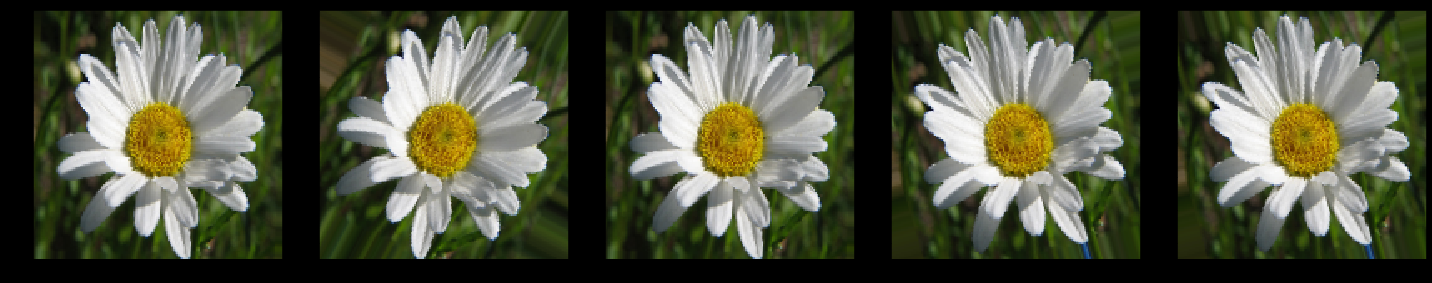
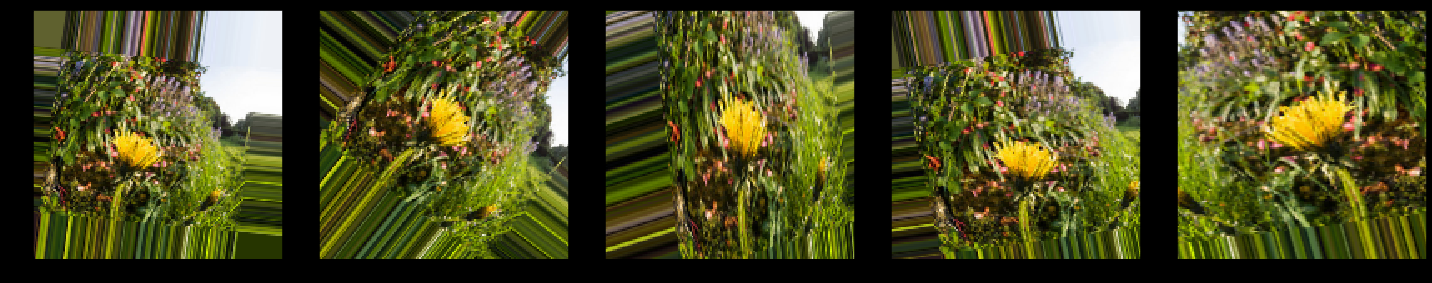
Google Code-In Object Detection App

I have created a mobile app that classifies 5 types of flowers:

1. Daisy
2. Dandelion
3. Roses
4. Sunflowers
5. Tulips.

I used TensorFlow to create the deep learning model. TensorFlow uses the Python language. I created the app using Android Studio which uses Java.

Creating the TensorFlow model: <https://github.com/SiddhantAttavar/FlowerDetection>

1. First I imported some libraries
2. Then I downloaded the tf\_flowers database from this url: [https://storage.googleapis.com/download.tensorflow.org/example\_images  
   /flower\_photos.tgz](https://storage.googleapis.com/download.tensorflow.org/example_images/flower_photos.tgz)
3. 
4. Then I split the database into training and validation datasets
5. I then applied data augmentation to increase the available data and reduce over-fitting in the following ways:
   1. Horizontal flip
   2. Rotation
   3. Zoom
   4. Width shift
   5. Height shift
   6. 
6. I used **tensorflow.keras.preprocessing.image.ImageDataGenerator** to create batches for training. In this step I also set the size of the images to be 150\*150
7. I then created the TensorFlow model. The layers of this model were:
   1. CNN layer
   2. Max Pooling layer
   3. CNN layer
   4. Max Pooling layer
   5. CNN layer
   6. Max Pooling layer
   7. Flatten layer
   8. Dropout layer (to reduce overfitting)
   9. Dense layer
   10. Dropout layer
   11. Dense layer
8. Then I compiled the model using the adam optimizer and the sparse categorial cross entropy loss function. The model had a validation accuracy of about 80%
9. Then I saved the model as a .tflite file using the tensorflow.lite.TFLiteConverter.from\_saved\_model() function and downloaded it

Creating the android app:

1. I first added this dependency: implementation 'org.tensorflow:tensorflow-lite:+'
2. I then added the camera permissions:  
   <uses-feature android:name="android.hardware.camera" android:required="true" />  
   <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
   <uses-permission android:name="android.permission.CAMERA" />  
   <uses-feature android:name="android.hardware.camera.autofocus" />  
   <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
   <uses-permission android:name="android.permission.CAMERA" />
3. I then copied my model.tflite to the assets folder under app/src/main. The MainActivity.java accessed the file from here
4. I also added a labels.txt to assets folder containing the names of the classes in the correct order.
5. Under the res folder of the app, I created a file\_paths.xml file and in this file I had written the folder in which pictures taken by the camera should be saved
6. I then created the layout for my app using xml under the res folder. The layout consisted of:
   1. An ImageView to display the captured image.
   2. A button for classifying the image
   3. A TextView to display the result
7. Lastly I created the MainActivity.java file.
   1. Here I imported several modules including:
      1. Tensorflow lite interpreter
      2. Modules for reading files
      3. Lists
      4. Bitmap related modules
   2. In the onCreate() method I first initialised all the UI elements, and called functions which took images depending on the Android version. It also called 2 functions which loads the .tflite file and the class names from the assets folder
   3. The image-taking functions take an image using the camera and save them as jpg files under the directory specified in file\_path.xml
   4. After the image is captured, the onActivityResult() method is called the image is also stored as a bitmap, and here it is converted to bytes and stored in ByteBuffer imgData, so that it can be analysed by the TensorFlow.
   5. Tflite.run() is called to do predictions on imgData. The results are stored in an array, and are displayed to the user.

**To recreate my project follow these steps:**

1. Install Android Studio
2. Create an Android Studio project with package com.example.android.flowerdetection;
3. Copy the following files into their respective folders. If they are already present, replace them with the files in the repository. *NOTE: you will have to create the assets folder under app/src/main and add the model.tflite and labels.txt files*:
   1. build.gradle
   2. AndroidManifest.xml
   3. file\_paths.xml
   4. strings.xml
   5. MainActivity.java
   6. activity\_main.xml
   7. model.tflite
   8. labels.txt
4. Run the app on your Android phone