# COMP3007 Flower Classification and Segmentation

Benjamin Jessel - 20255224


## ABSTRACT

*This paper investigates the effectiveness of deep Convolution Neural Networks for the classification and segmentation of flower images using the Oxford Flower Dataset. The objective was to develop 2 CNN models: one for classifying flower images by species and one for segmenting flower images into background and foreground.*

*The classification model was trained on 17 classes of 80 images each and provided an accuracy of 54.41% when evaluating on 256x256 RGB images. The segmentation model was trained on 80 images from a single class and provided an accuracy of 93.56%. It was shown that CNNs can be effective for the classification and segmentation of flower images.*


## 1. INTRODUCTION

Image classification and segmentation are well established topics due to their requirement for emerging automated technologies, such as self-driving vehicles and license plate recognition, although most established research centre on datasets containing multiple categories of object, such as Caltech256[1], this paper focuses on classifying different classes of flower.

Flowers species can be a difficult thing for even humans to classify due to their similar shapes and colours, often with small details that differ between categories that may be unnoticeable at first glance, as seen in figure 1. Because of this a machine learning classifier would be useful to botanists who prior to machine learning would rely on pattern recognition using features that often had to cause damage to the flower to detect[2]. In addition to these similarities between classes flowers, unlike more regularly shaped objects such as cars, vary in shape between members of the same category, due to their random nature. This poses a challenge for a machine learning model as it cannot rely on repeating patterns between images to classify them but must instead learn colours, textures and features of the flowers to classify them.

This paper will aim to classify a dataset of flowers into their respective species and segment one of these species into a binary image containing only flower pixels by using Convolution Neural Networks.

*Figure 1 An example of flowers from two different categories. Both flowers are very similar despite being from different species.*

*Figure 2 An example of 3 flowers taken from the same category. All 3 flowers consist of different colours, textures and shapes that can make it hard to group them as the same species.*

### 1.1 OVERVIEW

In this paper I will develop 2 Convolution Neural Network models to be used on the dataset provided for the purposes of classifying flower images by their species, and segmenting daffodil images into foreground and background.

For the classification model, I will process and augment the images before training a CNN consisting of 3 layers of convolution, batch normalisation, ReLU activation and pooling, with varying filter sizes to learn different complexities of features in the image sets. Results are evaluated on the validation data through confusion matrices and classification accuracy.

For or the Segmentation model, I will process the images to remove unwanted labels and focus on the flower type as our foreground. I will train a CNN consisting of 2 up-sampling layers followed by 2 down-sampling layers, before classifying the data as either flower or background. The result pixel classifications are then evaluated using a confusion matrix and weighted IoU.

### 1.2 DATASET

The dataset used for the models consisted of 17 categories of flower with 80 flowers per category from the Oxford Flower Dataset [3]. The images in the dataset vary in setting, with some images containing sky backgrounds and others containing grass or in some cases other flowers. Images also are taken from a variety of angles and distances, with some being taken from straight on and others taken level with the flower. The flower sets are such that individual categories have flowers that vary in colour and shape to provide a wide range of data that should help to reduce overfitting to certain flower features when categorising the flowers.

## 2.1 CLASSIFICATION METHOD

In order to train a flower classifier on the images in our dataset we need to consider how the data is to be presented to the model, as colour, patterns and textures of the flower petals are all defining features between the categories and thus should be preserved in the images provided. I first separate the flowers into 17 folders named by their category. Each folder contains 80 images and allows us to load the images into an image Datastore using the names of each subfolder as the category label for the flowers within, where images in folder 17flowers\daffodil belong to the daffodil category. The dataset is split between training and validation data, with 80% of images from each category being randomly selected as training data and the remaining 20% being used for validation to ensure that the datasets used are balanced as each category is represented equally within the training and validation dataset.

The datasets were then transformed, resizing each image to be 256x256 pixels with 3 channels per image, ensuring that colour data remained and that images were the same size for training. Images then underwent a random augmentation, consisting of Scaling the image between [0.95, 1.05] to increase and decrease the size of the flowers, rotating between [-45, 45] degrees to provide a variety of petal orientations and X and Y shearing of between [-10, 10] degrees to counter the difference in angles between flower images. The goal of this augmentation was to provide a more general dataset that would be effective against a larger range of unseen flower images.

