

GroceMate Report

Teng Wang

Github Link : <https://github.com/TiaWang19910725/GroceMate-Project.git>

1. Project Statement

What app do you want to develop?

GroceMate is an android app to help people locate a specific product in grocery store quickly.

What can the app do?

A user should provide the product name that he/she is looking for. The application will find six related images online and calculate most important features. When this user is moving in grocery store while holding the cellphone, the app will highlight the product in the image if it is detected.

Why do you want to develop such an application?

Shopping in a grocery store to look for a specific product usually costs a lot of time. Now product packages are colorful and some of them are very similar, piling hundreds of products on the same shelf makes it hard for people to pick out their desired product.

Luckily a store prefers to places similar products all in the same location. It makes it possible to design the product detection if we can find the features of a single product package that can match some of them placing next to each other. Also we want this app to adapt to every grocery store instead of a specific one. So instead of building up complicated recognition models for every possible product, we want the feature we choose to be as general as possible.

What kind of users will benefit from your application?

Every common customer can download GroceMate and use it in every grocery store. GroceMate can be especially helpful for people with vision and moving disabilities.

Is there any similar application already available on the Android Market?

Most current apps focus on understanding a specific class of objects and tracking it. For example, TensorFlow is a famous tool for image learning, but it mainly focuses on understanding general object classes such as “Apple”, “Table” instead of tracking any specific product. The detection of a specific products depends on pre-defined complex neural network models.

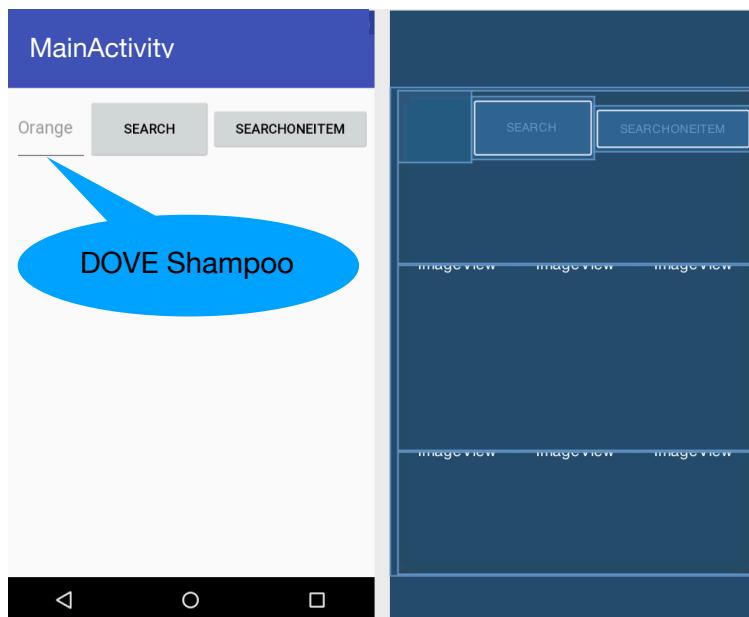
Is there any special requirement for using the app?

It only requires an android phone with camera, network and external storage permission.

2. Application Design

**What modules does your app have and what functions does each module provide?
External services such as a Web service, describe those services and how your app interacts with the services.**

You also need to provide a detailed description of each module.



Brief introduction :

This is the Main Activity, where the app begins with.

A user starts by typing in the product name that he/she is looking for. For example, DOVE Shampoo.

By clicking SEARCH, top 6 images will show up below.

SEARCHONEITEM button will bring you to the Camera page.

Details about this page:

An EditText is used to collect user's input.

```
input = (EditText) findViewById(R.id.input_item);
```

Then SEARCH button generates an URL in FLICKR format and collects JSON response.
This part of code can be found in file URL_functions.java and

The URL is in format : "https://api.flickr.com/services/rest/?

```
method=flickr.photos.search&api_key=cf638d1a814322c39e845db2e4c72138&tags=Dove"
```

An AsyncTask is running in background to send this URL and get JSON response.

The response is in format : jsonFlickrApi({ "photos": { "page": 1, "pages": 4580, "perpage": 6, "total": 27476, "photo": [{"id": "38344258784", "owner": "30478819@N08", "secret": "c1941b6469", "server": "4535", "farm": 5, "title": "Cold Medicine on a White Background", "ispublic": 1, "isfriend": 0, "ismember": 0 }] }}

Every image can be downloaded at http://farm{farm-id}.staticflickr.com/{server-id}/{id}_{secret}.jpg.

In this process, **network permission** should be required by :

```
<uses-permission android:name="android.permission.INTERNET" />
```

Downloaded images are shown in each ImageView in Bitmap format.

An R/G/B ratio is calculated from each bitmap, this part of code is in file ImageHandlers.java. So six images should have six <R,G,B> value tuples.

For example, the DOVE SHAMPOO RGB percentages: <0.3209569 0.3351231 0.34391996>

TaskCompleted.java is an interface to carry these ratio value we get in AsyncTask back to MainActivity by **mCallback.onTaskComplete(ratio_list);** Such that both MainActivity and URL_functions.java implements this interface.

