FLOWER CLASSIFICATION AND SEGMENTATION USING DEEP CONVOLUTIONAL NEURAL NETWORK

Alden Zheng Heng Sia
School of Computer Science
University of Nottingham, United Kingdom
hfyas6@nottingham.ac.uk

ABSTRACT

This study represents the development and evaluation of several convolutional neural networks for flower classification and segmentation. Several state-of-the-art CNN architectures such as **ResNet-50** and **GoogleNet** will be used to compare with the CNN model build from scratch to perform flower classification and segmentation. The 17-category flower dataset provided will be used to evaluate the performance of classification models and a subset of images with groundtruth labels are used for segmentation. Validation and testing accuracy, f1-score, mean IOU, and boundary F1 will be measured to compare the models. Results show that the ResNet50 model outperforms classifying flower images which achieve **96.32%** accuracy and the scratch CNN model perform better in segmentation tasks with **97.21%** accuracy with a testing dataset.

Keywords **—** Flower, classification, segmentation, convolutional neural network, deep learning

1. INTRODUCTION

Flowers are ubiquitous around us. Flower classification and segmentation have become significant challenges in recent decades with applications in plant agriculture, biology, and environment monitoring. However, most individuals cannot distinguish the difference between different types of flowers and manual classification is time-consuming and tedious when dealing with many images. Deep learning methods are an excellent approach to accomplishing these issues. Nonetheless, unlike differentiating and segmenting apparent groups such as vehicles, buildings and others, flower classification and segmentation are more challenging tasks due to high intra-class variance and inter-class similarities [1]. Convolution Neural Networks (CNN) have shown effectiveness for these tasks due to the ability to learn and extract features from the images. It is a type of artificial neural network used for image recognition and processing and recently attracted widespread attention due to superior accuracy compared with classical machine learning methods that rely on hand-crafted features [2]. As a result, this report aims to deliver several CNN architectures models to perform flower classification and segmentation with better accuracy.

2. RELATED WORK

Most researchers have employed machine learning-based approaches to segment and classify flowers. For example, Nilsback and Zisserman combined the features using multiple kernel frameworks with SVM algorithms to segment and categories the flower images. They retrieved features from the hue, SIFT and HOG, saturation, and value (HSV) color model. However, this technique requires more computational time and still requires improvement to achieve higher accuracy [3]. Another classification strategy is presented by using weighted Euclidean distance with HSV color model features and boundary shape of flowers but requires manual user intervention [4]. Furthermore, Baumgartner et al. proposed a hierarchical probabilistic model for modelling the segmentation at various resolutions which could yield more diverse and realistic segmentation results [5]. The machine-learning methods proposed achieved the greatest success in both tasks. Thus, this paper is targeting to explore the performance of the deep learning methods in segmentation and classification tasks in more detail and are expected to obtain excellent results.

3. METHODOLOGY

In this experiment, several deep learning models will be proposed and implemented. The models built from scratch will compare with other pre-trained CNN networks to determine and obtain the optimum models to perform flower image classification and segmentation.

3.1. Classification

The flower dataset for the classification task is unstructured and mixed which is unable to perform classification directly. Hence, all images will be split into different folders with corresponding labels which are bluebell, buttercup, coltsfoot, cowslip, crocus, daffodil, daisy, dandelion, fritillary, iris, lily Valley, pansy, snowdrop, sunflower, tiger lily, tulip, and windflower. After that, these images will be loaded and resized to the vector [256 256 3] which is the optimal size for deep learning to ensure consistency and the

same computational time for each image during training [6]. However, the resize vector might be different for other pre-trained models used which will be illustrated later. After that, the data will be split into training, validation, and testing sets with a 60:20:20 ratio randomly.

Then, the training dataset is applied with data augmentation. Data Augmentation is a technique applied to increase the size of the training dataset to obtain various transformations of the original data. This technique will improve the robustness of the CNN model and prevent overfitting. The training data will be randomly flipped, scaled, shift along the x-axis and y-axis and rotated randomly. By applying these data augmentations, the training dataset will be expanded with various variations which lead to improved generalization and performance of the CNN model on unseen data.

Three different models: ResNet-50, GoogleNet model and a CNN model built from scratch, are implemented to perform classification on flower images. ResNet-50 is a CNN-based model introduced by He et al. It consists of 48 convolutional layers with a Max Pooling Layer and an Average Pooling layer. ResNet architecture allowed CNN to operate with multiple layers. Deep neural networks with numerous stacked layers often produced more significant training error rates than models with fewer layers. This residual network framework allowed the addition of shortcut connections and usage of residual function, allowing stacked layers of deep neural networks to minimize training errors. The direct connection within the network architecture also makes the computation calculation lighter and has better performance in training the network [7].

