TTTC3413 ROBOT APPLICATION

SEMESTER 1 SESI 2021/2022

PROJECT 1

TITLE

FLOWERS RECOGNITION

LECTURER

DR. ABDUL HADI BIN ABD RAHMAN

GROUP 3

	MEMBER NAME	MATRIC NO.
1.	NURFARIESYA FATIN BINTI AHMAD SAFUAN	A175116
2.	AMAL MAJIDA BINTI MUNIR	A174807

CONTENTS

1.0 INTRODUCTION

There are thousand types of flowers in this world, some only can be found in the jungle, mountains and specific areas. However it is nearly impossible for common people with less knowledge of flower species to identify them accurately. This is because flowers have wide varieties of color, shape and texture. Furthermore, the flower recognition system has also evolved into a valuable system that helps researchers, students and practitioners especially in agriculture, forest, biodiversity, and botany. The two main features of flowers that can make them highly distinguishable are their shape and color. Therefore, our objective in this project is to make a flower recognition system from six types of flowers.

2.0 PROBLEM STATEMENT

1. Consume time to distinguish a huge number of flower's type.
2. Identifying a flower's type just from looking at the features may not be accurate due to human error.

3.0 TECHNICAL SOLUTION

We have obtained the dataset from Kaggle contributors, Alexander Mamaev and Bogdan Cretu. This dataset contains 5 types of flower images which are

- Daisy
- Dandelion
- Rose
- Sunflower
- Tulip

There are 4317 images in this dataset. However after analyzing each image, we have removed 517 corrupted or blur images.

In addition, we add another type of flower in this dataset, Hibiscus. There are a total of 616 images of Hibiscus flowers added.

Therefore, there are 6 types of flowers and a total of 4416 images used in this project.

To solve this problem, we decided to use convolutional neural networks (CNN). CNN is useful in identifying the flower's type using the fact that the appearance of the flower is easily

distinguishable. Then, the image augmentation with various settings is also applied on the training dataset in order to improve the classification performance.

4.0 DISCUSSION

In this project, we used a random split method with a ratio of 0.75 training set and 0.25 testing set. After testing and observing the result for multiple times, we conclude that this ratio produced the highest accuracy, consistent and best result. Furthemore, the accuracy of this model also increased after we manually filtered out the dataset to make sure no defective, blur and corrupted images are used in this model.

```
[ ]  #ratio train:test = 0.75:0.25
     x_train,x_test,y_train,y_test=train_test_split(X,y,test_size=0.25,random_state=42)
```

Figure 1.0: Random Split with ratio 0.75:0.25

For data augmentation, we use the ImageDataGenerator function. In this function, we applied a total of five techniques which are rotation_range, zoom_range, width_shift_range, height_shift_range, and horizontal_flip.

```
from keras.preprocessing.image import ImageDataGenerator

datagen = ImageDataGenerator(
        featurewise_center=False,  # set input mean to 0 over the dataset
        samplewise_center=False,  # set each sample mean to 0
        featurewise_std_normalization=False,  # divide inputs by std of the dataset
        samplewise_std_normalization=False,  # divide each input by its std
        zca_whitening=False,  # apply ZCA whitening
        rotation_range=10,  # randomly rotate images in the range (degrees, 0 to 180)
        zoom_range = 0.1, # Randomly zoom image
        width_shift_range=0.2,  # randomly shift images horizontally (fraction of total width)
        height_shift_range=0.2,  # randomly shift images vertically (fraction of total height)
        horizontal_flip=True,  # randomly flip images
        vertical_flip=False)  # randomly flip images


datagen.fit(x_train)
```