These ratio tuples will be carried to next activity by intent.putExtra("color_ratios", toPass); Then an intern from MainActivity to the next Activity is started.



Brief Introduction:

This activity opens camera and draw a rectangular on the area where the object is.

Details:

This activity requires camera permission

```
<uses-feature  
    android:name="android.hardware.camera.any"  
    android:required="true" />  
<uses-feature  
  
    android:name="android.hardware.camera.autofocus"  
    android:required="true" />  
  
<uses-permission  
    android:name="android.permission.CAMERA" />
```

More Details:

This layout contains two SurfaceViews in exactly the same size: CameraView and TransparentView.

CameraView shows realtime vision of the cellphone. If targets are found, it will pass coordinates of targets to TransparentView to draw a rectangle.

This Activity implements SurfaceHolder.Callback and Camera.PreviewCallback

In this activity, it calls ImageUtilize.java to analyze every frame grid by grid which are received from the camera preview. Initially grid size is set to be 1/100 of the whole image, which means grid_x = image.WIDTH/100, grid_y = image.HEIGHT/100. This is a small grid that can put at least one item in a grid if the camera is holding within a rational distance from the object. We calculate the <R,G,B> ratio of this small grid. If at least one grid detects the target, it will check out its neighbors. All neighbors showing detection of the target can be merged into one large grid. The coordinate of this large grid will be sent to TransparentView to draw a rectangular with Canvas paint.

CameraPreview keeps calling back to analyze the video until the app is terminated.

What type of device does your app target (smartphone, tablet, or both)?

The app can run successfully on my EMUI android 7.0 phone.

3. Application Implementation and Evaluation

What classes have you defined and implemented in the app.

This app contains Two activities: MainActivity and activity3(the camera activity)

Two classes to process images: ImageUtilize and Image_Handlers

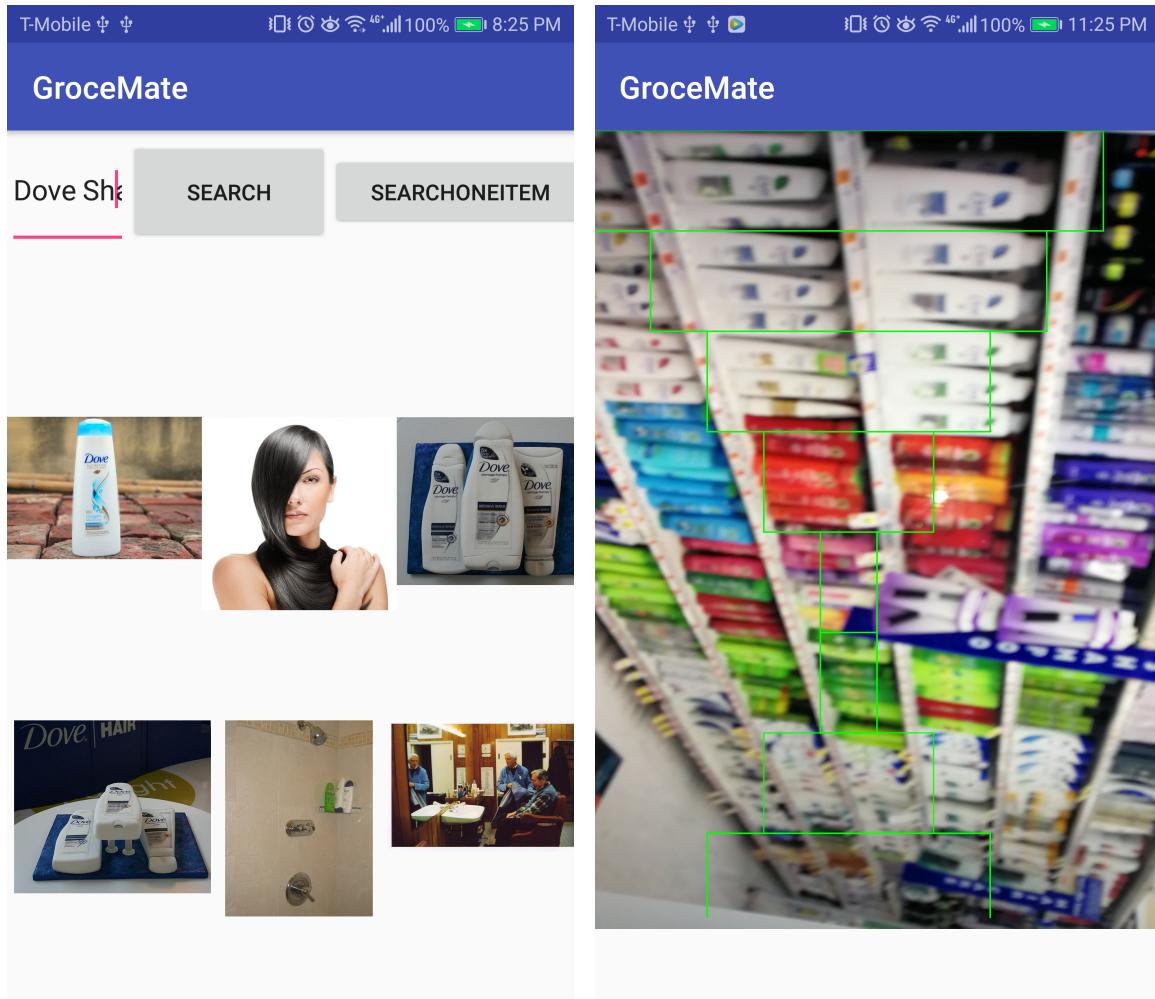
Two class to handle FLICKR API: Flickr_functions and URL_functions

An interface to pass values between AsyncTask and activity: TaskCompleted

How did you test the app, including functionality, performance, and usability?