On the other side, GoogleNet is introduced by Google at the 2014 ILSVRC14 and performs well in image classification. It is a type of convolution neural network based on the inception design to efficiently employ memory and computational resources. The inception module uses several filters with various sizes and then concatenates the outputs which enables the network to capture both local and global features. Additionally, it incorporates framework structures from AlexNet and LeNet with certain adjustments in network width and depth which consists of 22 layers, including 9 inception modules. It also contains global average pooling to average the activations of each feature map and a SoftMax layer to produce class probabilities. GoogleNet includes a 1x1 convolution layer that acts as a bottleneck for rectified linear activation and dimension reduction. This design is to address the issue of increasing computational complexity when the layer becomes deeper. The parallel structure constructed also greatly shortens the training cycle [8].

In this experiment, the ResNet-50 and GoogleNet models are utilized to retrain with the flower dataset by transfer learning. In transfer learning, the pre-trained weight in the model will be transferred to other network models for further action. This approach will help to reduce the training time and improve the accuracy of the pre-trained weight from millions of images [9]. Therefore, to fit the image data to ResNet-50 for classification, the image data will be preprocessed by resizing it to a uniform size of [224 224 3]. Then, the ResNet-50 model will be loaded from the pre-trained model library. The last layer of the model which contains the output class probabilities will be replaced to match the number of classes for the target task which is 17 classes. The modified Resnet-50 model will be trained by using back propagation algorithms.

For the GoogleNet model, the implementation will be the same as the ResNet-50 model. The training dataset will be resized to [224 224 3] which matches the input layer structure of GoogleNet. After that, the GoogleNet model will be loaded, and the final learnable and classification layer will be replaced with a new fully connected layer with the appropriate number of outputs which is 17 classes.

The CNN model built from scratch is designed based on the traditional deep neural network structure. This model architecture consists of several layers which used for feature extraction and detection. In the first layer, an image input layer with size [256 256 3] is created to match the input size of the training dataset. After that, there are overall 10 convolutional layers, 10 batch normalization layers and 10 ReLU activation layers. Each convolutional layer uses the same fitter size of 3x3, and the number of filters increases respectively from 32 to 512 per two layers. Between each convolution layer, there is a batch normalization layer and a ReLU activation layer. After every two-convolution layer, there is a max pooling layer with a pooling size of 2x2 with a stride of 2. This pooling function will help to reduce the spatial dimensionality of the output from the convolutional layer and aid in extracting more generalized features. Finally, there will be a fully connected layer and a softmax output layer with a classification layer in the last layer of the network. The output of this architecture will be a probability distribution of the input image belonging to each of the output classes.

3.2. Segmentation

Unlike classification tasks, the dataset provided for segmentation is only focusing on 1-class, daffodil flower which does not require reconstructing the files in the folder. On the other hand, the segmentation ground truth of the images is provided and exists as colour maps where the flower is labelled as “1” and the background is labelled as “3”. Hence, the pixel labels are created for each image by using the annotations given with appropriate labelling and class names. Moreover, the dataset for segmentation will be split in as same as a classification task in a ratio of 60:20:20. The data augmentation applied in the segmentation will

same as the one in classification tasks which be randomly flipped, scaled, and shift along the x-axis and y-axis and rotated the training dataset randomly to increase the variations of images.

Two CNN models are implemented for image segmentation which are ResNet-50 and a CNN model that build from scratch. The scratch CNN model is customized from classical U-net architecture. Firstly, the image layer input is created with the size of [256 256 3] to match the size of the training dataset which resizes in the data pre-processing section. Besides, the model consists of down-sampling (encoder) and up-sampling (decoder) layers which reference. The down-sampling layers are defined using a convolutional layer with a filter size of 3 and 64 filters, rectified linear unit (ReLU) for the activation layer and max pooling layers with a pool size of 2 and stride of 2 for down-sampling the image. These layers will make the data more manageable size which could reduce the dimensionality of the data and enable faster processing. Furthermore, the up-sampling layers consist similar structure to the down-sampling layer but use transposed convolution layers. It contains 64 filters with a size of 3 and uses ReLU for activation layers. The up-sampling network is used to increase the spatial resolution of the feature maps back to the original image size, refine the pixel-wise classification produced by the network, and produce a segmentation mask that accurately identifies the objects in the input image. Finally, the output layer is defined with a 1x1 convolutional layer with SoftMax and pixel classification layers.