Figure 1.1: Data Augmentation

The number of epochs we used is 60 as it resulted in a high accuracy and minimum loss. We also test with the number of epochs as 30, and 100 however 30 gives a lower accuracy while 100 consumes a lot of time, ram and gpu usage in our computer. The number of batch sizes implemented in this project are 96, 128 and 138. After multiple attempts, we decided to go with 128 as it produced the highest accuracy and best result compared to others.

```
Epoch 55/60
25/25 [==============================] - 198s 8s/step - loss: 0.2292 - accuracy: 0.9130 - val_loss: 0.5632 - val_accuracy: 0.8324
Epoch 56/60
25/25 [==============================] - 197s 8s/step - loss: 0.2141 - accuracy: 0.9262 - val_loss: 0.6594 - val_accuracy: 0.8134
Epoch 57/60
25/25 [==============================] - 199s 8s/step - loss: 0.2446 - accuracy: 0.9055 - val_loss: 0.7184 - val_accuracy: 0.8062
Epoch 58/60
25/25 [==============================] - 198s 8s/step - loss: 0.2425 - accuracy: 0.9102 - val_loss: 0.5968 - val_accuracy: 0.8243
Epoch 59/60
25/25 [==============================] - 198s 8s/step - loss: 0.1945 - accuracy: 0.9262 - val_loss: 0.6428 - val_accuracy: 0.8279
Epoch 60/60
25/25 [==============================] - 197s 8s/step - loss: 0.1949 - accuracy: 0.9287 - val_loss: 0.5897 - val_accuracy: 0.8397
```

Figure 1.2: Epochs Result

The accuracy and loss we obtained from this project for both training and testing dataset are:-

Result	Training	Testing
Accuracy	0.9287	0.8397
Loss	0.1949	0.5897

Table 1.0 : Accuracy and Loss Result

Lastly, we used random samples on the testing dataset to make predictions. Here we display a total of 36 images from the test dataset with its own label and the model predictions. The green colour is the correct classification while the red colour is the misclassified flower as shown in the figure below:-