*Figure 3 An example of 6 images from the training dataset post augmentation.*

The CNN used to classify the flowers was based on a network used in Flower classification using deep CNN[4], with a reduced number of blocks to save model training time. The CNN consisted of an Image Input layer, followed by 3 blocks of convolution, and then finally a fully connected layer, SoftMax layer and classification layer. Each convolution block consists of a 4x4 filter, chosen to ensure the details of the flower can be captured over a large area, a batch normalisation layer to improve the training gradient of the model, followed by a ReLU layer to introduce non-linearity and a pooling layer with stride 2 to reduce the output size. Each block was conducted with an increasing number of filters, of 64, 128 and 256, to ensure that we capture finer details as well as larger patterns in the image. The model was trained for 15 epochs with a mini batch size of 64 images, which proved to be enough training time to provide sufficient results without the risk of overfitting.

### 2.2 SEGMENTATION METHOD

For the Segmentation model I organised the images into 2 folders, daffodilSeg\ImagesRsz256 for flower images and daffodilSeg\LabelsRsz256 for the pixel labels for the corresponding images. Images were then loaded into an image Datastore and the pixel labels into a Pixel Label Datastore. The images provided in the dataset were already 256x256x3 so did not need resizing however the labels consisted of 5 classes, flower, leaf, background, sky and null but as we are only interested in segmenting the flowers from the background these labels were removed from the datastore by saving the pixel labels as a binary image, masked with whether each pixel was labelled as flower or not. These images were then reloaded from a temporary folder to be used for the model. The image and pixel datasets were split randomly into 80% training data and 20% validation data and combined into one training datastore and one validation datastore.

The CNN used for segmentation consisted of 2 blocks of down-sampling convolution layers followed by 2 blocks of up-sampling convolution layers and finally one block of output layers. The down-sampling blocks consisted of a 2d convolution layer with 3x3 filter size and 64 filters, followed by a ReLU activation layer and a 2x2 Pooling layer with stride 2, which aimed to reduce the complexity of the model due to the large amount of data needing to be classified while still learning the details of the images. The up-sampling blocks consisted of a transposed 2d convolution layer with a 4x4 filter and 64 filters, followed by a ReLU layer, which aimed to return the network to the original special dimensions. The output block consisted of a final convolution layer with a filter size of 1x1 and 4 filters, 1 for each class in the original dataset, followed by a softmax and pixel classification layer, with class weights that favoured flower against background classifications 2-1 to make up for the class imbalances between background pixels and flower pixels, to generate the final classifications for the model. After classification I used an opening filter to erode, then dilate the image with a 2x2 filter to remove small artifacts and create a cleaner final output classification.

## 3 EVALUATION

The results in this section were gathered from randomly assigned validation sets that consisted of 20% of the dataset provided for the task. Results consist of classification accuracy, precision, recall, f1 score and confusion matrices.

### 3.1 CLASSIFICATION EVALUATION

The classification model performed moderately with a mean accuracy of 0.5735 when asked to classify images from the validation set. The model provided a precision of 0.5735 and recall of 0.6546.

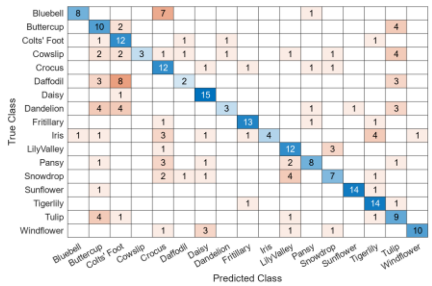


*Figure 4 The confusion matrix produced by predicting labels from the validation set.*

However, when predicting values from the training set the model achieved a mean accuracy of 0.7785 and f1 score of 0.8062. This disparity in performance between the training data and the validation data suggests that the model had become overfit to the training data.