Moreover, a ResNet50 network architecture will be utilized as the base CNN model for image segmentation. However, the ResNet50 model is designed for classification tasks. Thus, the DeepLabv3 network will be used to implement the ResNet-50 model for segmentation. DeepLabv3+ is a convolutional neural network which utilizes Atrous Convolutions in conjunction with spatial pyramid pooling to expand the field of view of filters to accommodate a broader environment and adjust the resolution of features extraction [10]. DeepLabv3 will be initialized with the ResNet-50 base model and act as the backbone. The input size, [256 256 3] and the number of classes, 17 are also passed as arguments to act as the input layer in this network. Moreover, to adapt the ResNet50 model for pixel-wise classification, the last layer for pixel classification in the original network will be replaced by the new layer that assigns the class labels which are flower and background so that the model will only focus on these two main classes.

4. EXPERIMENT

4.1. Dataset

The dataset of Oxford 17 flowers and ground truth labels are provided to perform image classification and segmentation. This dataset is a collection of 17 categories of flowers which consists of 1360 images in total where 80 images for each class. The images are varied in attitude, scale and light differences and there are classes with a wide range of images within the class and close similarities to other classes. This dataset will be used to perform classification tasks. On the other hand, a subset of images with groundtruth labelled are given to perform semantic segmentation task [11].

4.2. Hyperparameter Settings

Table 1: Classification & Segmentation Hyperparameter Settings

Training Options	Classification	Segmentation
Epoch	30	50
Learning rate	1e-4	1e-4
Mini-Batch Size	64	64
Shuffle	Every epoch	Every epoch
L2Regularization	1e-4	1e-4
Optimizer	Adam	Adam

These hyperparameter settings are applied to both classification and segmentation training. The model will pass through the training dataset over 30 times for classification and 50 times for segmentation. The learning rate is set as 1e-4 which is optimum for training the model to avoid overfitting and underfitting [12]. The mini-batch size is set to 64 which means the model will update its weight after processing 64 images per time. This will help in faster convergence and better generalization of the model. The order of images in each epoch will also be randomly shuffled to prevent model bias by the order of the training images. L2 regularization is set to 1e-4 to add a penalty term to the loss function by reducing the magnitudes of the weight parameter. Lastly, the Adam optimizer is used as it is a stochastic gradient descent optimization algorithm that has been shown to work well for deep learning models [13].

4.3. Evaluation Metrics

The performance of the models will be evaluated by using a variety of methods with validation and testing datasets. Classification accuracy of validation and testing dataset and F1 score will be calculated to evaluate the performance of models in classification tasks. For segmentation, mean IOU, segmentation accuracy of test set and validation set and F1 score will be calculated. The experiments for classification and segmentation will be conducted on MacBook M1 Pro.

5. EVALUATION

5.1. Classification & Segmentation Results

Table 2: Flower Classification Results

Model	Validation accuracy (30 epochs)	Testing accuracy	F1-score	Time Taken (minutes)
Scratch	55.88%	65.44%	0.67	105.95

Model				
ResNet50	94.12%	96.32%	0.97	94.56
GoogleNet	86.76	90.81%	0.92	30.47

Table 3: Flower Segmentation Results

Model	Validation accuracy (50 epochs)	Testing accuracy	Mean IOU	Boundary F1 (BF)	Time Taken (minutes)
Scratch Model	97.35%	97.21%	94.55%	0.79	6.3
ResNet50	95.07%	96.73%	91.55%	0.84	26.21

Tables 2 and 3 show the in both classification and segmentation tasks. In Table 3, the ResNet-50 model outstanding the other models by obtaining the best classification accuracy score on validation, testing and F1-score and training in a shorter time. GoogleNet model comes in the second and the scratch CNN model performs the worst to classify the flower images accurately. For segmentation tasks, the scratch CNN model outperforms the ResNet-50 model by achieving the highest accuracy in validation, testing and mean IOU and completing the training in a short time. However, the boundary F1 score of ResNet-50 is higher than the scratch model.

5.2. Discussion

In terms of image classification, the ResNet50 model obtains the best result to classify flowers. This can be attributed to its deeper architecture, which allows it to learn more complex features and patterns from the dataset. The network architecture with "skip connections" allow information from earlier layers directly propagated to deeper layers and bypassing the intermediate layers. This allows the network to learn the underlying features of an image more effectively and can help to mitigate the problem of vanishing gradients in deep networks. Moreover, ResNet contains features of heavy batch normalization which help to address the problem of internal covariate shift that could slow down training and make the network difficult to converge. These techniques allow ResNet50 to outperform the other models. ResNet50 is also pre-trained with a large dataset, which could help it generalize well to other image classification tasks.

In contrast, GoogleNet performed moderately well. The complex "Inception" architecture that involves multiple parallel convolutional layers of different sizes allows for efficient use of computational resources that can help to prevent overfitting. Although this technique might not be effective as ResNet at learning underlying features, the parallel network architecture and the regularization techniques allow the GoogleNet model to train fast than ResNet50 while still achieving a good result. On the other hand, the scratch CNN model performs the worst in every evaluation result. It is because the scratch model is a simple convolutional neural network without any pre-trained weight. The large depth layers consisting in the scratch model also increase the complexity of the network which increases learning rate error during training.

