow, known as Sir Syed House, lay in a dilapidated condition till it was fully renovated and restored to its original grandeur in the early 1970S. It was decided to Convert the Building into a memorial for the founder and

Result Displayed.

Figure 2.3.3 Search by image result page

Other than that, the app will also allow user to search places using Google Map API (application programming interface) and also search for nearby places.

Weakness and limitation of this system

This app did not show the locations of the tourist attraction places in their database on the Google map feature they have in the app. Instead, they just put the Google map API in without any feature except searching for places which regular Google map app can do better. So, in my project, I will be attempting to make use of the Google Map API to show the user nearby tourist attraction as well as navigating them to it and showing the crowd of the place.

2.4 Tourist Attractions Classification using ResNet

The work of (Nanda Maulina Firdaus, et al., n.d.) tried to identify tourist attractions in Jakarta and Depok using ResNet. Raw datasets are taken from Google with different angle, lighting and size. Then, the datasets are preprocessed by converting the color model into RGB and resizing them to the same size for easier classification process. K-fold cross validation method is used to train the model. The way that k-fold cross validation work is to divide the set of samples N into k parts and then using one of the parts to do testing and the rest become training data. In this system, k=5 is used.

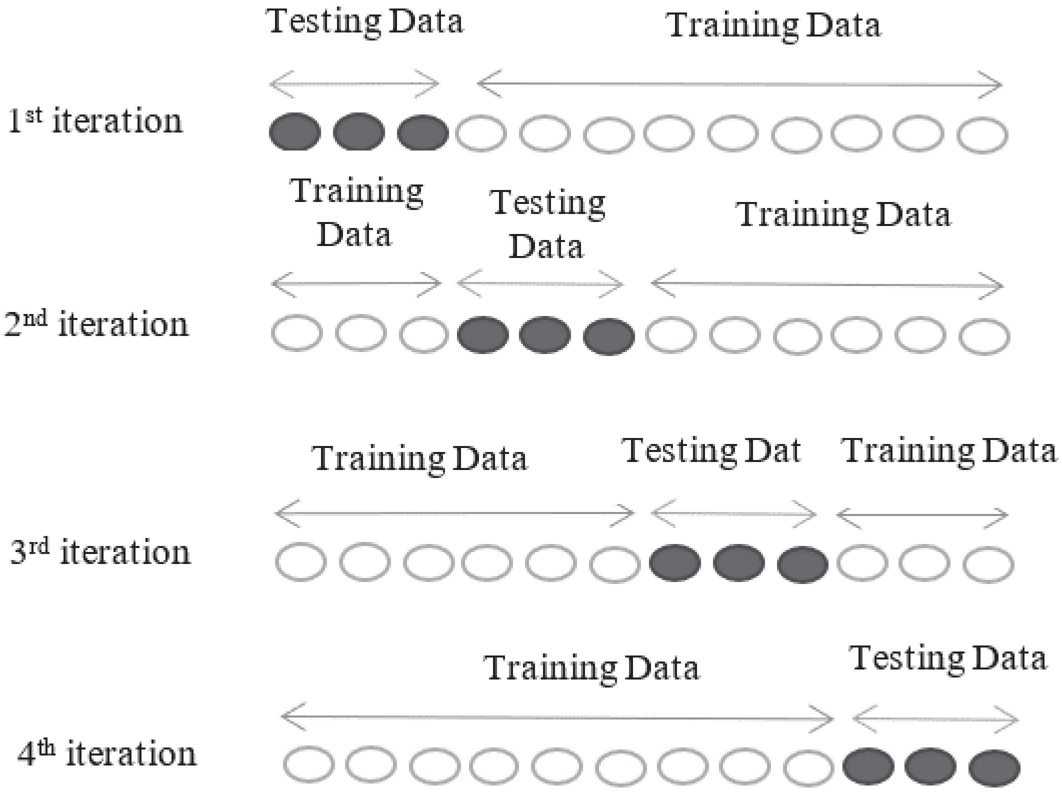


Figure 2.4.1: K-fold Cross Validation

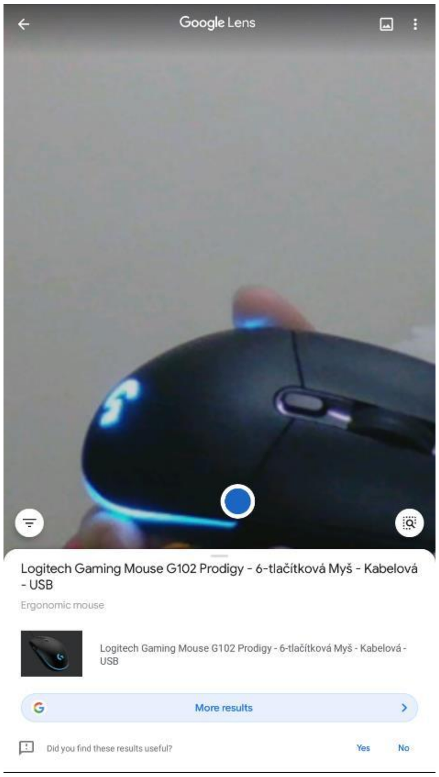
Next, they started to use ResNet to solve the image classification. ResNet50 is used which is ResNet with 50 layers because it is faster if there is less layers. The data is finally evaluated and accuracy is calculated.

Weakness and limitation of this system

Although the accuracy is very high using ResNet, it can be further improved by increasing the number of ResNet layers and value of k in k-fold cross validation.

2.5 Google Lens

Google Lens is an app developed by Google using their image recognition technology. It can do many things that is related to image such as translate text, identifying plant and animal, identify objects and building, scan (Quick Response) QR code and barcodes and many more. The interface is very user friendly as everything is displayed on the homepage and labelled and there or not much button to click.



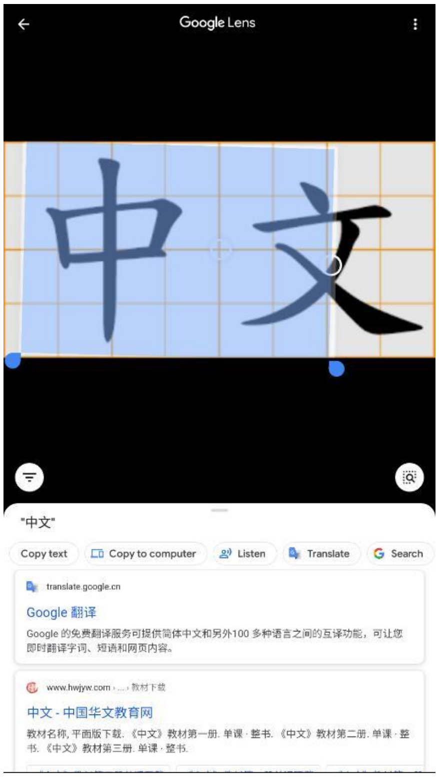
Logitech Gaming Mouse G102 Prodigy - 6-tlačítková Myš - Kabelová - USB

Ergonomic mouse

Logitech Gaming Mouse G102 Prodigy - 6-tlačítková Myš - Kabelová - USB

Did you find these results useful?

Figure 2.5.1: Detecting Object



"中文"

translate.google.cn

Google 翻译

Google 的免费翻译服务可提供简体中文和另外100 多种语言之间的互译功能，可让您即时翻译字词、短语和网页内容。

www.hwjyw.com › ... › 教材下载

中文 - 中国华文教育网

教材名称, 平面版下载. 《中文》教材第一册. 单课 · 整书. 《中文》教材第二册. 单课 · 整书. 《中文》教材第三册. 单课 · 整书.

Figure 2.5.2: Detecting text and translate

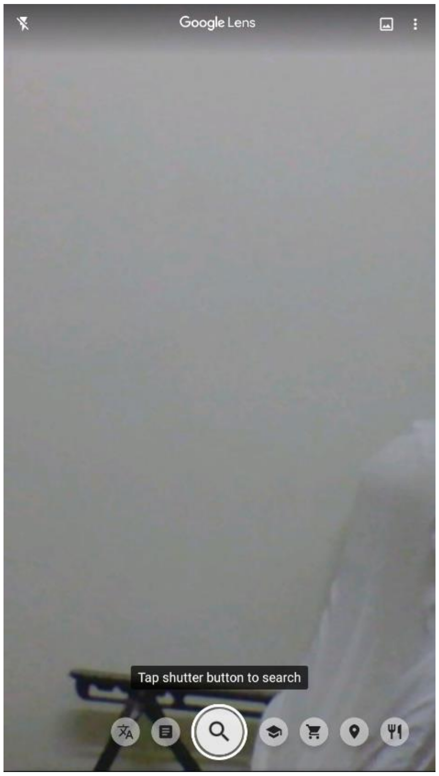


Figure 2.5.3: Home page

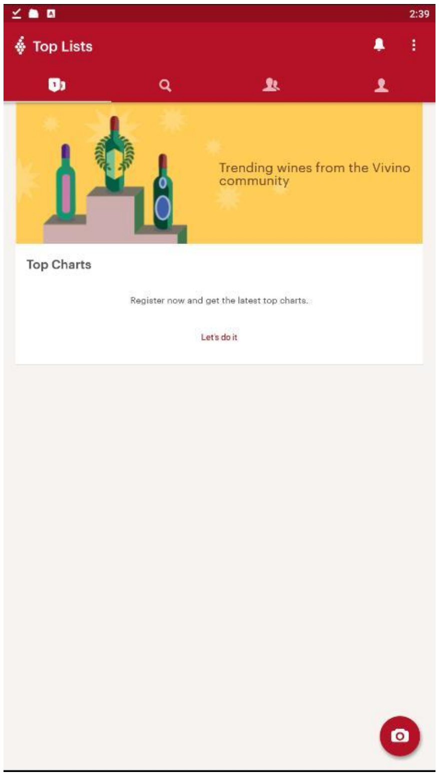
As we can see in the homepage, the camera is already enabled in the home page so that user can open the app and identify thing quickly. User can not only detect images from camera, there is a gallery option on the top right corner which allows user to select from their library. The list of type of image recognition is displayed at the bottom section of the app.

<u>Weakness and limitation of this app</u>

Google Lens is not supported on all phone models. Older phone models that does not have suitable requirement would not able to use it. Other than that, uploading image to the Google Lens platform, it could violate our privacy.

2.6 Vivino

Vivino is developed by a company called “Vivino” and it is used to identify wines using image recognition. After scanning, important information such as its facts, the taste characteristic such as light, dry and soft and even its ingredient. Other than that, there is also a community talking about the wine so that people that love wine can discuss in it. After checking the facts, the app will also show the price of the wine and direct user to the wine official webpage for user to buy.



Trending wines from the Vivino community

Top Charts

Register now and get the latest top charts.

Figure 2.6.1: Home page

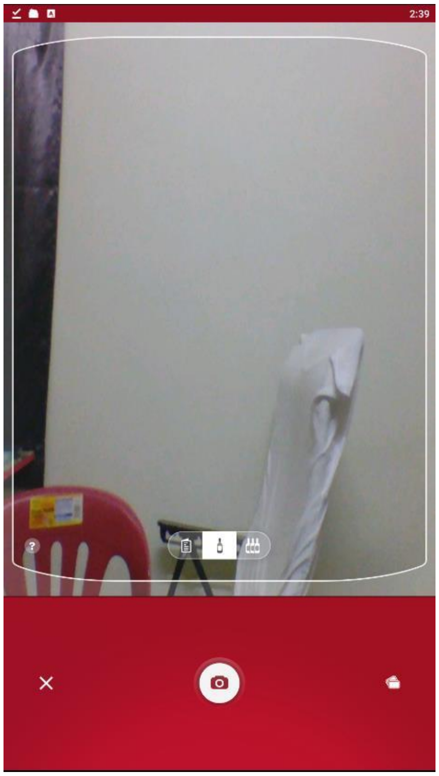
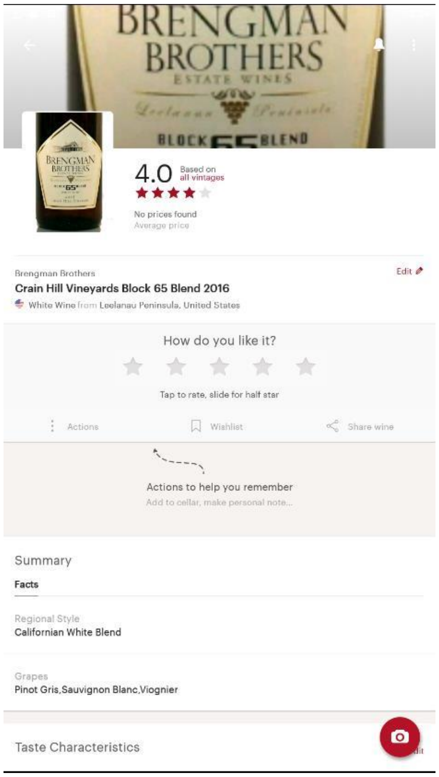


Figure 2.6.2: Scanning feature



4.0 Based on all vintages

★★★★☆

No prices found
Average price

Brengman Brothers
Crain Hill Vineyards Block 65 Blend 2016
White Wine from Leelanau Peninsula, United States

How do you like it?

☆ ☆ ☆ ☆ ☆

Tap to rate, slide for half star

Actions to help you remember
Add to cellar, make personal note...

Summary

Facts

Regional Style
Californian White Blend

Grapes
Pinot Gris,Sauvignon Blanc,Viognier

Taste Characteristics

Figure 2.6.3: Wine info

Weakness and limitation of this app

The image recognition feature of this app is very poor. The image has to have a clear text of the wine label in order for the app to identify the type of wine. If it is taken faraway and the text isn't clear, it would have error. This makes its more like text recognition instead of image recognition.

2.7 Transfer Learning of a Deep Learning Model for Exploring Tourists' Urban Image Using Geotagged Photos

The work of (Youngok Kang, et al., n.d) is to deep learning photo of tourist attractions in Seoul. Hundred thousand of photos were extracted from Flickr, containing 75 scenes and 13 categories. The dataset is categorized into their respective categories such as shopping, people and animal and split into test and train set. After that, the trainset is continuously fed into the Inception V3 model to fine tune to model. For example, the normal training dataset is first put through the model then at the 2nd time, data augmentation is performed to alter the dataset a bit so it does not overfit. Then, the final model is selected to be applied to the test set. Their dataset is on a 60% train, 20% validate and 20% test ratio and used AWS web services as the platform for training the model. Overall, the accuracy is quite high at 85.77%.

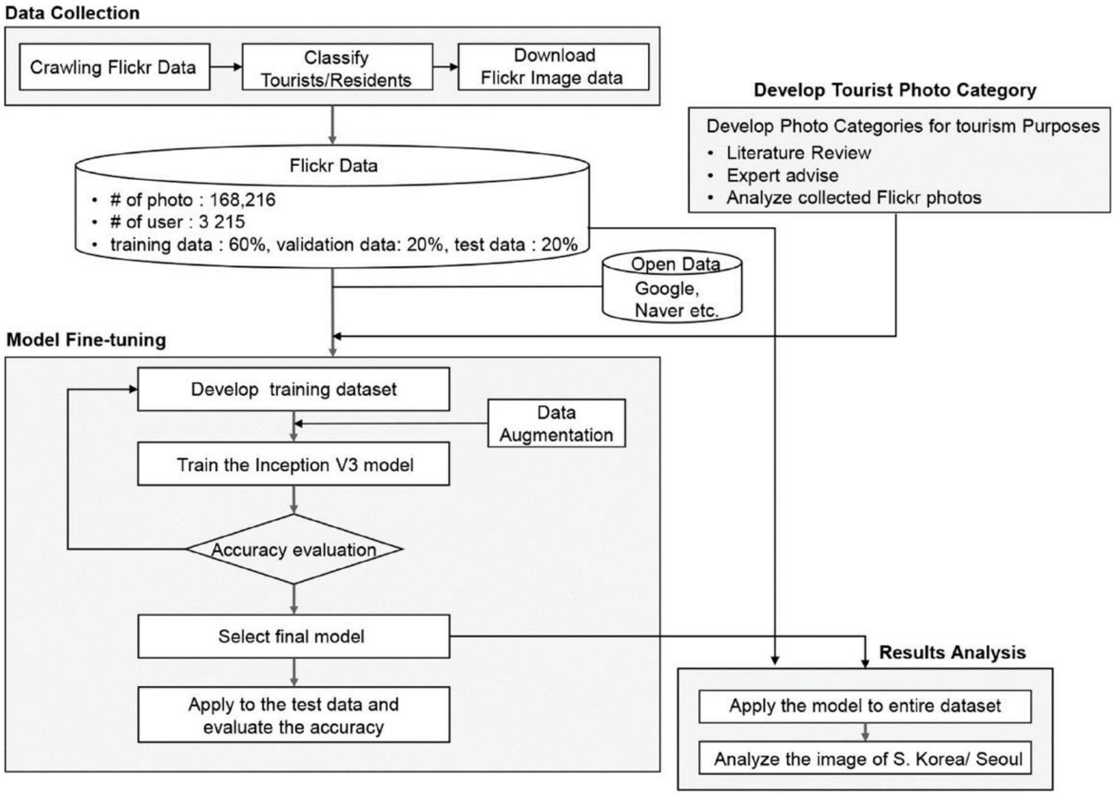

Data Collection
Crawling Flickr Data
Classify Tourists/Residents
Download Flickr Image data
Develop Tourist Photo Category
Develop Photo Categories for tourism Purposes
• Literature Review
• Expert advise
• Analyze collected Flickr photos
Flickr Data
• # of photo : 168,216
• # of user : 3 215
• training data : 60%, validation data: 20%, test data : 20%
Open Data
Google, Naver etc.
Model Fine-tuning
Develop training dataset
Data Augmentation
Train the Inception V3 model
Accuracy evaluation
Select final model
Apply to the test data and evaluate the accuracy
Results Analysis
Apply the model to entire dataset
Analyze the image of S. Korea/ Seoul

Figure 2.7.1 Process of training the model

Weakness and limitation of this system

The training process is very time consuming and resource intensive. In the training set, there are over 100,000 photos to train not including the image that had been augmented and for normal computers, it will take a long time just to go through one training process.

CHAPTER 3: System Design

Overall architecture diagram

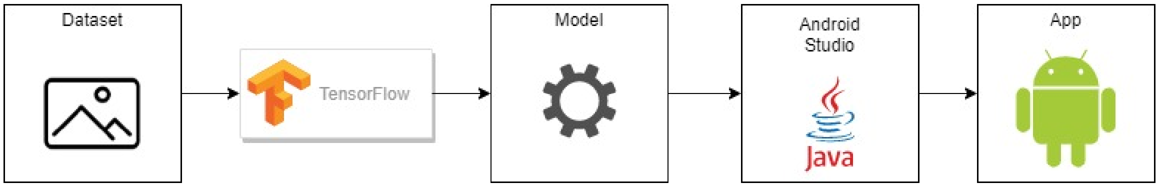


Figure 3.1 Overall architecture diagram

First, the dataset is gathered using phone. There are a total of 17 classes. After specifying the image path, TensorFlow will detect the classes separated into folders and split the dataset into train and test. Then, the pictures are fed into a neural network predefined by TensorFlow and train a model. The model output will be in the .tflite format which can be imported into Android Studio and integrate into the Java code as there are predefined functions in the model. Lastly, the model will be called and return the result to user.

Flowchart

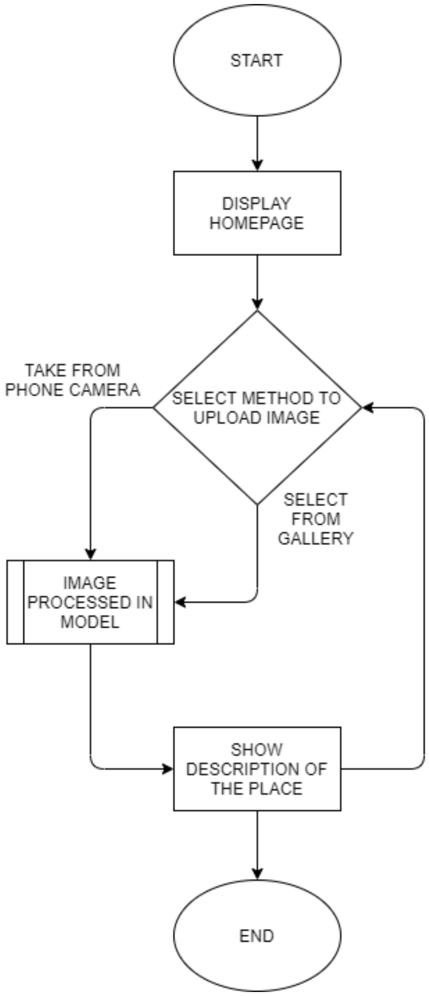


Figure 3.2: Flowchart

The above figure is the flowchart for the app of this project. When the user starts the app, a homepage will greet the user with the add picture button. When the add picture button is pressed, user can choose either to take photo from their camera or select picture from the gallery and upload it to the app. After selecting or taking a photo, the photo will be processed through a model to identify the building or structure in the photo taken. The app will also display the details such as the history and story of the place. User can then click the add photo button again to select or take a new picture for the app to identify.

Use case diagram

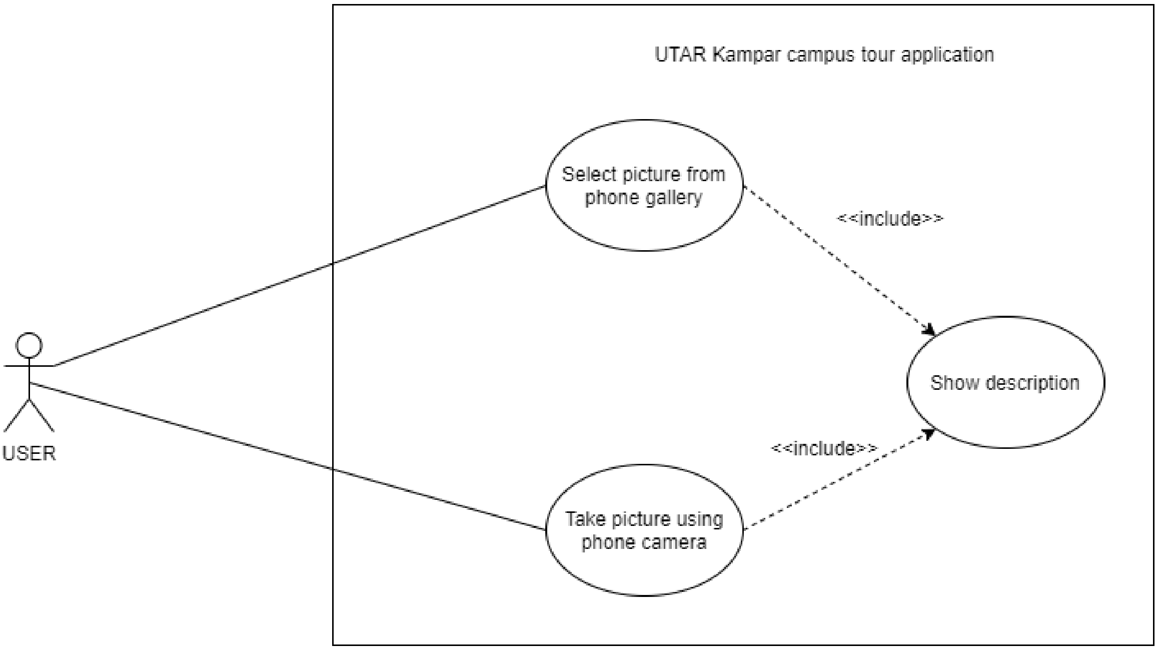

UTAR Kampar campus tour application
Select picture from phone gallery
<<include>>
Show description
<<include>>
Take picture using phone camera
USER

Figure 3.3: Use-case diagram

The use case diagram illustrating the function that the user can perform

Flow for developing the model

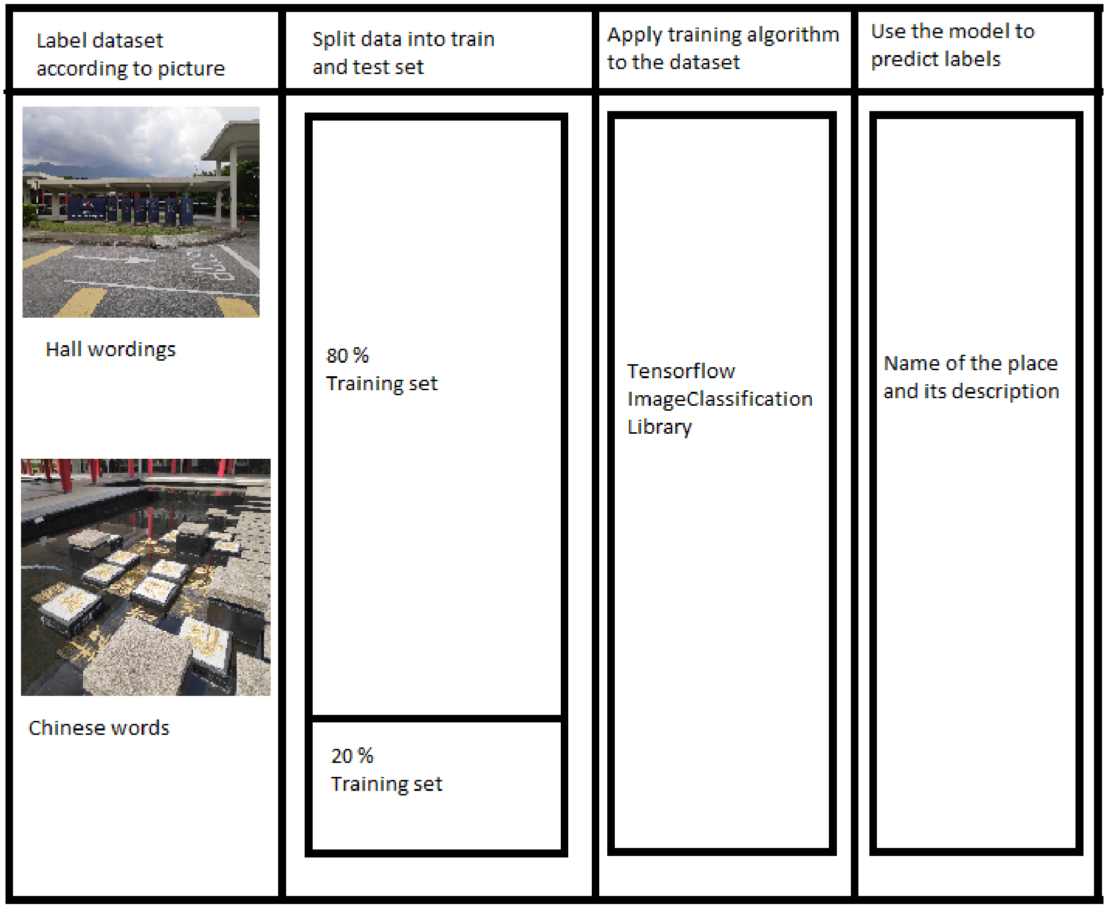
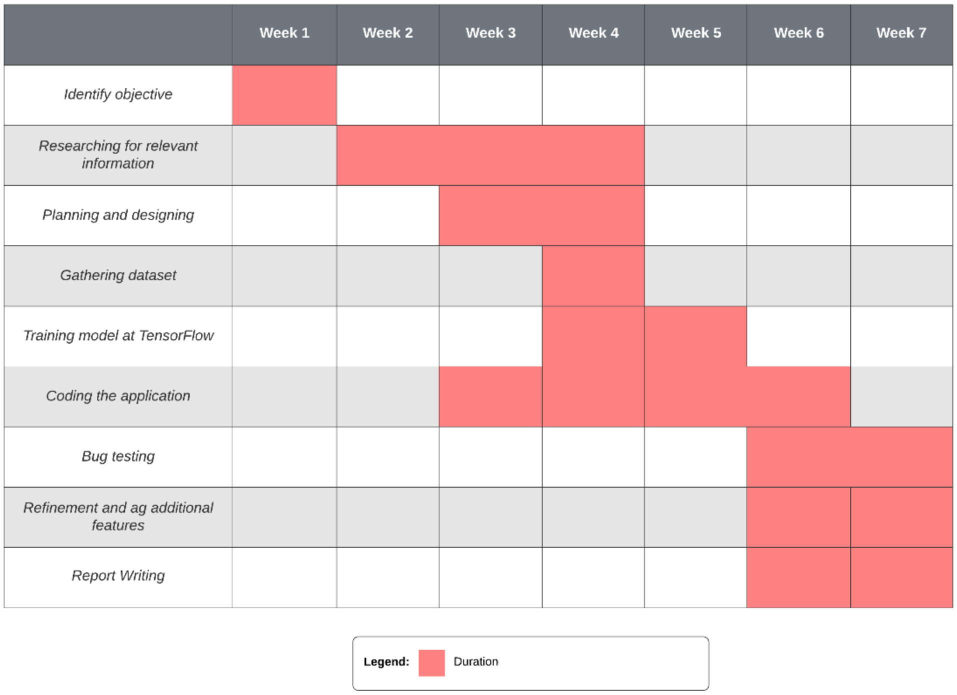

Label dataset according to picture
Split data into train and test set
Apply training algorithm to the dataset
Use the model to predict labels
Hall wordings
Chinese words
80 % Training set
20 % Training set
Tensorflow ImageClassification Library
Name of the place and its description

Figure 3.4: Flow of model training

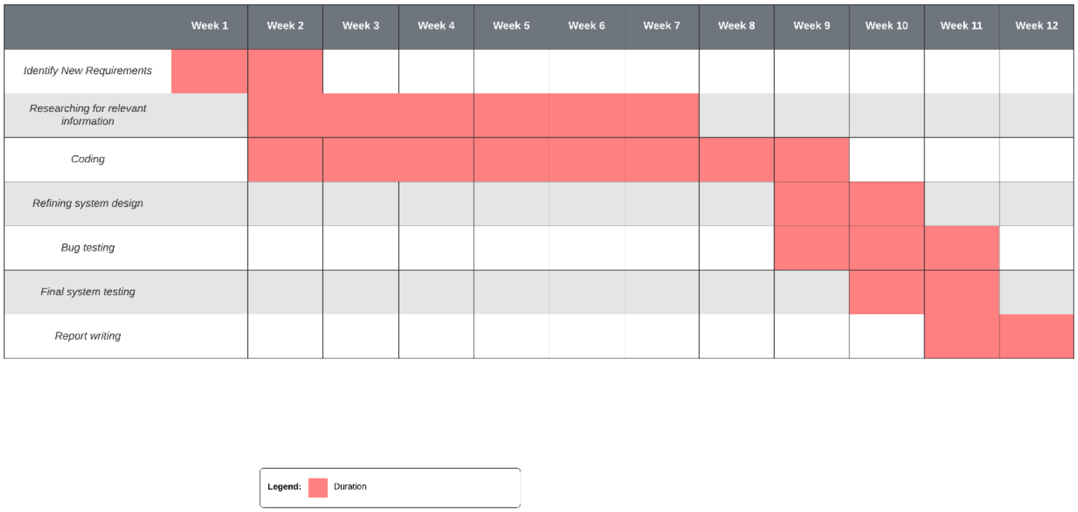
Project development timeline



	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Identify objective	🟥						
Researching for relevant information		🟥	🟥	🟥			
Planning and designing			🟥	🟥			
Gathering dataset				🟥			
Training model at TensorFlow				🟥	🟥		
Coding the application			🟥	🟥	🟥	🟥	
Bug testing						🟥	🟥
Refinement and ag additional features						🟥	🟥
Report Writing						🟥	🟥

Legend: 🟥 Duration

Figure 3.5: Gantt chart for Project 1



	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12
Identify New Requirements	🟥	🟥										
Researching for relevant information		🟥	🟥	🟥	🟥	🟥	🟥					
Coding		🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥			
Refining system design									🟥	🟥		
Bug testing									🟥	🟥	🟥	
Final system testing										🟥	🟥	
Report writing											🟥	🟥

Legend: 🟥 Duration

Figure 3.6: Gantt chart for Project 2

The first figure is the Gantt chart for Project 1 during the October 2020 trimester and second is for the January 2021 trimester.

Activity diagram

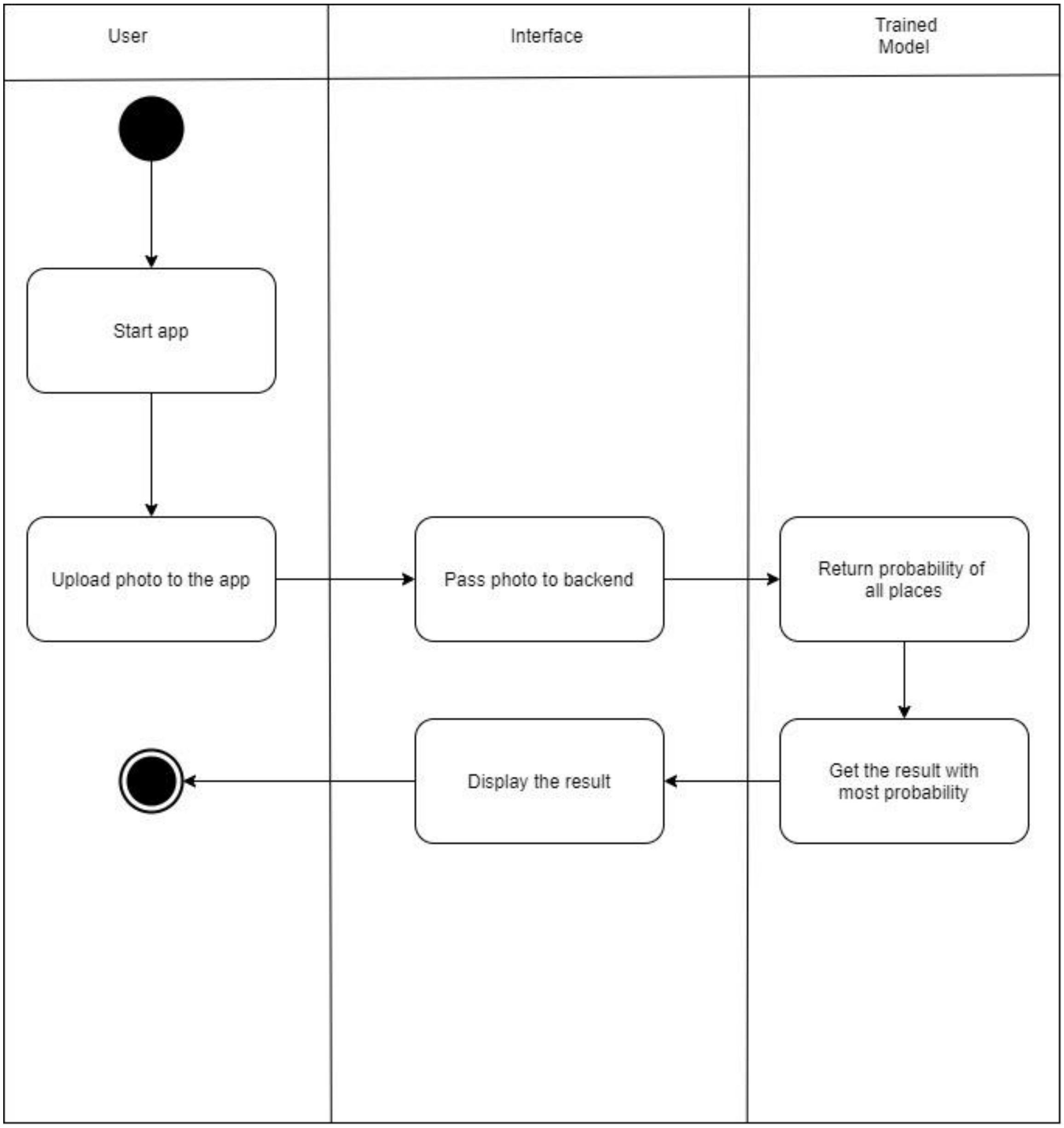


Figure 3.7 Activity Diagram

In the use case diagram above, user first start the app and they can upload the photo to the app with either camera or gallery. Then, the app will analyze the photo return a list of probability for each photo. Then, an algorithm will be run to get the result with most probability and return it to user.

CHAPTER 4: Methodology and tools

4.1 Methodology

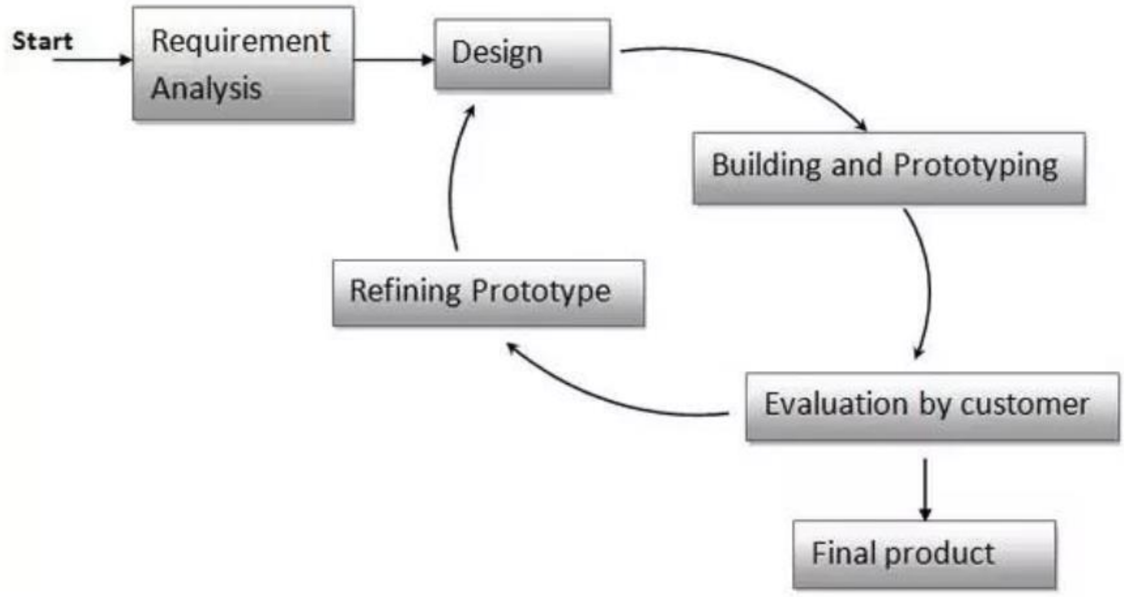


Figure 4.1.1 Prototyping

The chosen software development method will be prototyping. The reason why prototyping is chosen is because when we are developing this software, we can develop a prototype first, then see what is wrong and lacking in the software and just add in easily. Compared to other method, adding in new features during development will be harder. Other than that, this is a non-critical application that does not need careful planning. Using this method, user can try out the features quickly and discover new requirement so that early requirement planning become not that important as it will change over time as well. The stages of prototyping are requirement analysis, design, building and prototyping, evaluation by customer, refining prototype, delivering the final product and ultimately maintaining it. There are four types of prototyping, namely: rapid throwaway prototypes, evolutionary prototype, incremental prototype and extreme prototype. In this project, incremental prototype will be used.

1. Requirement Analysis

   The main features and requirement are determined at this stage.
2. Design

User interface and database design are produced.

3. Building and prototyping

   A version of the product is developed.

4. Evaluation by customer

   The app is shown to customer for comment and evaluation.

5. Refining prototype

   The comment and additional requirement requested by customer is jotted down and will be implemented in the next version.

6. Final product

   The customer is finally satisfied with the product, the product is delivered and will proceed to product maintenance phase.

4.2 Technologies involved

Platform:

1. Android

   Android is the most used mobile operating system right now with 74.6% of mobile phone user being android user as of May 2020. This allows more user to have access to the app to increase the app popularity.

2. Windows

   Windows is an operating system for laptop and desktop pcs. Between 77% and 87.8% of devices in the world are running on Windows.

Programming language:

1. Java

   Java is an object-oriented programming language and it is one of the most popular language among programmers right now which has a 16.4% share among other programming language and it is the top 2 programming language, just behind Python. Object oriented language such as Java has many advantages such as it is reusable through inheritance. For example, class and subclasses can be created and the subclasses will inherit all attributes and method that its class has, eliminating the duplication of attributes and methods.

2. Python

   This project require some machine learning and Python is the best language to do it, so Python will be used in this project.

Software:

1. Android Studio

   Android Studio a software to develop Android app. It supports various languages such as Dart, Java and Kotlin.

2. Visual Paradigm

It is a software used to make diagrams such as Entity Relationship Diagram (ERD) to design databases, sequence diagram to design the flow of process and many more.

3. Jupyter notebook

   A software that allows users to write python codes.

Libraries:

1. TensorFlow

   TensorFlow is a library in python for user to perform machine learning. The layers of network are predefined so that it is easier to use.

Hardware:

1. Laptop

   The main device used to complete this project.

   Model: Asus TUF FX705

   Processor: Intel Core i7 – 8750H

   Graphic card: NVIDIA GTX 1060

   RAM: 16GB

   OS: Windows 10

2. Smartphone

   The main device used to run the mobile app prototype.

   Model: Xiao Mi Pocophone F1

   Processor: Snapdragon 845

   RAM: 6GB

   OS: Android 10

4.3 User requirements

Functional requirements:

- User should be able to take photo through the app.
- User should be able to select photo from the gallery.

- User should be able to know the information of UTAR Kampar campus building and structures through the app.
- User should be able to list the places that the app can recognize.
- User should be able to use all the buttons in the app.

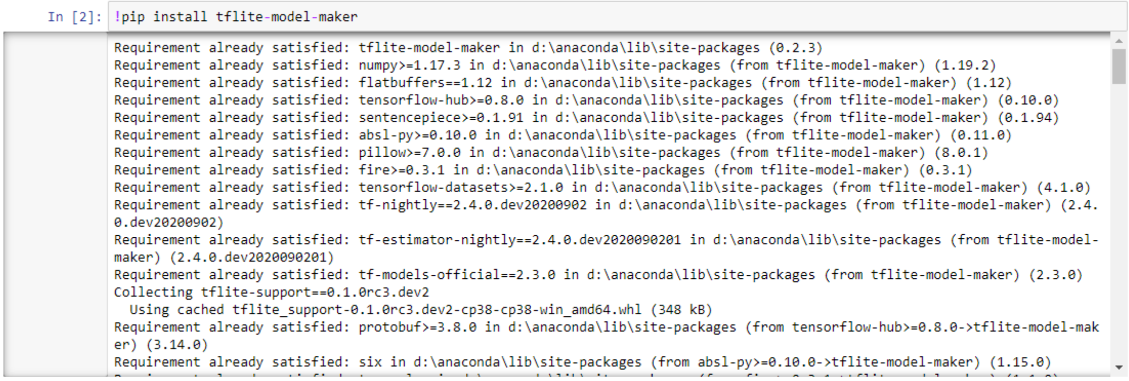
Non-functional requirement:

- User should be able to run the app if they are using an Android device.
- User should be able to use the app without major bug.
- User should expect a maximum load time of 5 seconds.
- User should be able to maintain their privacy when using the app and ensuring their photos are not leaked.

Chapter 5: Specification and test results

5.1 TensorFlow

TensorFlow is chosen to be my method of training my model mainly because of its simplicity and ease of use. Furthermore, the trained model is quite good. Below are the steps on the process of using TensorFlow to train a model.

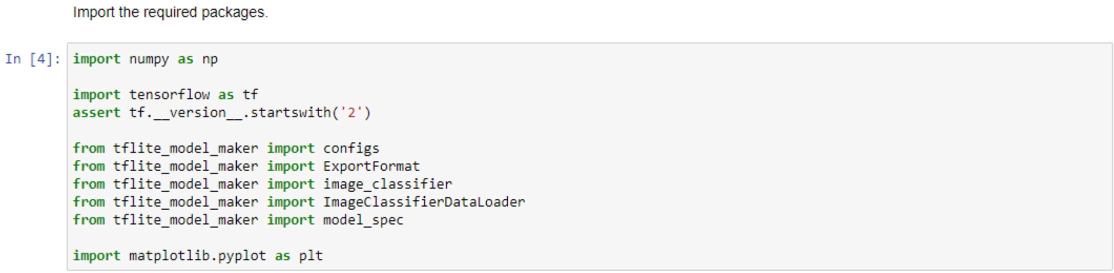


```
In [2]: !pip install tflite-model-maker
```

```
Requirement already satisfied: tflite-model-maker in d:\anaconda\lib\site-packages (0.2.3)
Requirement already satisfied: numpy>=1.17.3 in d:\anaconda\lib\site-packages (from tflite-model-maker) (1.19.2)
Requirement already satisfied: flatbuffers==1.12 in d:\anaconda\lib\site-packages (from tflite-model-maker) (1.12)
Requirement already satisfied: tensorflow-hub>=0.8.0 in d:\anaconda\lib\site-packages (from tflite-model-maker) (0.10.0)
Requirement already satisfied: sentencepiece>=0.1.91 in d:\anaconda\lib\site-packages (from tflite-model-maker) (0.1.94)
Requirement already satisfied: absl-py>=0.10.0 in d:\anaconda\lib\site-packages (from tflite-model-maker) (0.11.0)
Requirement already satisfied: pillow>=7.0.0 in d:\anaconda\lib\site-packages (from tflite-model-maker) (8.0.1)
Requirement already satisfied: fire>=0.3.1 in d:\anaconda\lib\site-packages (from tflite-model-maker) (0.3.1)
Requirement already satisfied: tensorflow-datasets>=2.1.0 in d:\anaconda\lib\site-packages (from tflite-model-maker) (4.1.0)
Requirement already satisfied: tf-nightly==2.4.0.dev20200902 in d:\anaconda\lib\site-packages (from tflite-model-maker) (2.4.
0.dev20200902)
Requirement already satisfied: tf-estimator-nightly==2.4.0.dev2020090201 in d:\anaconda\lib\site-packages (from tflite-model-
maker) (2.4.0.dev2020090201)
Requirement already satisfied: tf-models-official==2.3.0 in d:\anaconda\lib\site-packages (from tflite-model-maker) (2.3.0)
Collecting tflite-support==0.1.0rc3.dev2
  Using cached tflite_support-0.1.0rc3.dev2-cp38-cp38-win_amd64.whl (348 kB)
Requirement already satisfied: protobuf>=3.8.0 in d:\anaconda\lib\site-packages (from tensorflow-hub>=0.8.0->tflite-model-mak
er) (3.14.0)
Requirement already satisfied: six in d:\anaconda\lib\site-packages (from absl-py>=0.10.0->tflite-model-maker) (1.15.0)
```

Figure 5.1.1: Installing TensorFlow

Installing the TensorFlow model maker library. The neural network of TensorFlow model maker library is predefined so that I do not have to define each layer which make the model training a lot simpler and faster.

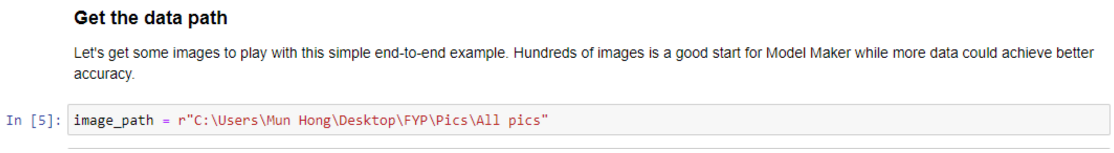


Import the required packages.

```
In [4]: import numpy as np

        import tensorflow as tf
        assert tf.__version__.startswith('2')

        from tflite_model_maker import configs
        from tflite_model_maker import ExportFormat
        from tflite_model_maker import image_classifier
        from tflite_model_maker import ImageClassifierDataLoader
        from tflite_model_maker import model_spec

        import matplotlib.pyplot as plt
```

Figure 5.1.2: Importing libraries

Import required library such as numpy and tensorflow. Numpy is a very common library for python to do most basic functionalities.



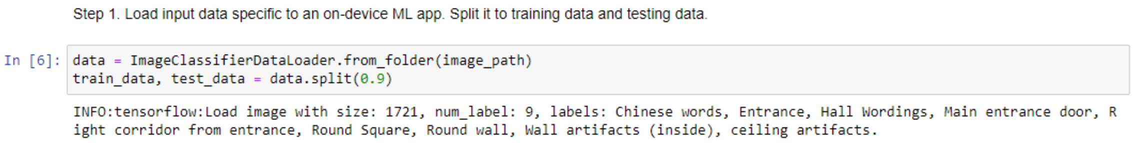
Get the data path

Let's get some images to play with this simple end-to-end example. Hundreds of images is a good start for Model Maker while more data could achieve better accuracy.

```
In [5]: image_path = r"C:\Users\Mun Hong\Desktop\FYP\Pics\All pics"
```

Figure 5.1.3: Specifying path

Specify the path that contain the image dataset. The image is gathered using my phone and they are taken with different angle to make the model less overfit. There are 17 classes in total and around 80-100 images for each class to be split into train and test set.

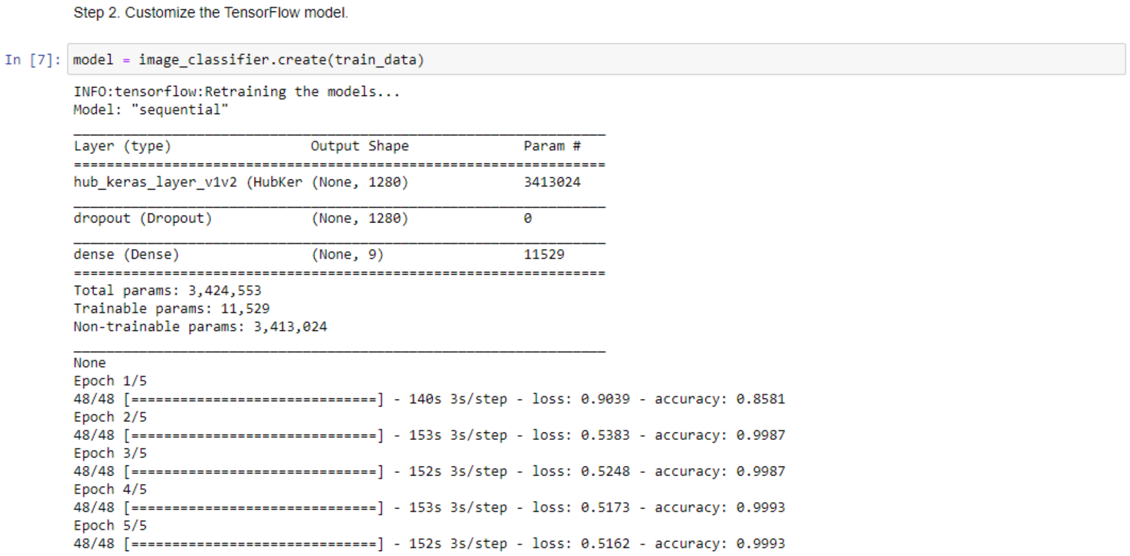


Step 1. Load input data specific to an on-device ML app. Split it to training data and testing data.

```
In [6]: data = ImageClassifierDataLoader.from_folder(image_path)
        train_data, test_data = data.split(0.9)

        INFO:tensorflow:Load image with size: 1721, num_label: 9, labels: Chinese words, Entrance, Hall Wordings, Main entrance door, R
        ight corridor from entrance, Round Square, Round wall, Wall artifacts (inside), ceiling artifacts.
```

Figure 5.1.4: Split dataset

Split the dataset into train and test set from the same image pool. The ratio is 9 train to 1 test. TensorFlow automatically detect the classes and train them accordingly.

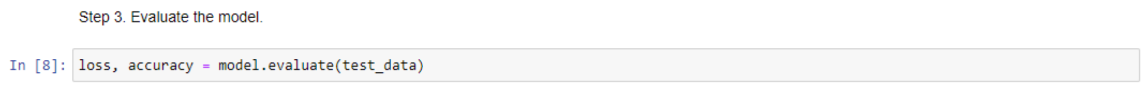


Step 2. Customize the TensorFlow model.

```
In [7]: model = image_classifier.create(train_data)

        INFO:tensorflow:Retraining the models...
        Model: "sequential"
        _________________________________________________________________
        Layer (type)                 Output Shape              Param #
        =================================================================
        hub_keras_layer_v1v2 (HubKer (None, 1280)              3413024
        _________________________________________________________________
        dropout (Dropout)            (None, 1280)              0
        _________________________________________________________________
        dense (Dense)                (None, 9)                 11529
        =================================================================
        Total params: 3,424,553
        Trainable params: 11,529
        Non-trainable params: 3,413,024
        _________________________________________________________________
        None
        Epoch 1/5
        48/48 [==============================] - 140s 3s/step - loss: 0.9039 - accuracy: 0.8581
        Epoch 2/5
        48/48 [==============================] - 153s 3s/step - loss: 0.5383 - accuracy: 0.9987
        Epoch 3/5
        48/48 [==============================] - 152s 3s/step - loss: 0.5248 - accuracy: 0.9987
        Epoch 4/5
        48/48 [==============================] - 153s 3s/step - loss: 0.5173 - accuracy: 0.9993
        Epoch 5/5
        48/48 [==============================] - 152s 3s/step - loss: 0.5162 - accuracy: 0.9993
```

Figure 5.1.5: Model training

Training the model. The model is running for 5 epochs. The number can be increased for higher accuracy but to avoid overfit, a lower number of epochs is recommended.

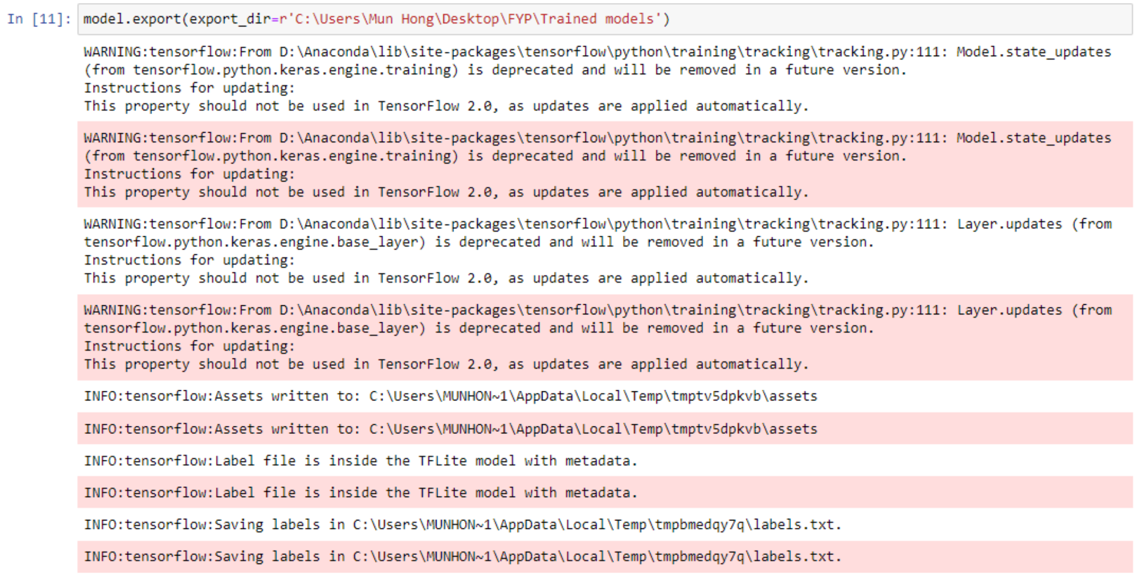


Step 3. Evaluate the model.

```
In [8]: loss, accuracy = model.evaluate(test_data)
```

Figure 5.1.6: Model testing

Testing the data using the trained model. The test data that is split earlier is evaluated through the model trained.



```
In [11]: model.export(export_dir=r'C:\Users\Mun Hong\Desktop\FYP\Trained models')

WARNING:tensorflow:From D:\Anaconda\lib\site-packages\tensorflow\python\training\tracking\tracking.py:111: Model.state_updates
(from tensorflow.python.keras.engine.training) is deprecated and will be removed in a future version.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.

WARNING:tensorflow:From D:\Anaconda\lib\site-packages\tensorflow\python\training\tracking\tracking.py:111: Model.state_updates
(from tensorflow.python.keras.engine.training) is deprecated and will be removed in a future version.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.

WARNING:tensorflow:From D:\Anaconda\lib\site-packages\tensorflow\python\training\tracking\tracking.py:111: Layer.updates (from
tensorflow.python.keras.engine.base_layer) is deprecated and will be removed in a future version.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.

WARNING:tensorflow:From D:\Anaconda\lib\site-packages\tensorflow\python\training\tracking\tracking.py:111: Layer.updates (from
tensorflow.python.keras.engine.base_layer) is deprecated and will be removed in a future version.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.

INFO:tensorflow:Assets written to: C:\Users\MUNHON~1\AppData\Local\Temp\tmptv5dpkvb\assets

INFO:tensorflow:Assets written to: C:\Users\MUNHON~1\AppData\Local\Temp\tmptv5dpkvb\assets

INFO:tensorflow:Label file is inside the TFLite model with metadata.

INFO:tensorflow:Label file is inside the TFLite model with metadata.

INFO:tensorflow:Saving labels in C:\Users\MUNHON~1\AppData\Local\Temp\tmpbmedqy7q\labels.txt.

INFO:tensorflow:Saving labels in C:\Users\MUNHON~1\AppData\Local\Temp\tmpbmedqy7q\labels.txt.
```

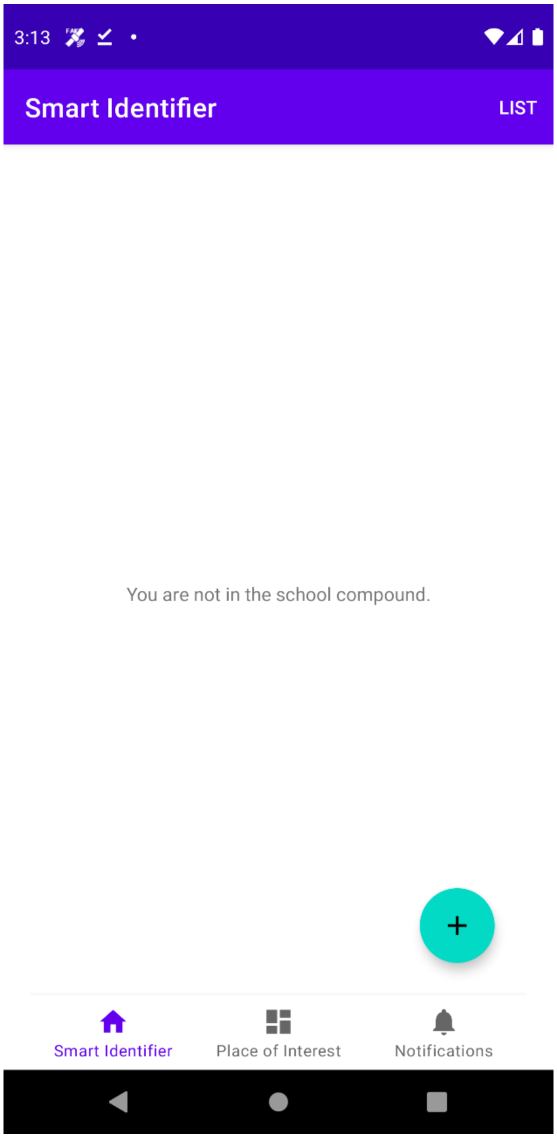
Figure 5.1.7: Exporting model

Exporting the model trained to be used in Android Studio. The output file will be of a .tflite file format and can be used in Android Studio to develop the app later.

5.2 Java

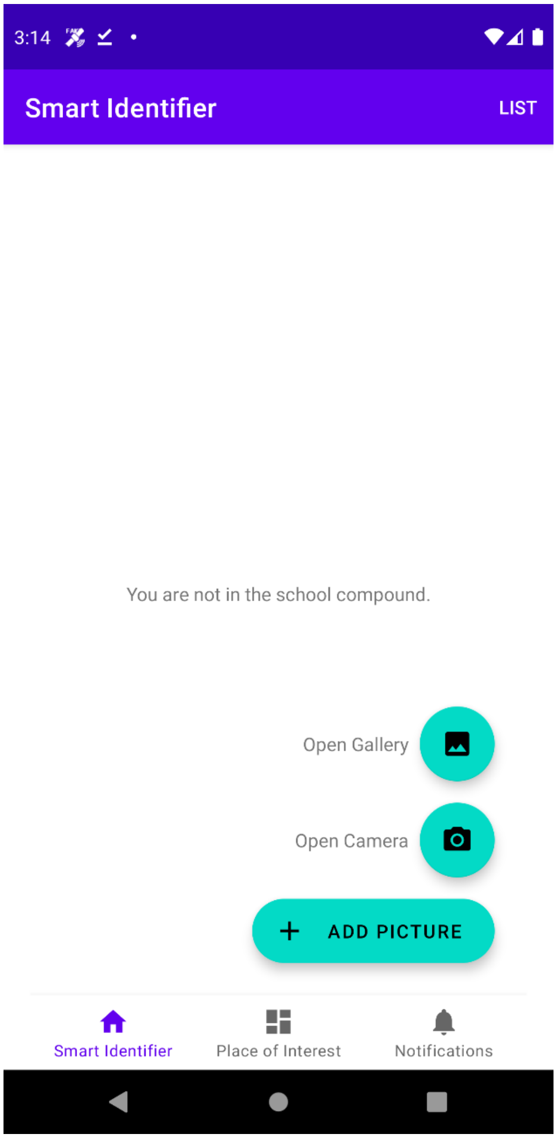
Java is the selected programming language as it is my most familiar language compared to others so I have used it for the development of my app.

5.3 Design



You are not in the school compound.

Figure 5.3.1 Homepage



You are not in the school compound.

Figure 5.3.2 Choose photo method

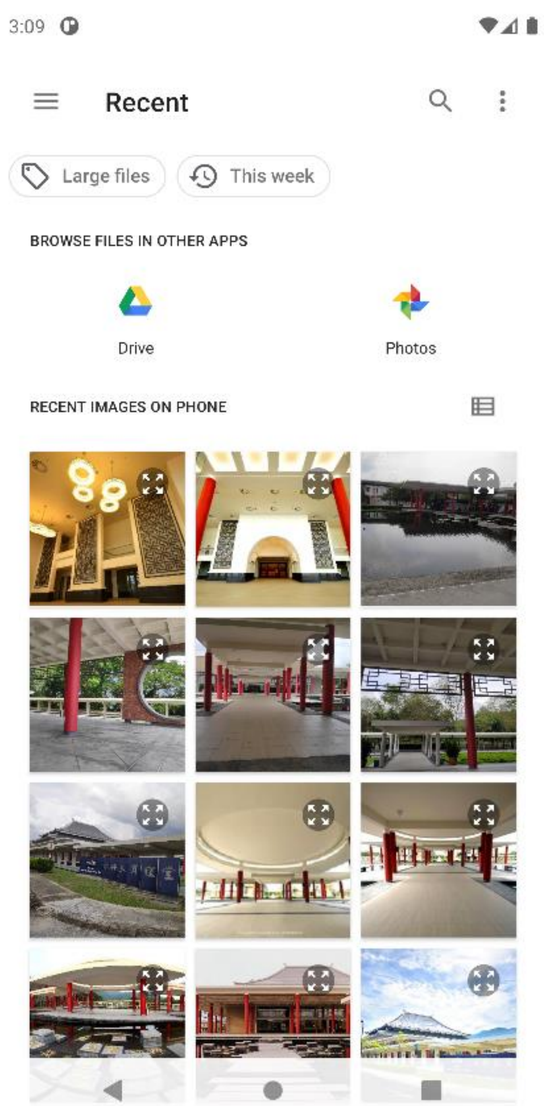


Figure 5.3.3: Select photo from gallery

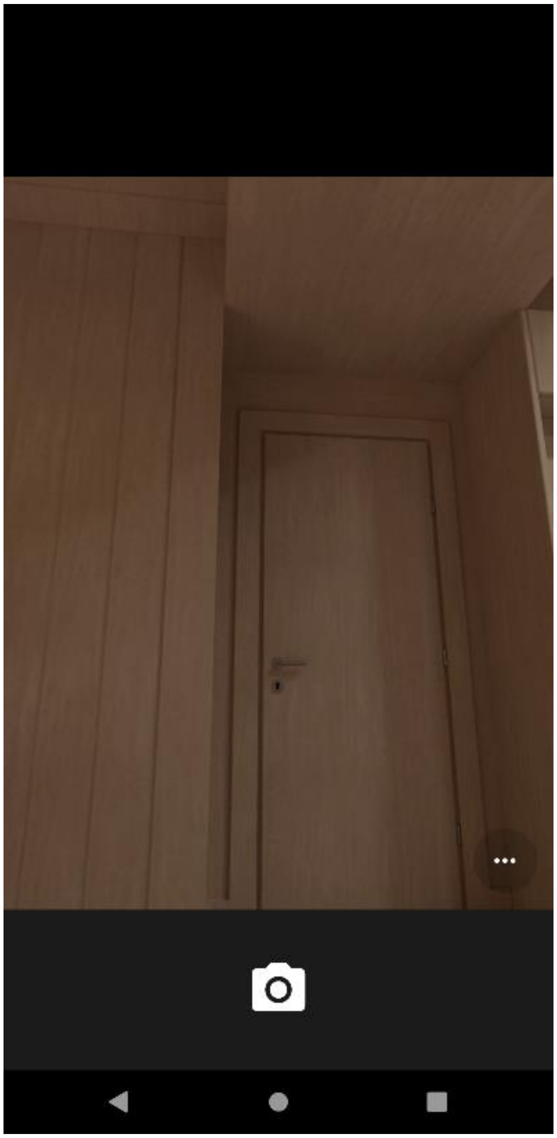


Figure 5.3.4: Take photo from camera

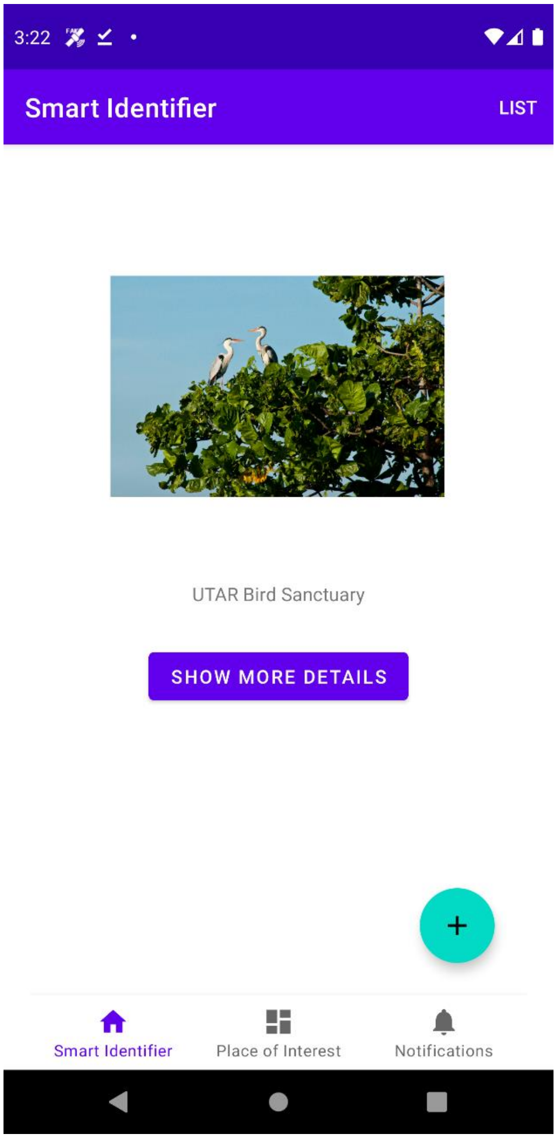
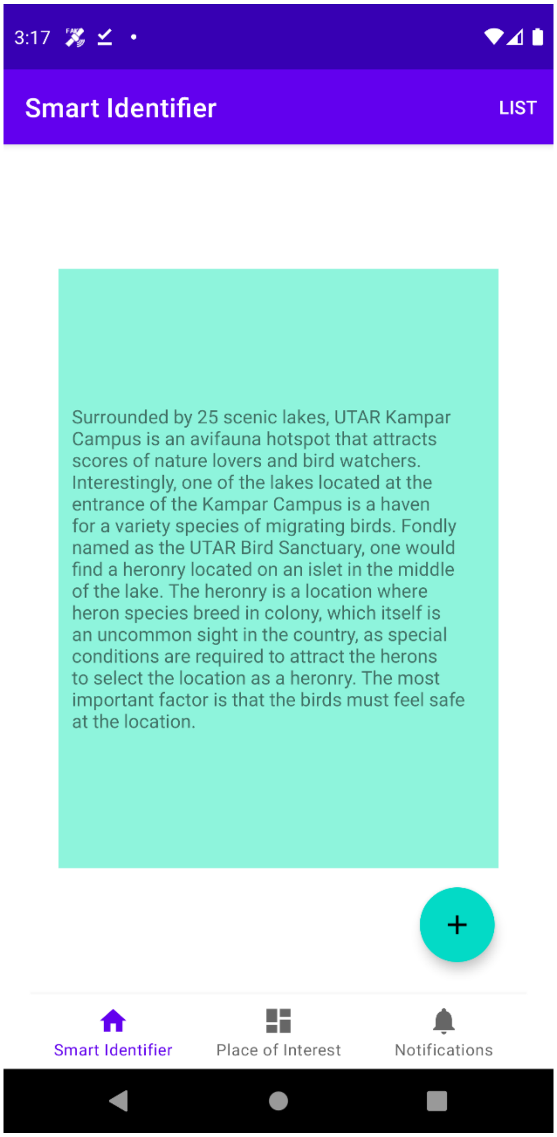
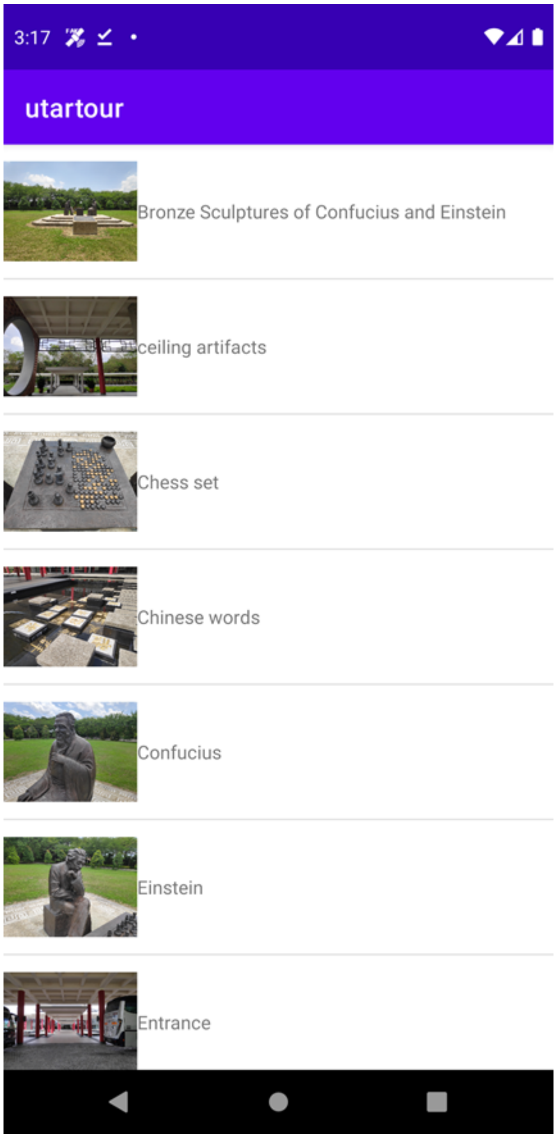


Figure 5.3.5: Photo identified



Surrounded by 25 scenic lakes, UTAR Kampar Campus is an avifauna hotspot that attracts scores of nature lovers and bird watchers. Interestingly, one of the lakes located at the entrance of the Kampar Campus is a haven for a variety species of migrating birds. Fondly named as the UTAR Bird Sanctuary, one would find a heronry located on an islet in the middle of the lake. The heronry is a location where heron species breed in colony, which itself is an uncommon sight in the country, as special conditions are required to attract the herons to select the location as a heronry. The most important factor is that the birds must feel safe at the location.

Figure 5.3.6: Display description



Bronze Sculptures of Confucius and Einstein

ceiling artifacts

Chess set

Chinese words

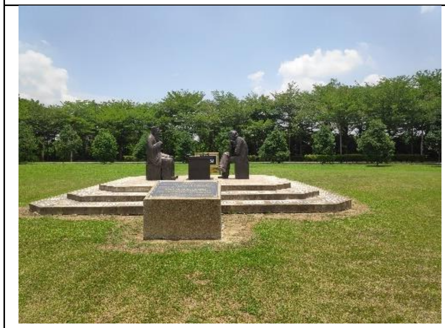
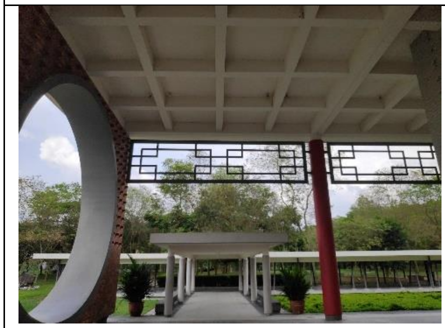
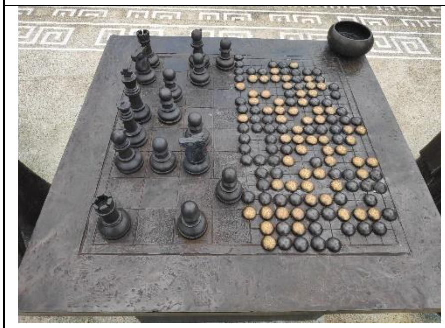
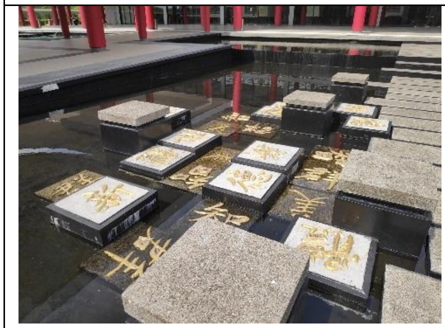
Confucius

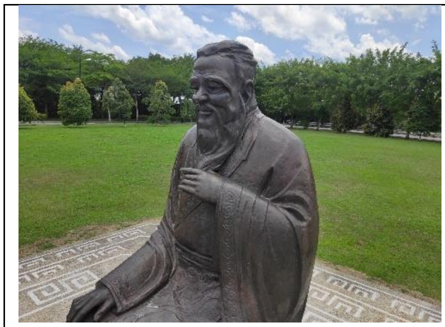
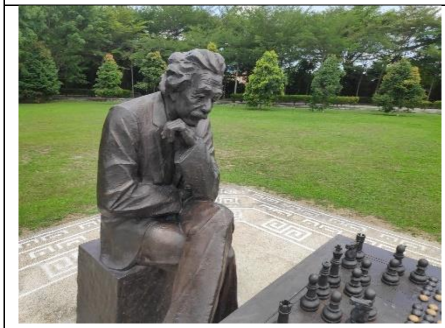
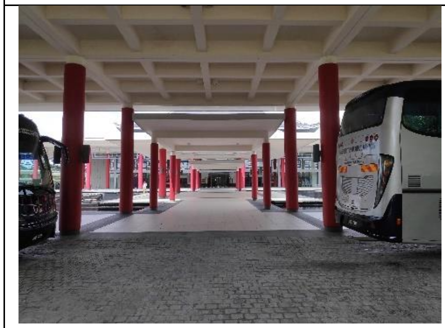
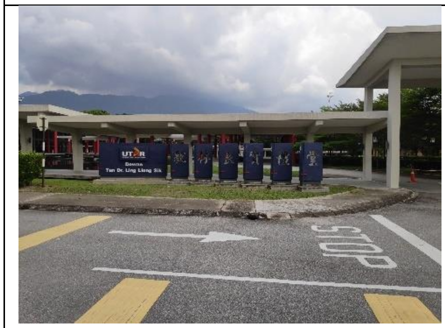
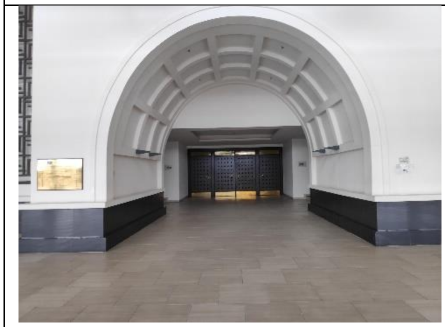
Einstein

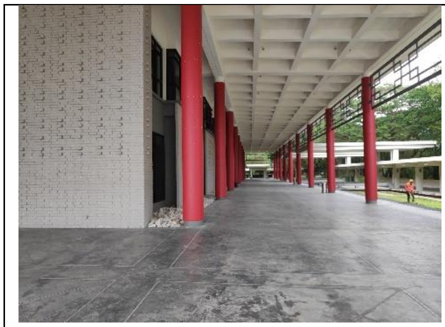
Entrance

Figure 5.3.7 Place list

5.4 Testing

Test case	Output	Correct/test image number	Accuracy (%)
	Bronze Sculptures of Confucius and Einstein	6/10	60
	ceiling artifacts	8/10	80
	Chess set	9/10	90
	Chinese words	6/10	60

	Confucius	5/10	50
	Einstein	5/10	50
	Entrance	6/10	60
	Hall Wordings	8/10	80
	Main entrance door	9/10	90

	Right corridor from entrance	6/10	60
	Round Square	6/10	60
	Round wall	7/10	70
	The UTAR Tree	5/10	50
	The water pool at library	6/10	60

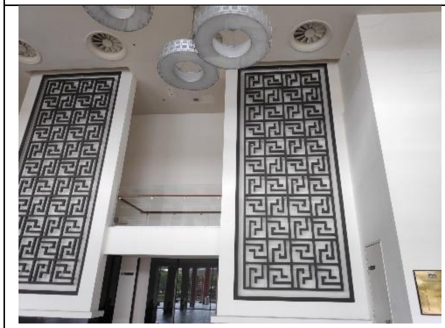
	UTAR Bird Sanctuary	4/10	40
	UTAR Library	8/10	80
	Wall artifacts (inside)	7/10	70
	Image not recognizable	9/10	90

Table 5.4.1 Test case table

The testing images from the test cases above is a mixture of online images and image that has been captured by me. The reason that I mixed some of my own picture to it is because the online images is not much as my photo is very specific for example, the photo of only the chess set. Online does not have many photos of only the chess set so I need to add some of the photo to make it become 10.

In the test result above, there are some places have a lower accuracy compared to other places. One of them is the bird sanctuary. This is because the bird sanctuary is very similar to the UTAR tree in the photos. There are many leaves and not much distinguishing features. I hope to take a closer photo of the bird sanctuary so that it is more distinguishable but I do not have the equipment.

The next one is the “round square” and entrance. They are relatively lower than other because they both look alike. The “round square” is basically just a closer shot of the entrance.

The third one is the Confucius and Einstein sculpture. They also have a lower accuracy as they are really hard for machine to identify because the outline is not obvious and they are of the same color.

For other places, the accuracy is quite acceptable.

Chapter 6: Conclusion

6.1 Project review

In conclusion, this project aims to create an app that helps people identify buildings in UTAR Kampar campus and show them the stories and history behind it. It can help both people and the university because it helps to publicize for the university and the app also can provide answers to the question of users. This app streamlined the delivery of information form the internet to the user. The objective of this project is achieved in the app.

During the development of this project, several challenges had been faced. First being the lack of experience in mobile app. Although I am familiar with the basics of Java programming language, developing an Android app is entirely new for me. So, a lot of research and effort had been put into to develop the app. Second, is the dataset gathering part. All of the dataset is gathered by myself using my own phone so it is very time consuming and memory intensive. Furthermore, capturing using a smartphone has some limitations too for example, the zoom cannot be to high or else the image will be blurry and phone will get very hot from taking so much photos at a very short time.

Despite all the challenges, I am satisfied with this project because I learnt a lot of things. The two main things that I have learnt is training a model and developing an Android app using java. During the training of my model, I improved my Python skill and became more familiar to TensorFlow, a machine learning library. Although I have developed an Android app prior to this project, I coded them using Flutter instead of Java so developing Android app using Java is new to me. However, I am happy that I can complete this app without major issue and learnt a lot of things along the process.