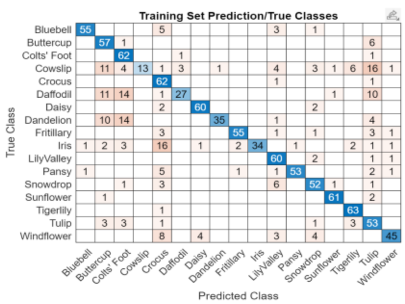


*Figure 5 The confusion matrix produced by predicting labels from the training dataset.*

### 3.2 SEGMENTATION EVALUATION

The segmentation model performed well with a mean accuracy of 0.92165 pixels correctly categorised by the model. I achieved a precision of 0.9216 suggesting that a high proportion of the pixels identified as flower were in fact flowers, additionally a recall of 0.9144 suggesting that most of the true flower pixels were correctly identified, and an f1-score of 0.9180. The model had a mean IoU of 0.85058 showing that it was effective at finding the boundaries between foreground and background.

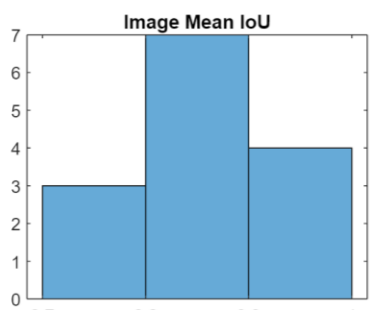


*Figure 6 Mean IoU histogram of the segmentation model*

To better understand how the model categorised the pixels a confusion matrix was produced (figure 7), showing the percentage of pixels categorised correctly and incorrectly. The confusion matrix shows that the model was better at correctly identifying background than at identifying flower pixels, with 10.8% of flower pixels being incorrectly

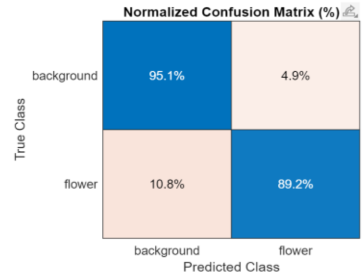


*Figure 7 The confusion matrix produced by the segmentation model.*

labelled as background compared to 4.9% of background pixels being labelled as flowers.

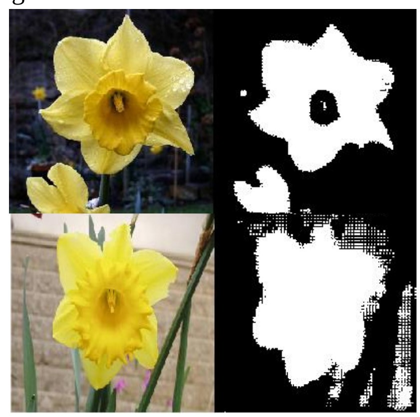

*Figure 8 An example of an original image from the validation set next to the binary image produced.*

When comparing the original images to the corresponding binary image produced the setbacks of the model become clear. As shown in figure 8, the model struggled to identify the centre of flowers as part of the flower, possibly due to the darker shadows in the image. It did however identify a smaller flower in the background suggesting that the model relies primarily on colour to identify the flowers.

### CONCULSION

In this paper developed and evaluated 2 deep convolution neural networks for the purposes of classification and segmentation of flower images. I have shown that deep convolution neural networks can be effective at classifying flower species.

The results gathered varied between the models, with the classification model having an f1 score of 0.5609, while the segmentation model had an f1 score of 0.9180. The classification model was able to classify flower images to an extent but was found to suffer from overfitting so requires further tuning of the hyperparameters to produce accurate results.

On the other hand, the segmentation model proved to be very effective at classifying flower pixels from background pixels, with most pixels that were incorrectly identified as background being at the very centre of the daffodils, and most pixels incorrectly identified as flower being light, textured parts of the background image, such as a wall section.

Overall, both models show promise of being useful to some capacity in their respective field, with results that prove the models are successful in their task, although require refinement before they were to be used accurately.

## REFERENCES

[1] G. Griffin, A. Holub, and P. Perona, “Caltech-256 Object Category Dataset,” Mar. 10, 2007. https://resolver.caltech.edu/CaltechAUTHORS:CNS-TR-2007-001 (accessed May 02, 2023).

[2] T. Elpel, *Botany in a Day. HOPS Press,*. HOPS Press, 2013.

[3] “Visual Geometry Group - University of Oxford.” https://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html (accessed May 03, 2023).

[4] H. Hiary, H. Saadeh, M. Saadeh, and M. Yaqub, “Flower classification using deep convolutional neural networks,” *IET Comput. Vis.*, vol. 12, no. 6, pp. 855–862, Sep. 2018, doi: 10.1049/iet-cvi.2017.0155.