For the flower segmentation task, although both the scratch CNN model and ResNet50 model have similar performance, the scratch CNN model outperforms the ResNet50 to achieve better validation and testing accuracy, mean IOU and significantly take lesser time to train the network. This is likely because the scratch model's skip connections help preserve the spatial information lost during down-sampling. The U-net architecture also can learn with very few labelled images that are suited for image segmentation which means the model is adapted to the small dataset given. Besides, the scratch U-Net model is specifically designed for segmentation tasks but ResNet-50 is a more general-purpose network architecture that takes a long time to adapt to the specific characteristics of a small training dataset. However, ResNet50 models obtain better F1-score compared to the scratch model which indicated that it may perform better in certain situations. Overall, both models show effectiveness in the task of image segmentation.

6. CONCLUSION

In conclusion, several deep learning-based methods have successfully proposed to classify and segment flower images. Results analysis revealed and proved the use of CNN allows a more robust classifier and segmentation. The transferred weights from the pre-trained model allow faster convergence and a more accurate solution when optimizing the weights by retraining the models in both classification and segmentation tasks. However, the scratch CNN model outperforms ResNet-50 in all metrics due to the U-net architecture which is more effective in segmentation tasks. Additionally, ResNet-50 shows the ability to adapt to both classification and segmentation tasks due to its robust architecture although requires more computational time. Overall, this research demonstrates the potential of deep learning techniques in flower image classification and segmentation tasks and highlights the importance of selecting appropriate architecture to achieve optimal performance.

For future work, the dataset for classification could be expanded with more diverse flower images such as 102 categorical flower datasets provided by Oxford to improve the generalization of the models. Moreover, other deep learning techniques such as instance segmentation and object detection would also be considered for more details analysis of flower images for classification and segmentation. Different pre-processing techniques such as geometric transformation, and image filtering will be considered to apply to improve the quality of the image to obtain high accuracy. Further research in these fields should be continuing to obtain 100% accuracy for both tasks.

8. REFERENCES

[1] Y. Liu, F. Tang, D. Zhou, Y. Meng, and W. Dong, "Flower classification via Convolutional Neural Network," *2016 IEEE International Conference on Functional-Structural Plant Growth Modeling, Simulation, Visualization and Applications (FSPMA)*, pp. 110–116, 2016. doi:10.1109/fspma.2016.7818296

[2] H. Hiary, H. Saadeh, M. Saadeh, and M. Yaqub, "Flower classification using deep convolutional Neural Networks," *IET Computer Vision*, vol. 12, no. 6, pp. 855–862, May 2018. doi:10.1049/iet-cvi.2017.0155

[3] M.-E. Nilsback and A. Zisserman, "Automated Flower classification over a large number of classes," *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729, 2008. doi:10.1109/icvgip.2008.47

[4] T.-H. Hsu, C.-H. Lee, and L.-H. Chen, "An interactive flower image recognition system," *Multimedia Tools and Applications*, vol. 53, no. 1, pp. 53–73, May 2010. doi:10.1007/s11042-010-0490-6

[5] C. F. Baumgartner *et al.*, "Phiseg: Capturing uncertainty in medical image segmentation," *Lecture Notes in Computer Science*, pp. 119–127, 2019. doi:10.1007/978-3-030-32245-8_14

[6] O. Rukundo, "Effects of image size on Deep Learning," *Electronics*, vol. 12, no. 4, p. 985, Feb. 2023. doi:10.3390/electronics12040985

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. doi:10.1109/cvpr.2016.90

[8] C. Szegedy *et al.*, "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. doi:10.1109/cvpr.2015.7298594

[9] S. J. Pan and Q. Yang, "A survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010. doi:10.1109/tkde.2009.191

[10] A. Wagh *et al.*, "Semantic segmentation of smartphone wound images: Comparative analysis of AHRF and CNN-based approaches," *IEEE Access*, vol. 8, pp. 181590–181604, Aug. 2020. doi:10.1109/access.2020.3014175

[11] M.-E. Nilsback and A. Zisserman, "17 category Flower Dataset," Visual Geometry Group - University of Oxford, https://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html.

[12] L. Fan, T. Zhang, X. Zhao, H. Wang, and M. Zheng, "Deep Topology Network: A Framework based on feedback adjustment learning rate for Image Classification," *Advanced Engineering Informatics*, vol. 42, Oct. 2019. doi:10.1016/j.aei.2019.100935

[13] S. J. Reddi, S. Kale, and S. Kumar, "On the Convergence of Adam and Beyond," arXiv.org, https://arxiv.org/abs/1904.09237.