6.2 Novelties and contribution

UTAR Kampar campus has a lot of beautiful spots and scenery but some people nowadays neglected the story and information behind it. So, this app aims to let more people learn more about the buildings in UTAR. Users can use this app to take photo of a building in UTAR and it will show users the information, which is more interactive and quicker compared to searching the information up online. With this app, I hope that this information can be spread faster and let more people know about UTAR. Other than taking a photo of the building to access the information, user also can list the places that the app can recognize and read them by clicking on the places. This is helpful during this time with the pandemic as students cannot go into campus without permission.

6.3 Future work

Although the app can perform most of the main function, there are still some features that can be added to make it more attracting to users. However due to some constraints, it will be the future work.

One of the future works is to add navigation in the app. For example, user can know how to navigate inside the campus from block C to block F. By doing this, users especially freshman will be more motivated to download and use the app.

Other than that, mini games also can be implemented in the app that tells user to take photo with one of the places and the app will reward users with some point that be used can later to redeem some coupons. Collaboration with local shops is needed to get the coupons. This is a great feature to boost the Kampar economy as well as promoting UTAR.

Lastly, I hope that this app can recognize the places in Kampar instead of just UTAR. However due to time constraint, I can only narrow down the scope. However, this will be included in my future work so that my app can attract to a wider variety of users as well as boosting the tourism in Kampar.

Bibliography

Fatima, R., Zarrin, I., Qadeer, M.A. and Umar, M.S. (2016). Mobile travel guide using image recognition and GPS/Geo tagging: A smart way to travel. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/7759893 [Accessed 11 Sep. 2020].

Firdaus, N.M., Chahyati, D. and Fanany, M.I. (2018). Tourist Attractions Classification using ResNet. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8618235 [Accessed 11 Sep. 2020].

Github.io. (2019). PYPL PopularitY of Programming Language index. [online] Available at: http://pypl.github.io/PYPL.html.

Jiafa, M., Weifeng, W., Yahong, H. and Weiguo, S. (2019). A scene recognition algorithm based on deep residual network. Systems Science & Control Engineering, 7(1), pp.243–251.

Statista. (2019). Mobile OS market share 2019 | Statista. [online] Available at: https://www.statista.com/statistics/272698/global-market-share-held-by-mobile operating-systems-since-2009/.

Yue, C. (2016). Recognition of Tourist Attractions Yuanfang Li. [online] www.semanticscholar.org. Available at: https://www.semanticscholar.org/paper/Recognition-of-Tourist-Attractions Yuanfang-Li-Yue/1a996d5556bf0c23da4b6fa309d8e6f2e2323be3 [Accessed 11 Sep. 2020].

Kang, Y., Cho, N., Yoon, J., Park, S. and Kim, J. (2021). Transfer Learning of a Deep Learning Model for Exploring Tourists' Urban Image Using Geotagged Photos. ISPRS International Journal of Geo-Information, [online] 10(3), p.137. Available at: https://www.mdpi.com/2220-9964/10/3/137 [Accessed 15 Apr. 2021].

POSTER

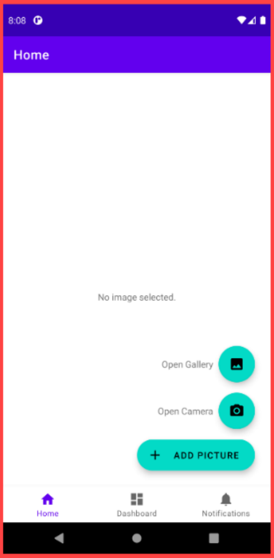
Poster

Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus

Developed by:
Lee Mun Hong

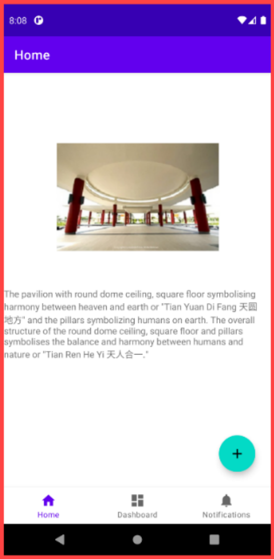
Supervised by:
Dr. Liew Soung Yue

MAIN FEATURES:



No image selected.

User can select photo from gallery of camera to be recognized



The pavilion with round dome ceiling, square floor symbolising harmony between heaven and earth or "Tian Yuan Di Fang 天圆地方" and the pillars symbolizing humans on earth. The overall structure of the round dome ceiling, square floor and pillars symbolises the balance and harmony between humans and nature or "Tian Ren He Yi 天人合一."

Showing the description of the photo taken

Project Background:

Creating an app that can accurately identify the buildings in the photo taken by user or chosen by the user and providing the relevant information such as its history or story and display it on the screen.

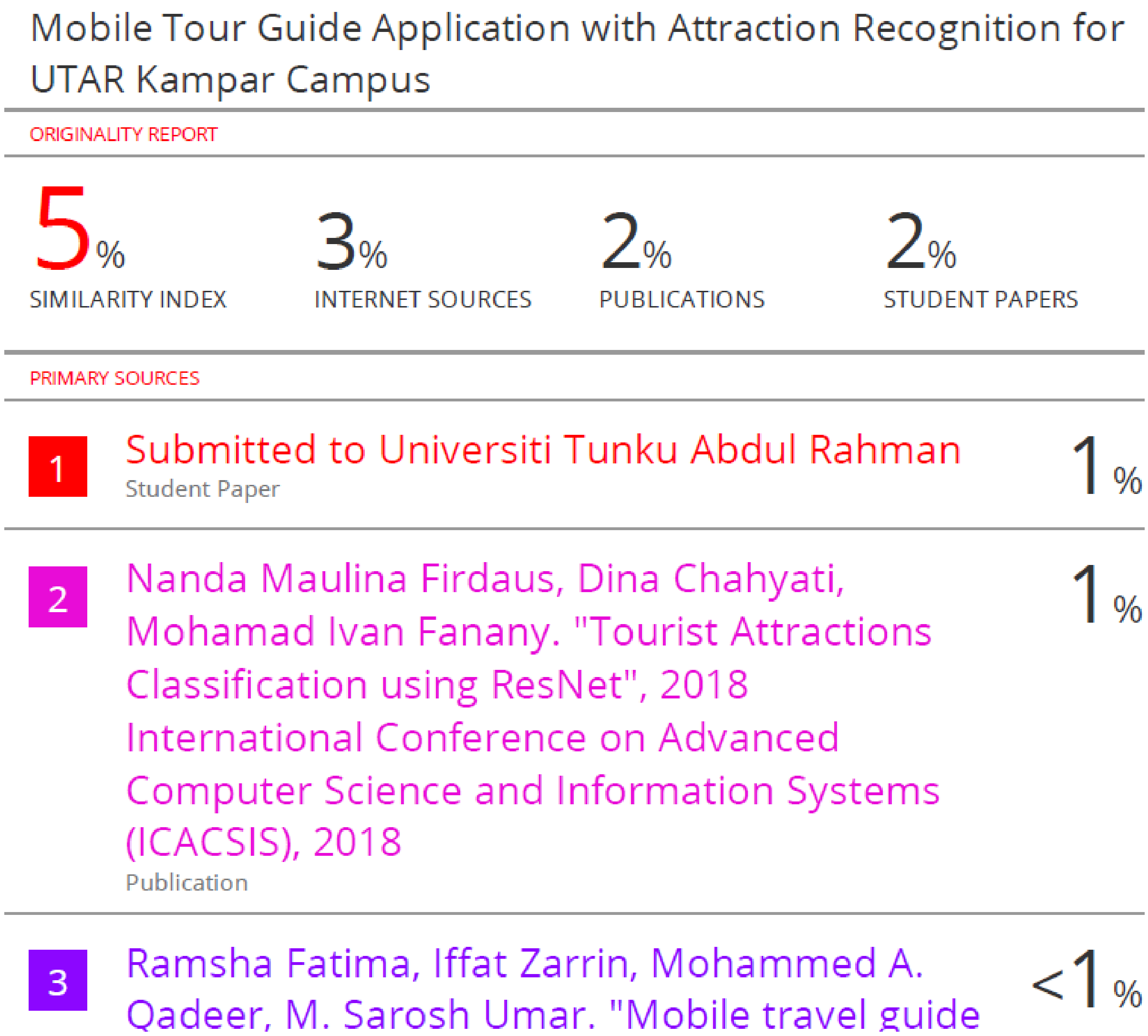
Project Objective:

To promote the stories of UTAR to the public and allow user to know the informations in a more engaged manner .

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

Java Android Studio python jupyter TensorFlow

Plagiarism check result



Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus

ORIGINALITY REPORT

5%	3%	2%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Universiti Tunku Abdul Rahman Student Paper	1%
2	Nanda Maulina Firdaus, Dina Chahyati, Mohamad Ivan Fanany. "Tourist Attractions Classification using ResNet", 2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2018 Publication	1%
3	Ramsha Fatima, Iffat Zarrin, Mohammed A. Qadeer, M. Sarosh Umar. "Mobile travel guide	<1%

PLAGIARISM CHECK RESULT

Universiti Tunku Abdul Rahman			
Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Lee Mun Hong
ID Number(s)	17ACB02902
Programme / Course	Bachelor of Computer Science (Honours)
Title of Final Year Project	Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus

Similarity	**Supervisor's Comments** **(Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 5 % **Similarity by source** Internet Sources: 3 % Publications: 2 % Student Papers: 2 %	Within the required range
Number of individual sources listed of more than 3% similarity: 0	Within the required range
Parameters of originality required and limits approved by UTAR are as follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

____
Signature of Supervisor
Name: Liew Soung Yue

Date: 16/4/2021

____
Signature of Co-Supervisor
Name: ______________________

Date: ______________________

Weekly Report

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 1**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-Identifying new requirement

2. WORK TO BE DONE

- Identifying new requirement

3. PROBLEMS ENCOUNTERED

-

4. SELF EVALUATION OF THE PROGRESS

-

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 3**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE [Please write the details of the work done in the last fortnight.] -Gathering new dataset and retraining the model
2. WORK TO BE DONE - Gathering new dataset and retraining the model
3. PROBLEMS ENCOUNTERED **-**
4. SELF EVALUATION OF THE PROGRESS **-**

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 5**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Gathering new dataset and retraining the model

2. WORK TO BE DONE

- Gathering new dataset and retraining the model

3. PROBLEMS ENCOUNTERED

-

4. SELF EVALUATION OF THE PROGRESS

-

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 7**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-Redesigning the interface

2. WORK TO BE DONE

- Redesigning the interface

3. PROBLEMS ENCOUNTERED

-

4. SELF EVALUATION OF THE PROGRESS

-

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 9**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-Creating the list interface for places

2. WORK TO BE DONE

- Creating the list interface for places

3. PROBLEMS ENCOUNTERED

-

4. SELF EVALUATION OF THE PROGRESS

-

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 11**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE [Please write the details of the work done in the last fortnight.] -Fixing existing bugs
2. WORK TO BE DONE - Fixing existing bugs
3. PROBLEMS ENCOUNTERED -
4. SELF EVALUATION OF THE PROGRESS -

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.: 13**
Student Name & ID: Lee Mun Hong 1702902	
Supervisor: Dr. Liew Soung Yue	
Project Title: Mobile Tour Guide Application with Attraction Recognition for UTAR Kampar Campus	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

-Fixing existing bugs

2. WORK TO BE DONE

- Fixing existing bugs

3. PROBLEMS ENCOUNTERED

-

4. SELF EVALUATION OF THE PROGRESS

-

Supervisor's signature

Student's signature

Checklist

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	17ACB02902
Student Name	Lee Mun Hong
Supervisor Name	Dr. Liew Soung Yue

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Front Cover
√	Signed Report Status Declaration Form
√	Title Page
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
N/A	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
N/A	Appendices (if applicable)
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report. ________________________ (Signature of Student) Date: 14.04.2021	Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction. ________________________ (Signature of Supervisor) Date: 16/4/2021