Does the app always work correctly? What errors or problems have you identified during the testing? Have you been able to solve them?

<1> Dove Shampoo:



Good result:

The color ratio of the 3rd graph is 0.3209569 0.3351231 0.34391996, we can see that green and blue color is having a higher weight which matches the image. During grocery store testing, we see that most DOVE are detected and reported correctly.

Image searching takes around 2 seconds in my network environment. At least five calculation can be finished within one second. The fast calculation guarantees that the app can work smoothly during moving.

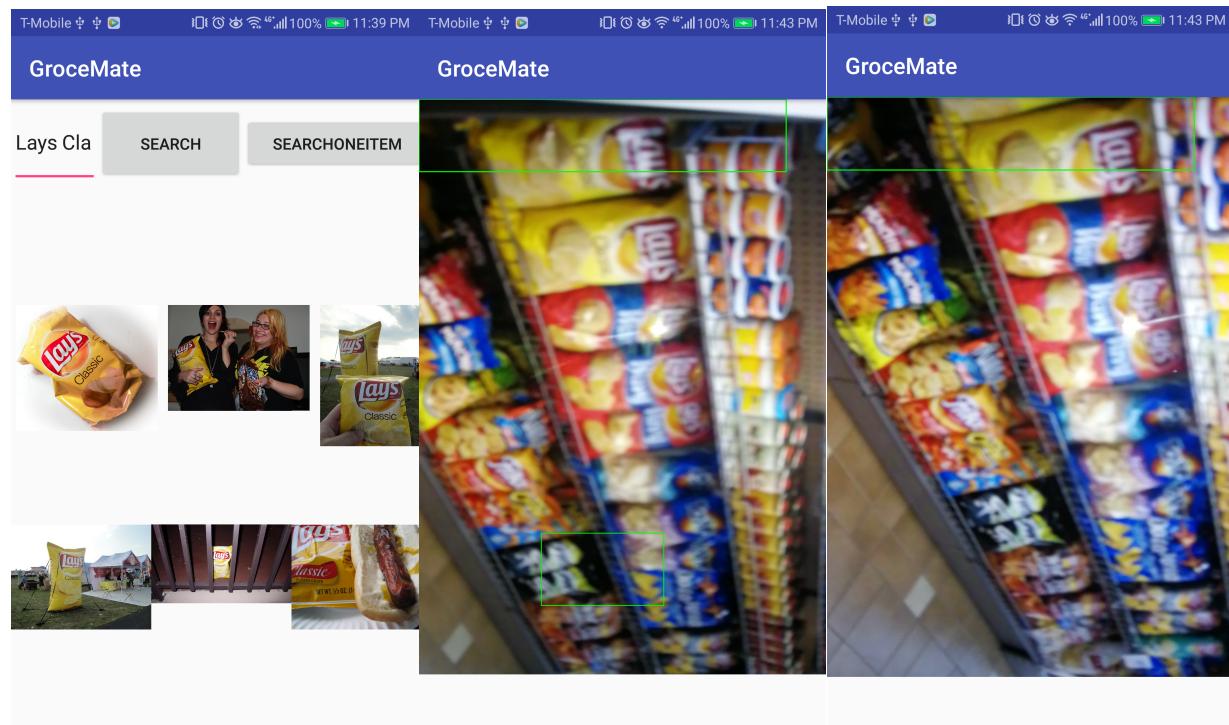
Limitation:

We can see that images we downloaded from FLICKR are not exactly the object that we are looking for. This is one limitation of FLICKR photo pool. The 2nd, 5th and 6th images cannot be considered as a model in the future comparison. In the testing we can see that some red bottles are pointed out, too. That part matches the <RGB> of photo6, that is why it is reported, too. Currently our app is not able to eliminate downloaded images that does not match what we are searching, this is one limitation of our current app.

Solution:

FLICKR supports searching with many parameters. Here we only search the tag of the image, which may be the reason why some downloaded images are not related.

<2> Lays Classic



4. References

FLICKR offers image searching API at <https://www.flickr.com/services/api/flickr.photos.search.html>

Andoird how to manipulate camera: <https://stackoverflow.com/questions/5991319/capture-image-from-camera-and-display-in-activity>

TensorFlow on Android: <https://github.com/tensorflow/tensorflow/tree/master/tensorflow>

5. Experiences and Thoughts

Share your experiences and thoughts during the project. For example, you may describe here the functions that are nice to have for your app but you were not able to finish due to time limit.

The app should add pronouncing function to tell people with vision disability that the object is detected, and say the direction for them to follow.