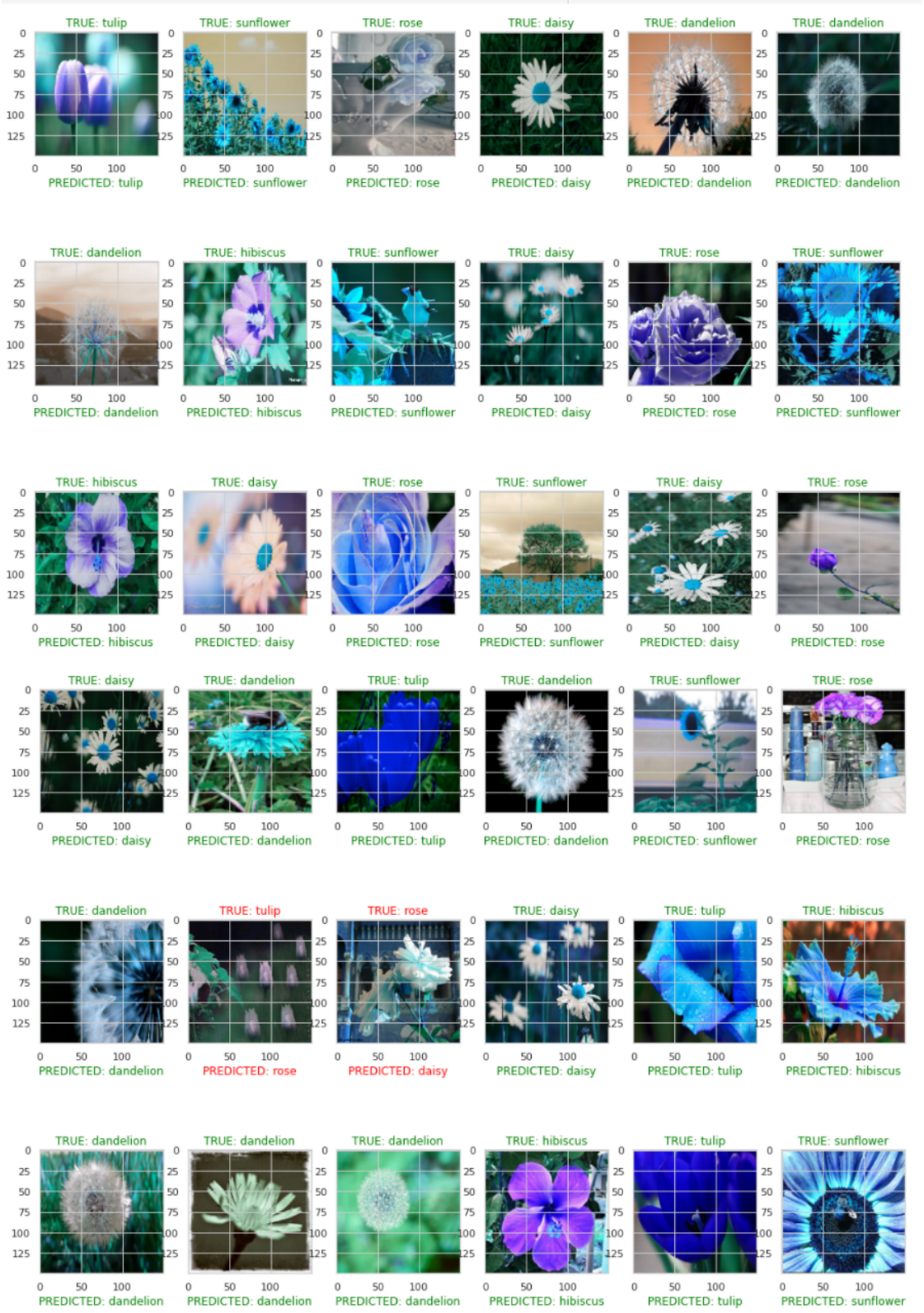

Figure 1.3: Predictions Result

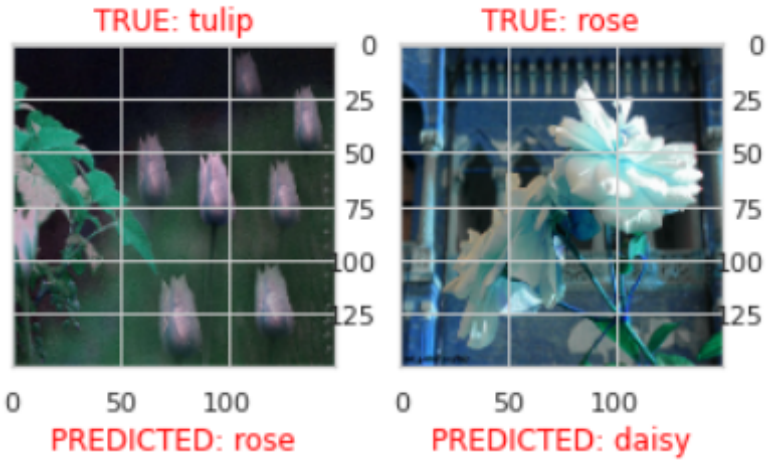

Figure 1.4 : Misclassified Flower

5.0 CONCLUSION

In conclusion, we used CNN model to perform a flower recognition for this project. Based on the experimental results, it shows that we obtained a good accuracy. The percentage of training accuracy achieved in this recognition of flowers project is approximately 92.87% while validation accuracy is 83.97%. This may be due to the number of convolutional layers that we used in this model. We designed four convolutional layers where each convolutional layer has a different number of filters which improve the accuracy in recognition. Overall, CNN has proven to be one of the most efficient to recognize objects. As there are millions of flower types around the world, CNN is the best algorithm to be used in flower recognition for recognizing patterns such as shapes, color and textures of flowers.

6.0 REFERENCE

1. Bae, K. I., Park, J., Lee, J., Lee, Y., & Lim, C. (2020). Flower classification with modified multimodal convolutional neural networks. *Expert Systems with Applications*, *159*, 113455. https://doi.org/10.1016/j.eswa.2020.113455

2. Lv R, Li Z, Zuo J, Liu J (2021) Flower Classification and Recognition Based on Significance Test and Transfer Learning. 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE). https://doi.org/10.1109/iccece51280.2021.9342468

3. Narvekar, C., & Rao, M. (2020). Flower classification using CNN and transfer learning in CNN- Agriculture Perspective. *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*. https://doi.org/10.1109/iciss49785.2020.9316030

4. *tf.keras.preprocessing.image.ImageDataGenerator*. (2021). TensorFlow. https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator