A red blue and black flag

Description automatically generated

**Flower Image Type Classification Using Deep Learning Models**

**I.Introduction**

The image-based classification of flowers is an important deep learning use case aiding agriculture and horticulture development and environmental conservation efforts and educational progress. The auto-identification of flowers allows researchers to recognize species plants and enables thorough biodiversity monitoring and offers explanatory learning materials for information seekers ranging from pupils to hobbyists. The use of computer vision has become an important element for such innovations due to the industry requiring real-time accurate and scalable classification platforms.Convolutional Neural Networks (CNNs) greatly improved image recognition capabilities over the last decade. Researchers have embraced AlexNet and also VGG16 and ResNet and MobileNet since they extract visual data features automatically in a spatial and semantic way for classification problems [3]. Such type of architecture facilitated innovation towards DenseNet and RegNet and EfficientNet to have optimal performance levels through lowering parameter size.Researchers have been able to apply transformer models derived from methodologies in natural language processing to perform image classification functions. Three transformer models such as DEIT, VIT and Swin Transformer, LeVIT have surpassed traditional CNNs by virtue of self-attention models capable of retaining entire image contexts in various benchmarks [1][2]. The design of architectural space has improved significantly with this shift since transformers allow global feature fusion which outscopes localized receptive field approaches. The study entails experimenting and examining ten deep learning models that consist of both convolutional models and transformer models to trace flower images. The objective of the study is to evaluate performance levels when examining both conventional CNN-based networks and novel transformer-based approaches in visual recognition systems whose detailed categorization is essential.

**II. Dataset**

The data is sourced from the 5 Flower Types Classification Dataset on Kaggle and is utilized in this work. This dataset has numerous pictures of flowers belonging to the five classes: Lilly, Lotus, Orchid, Sunflower and Tulip. Various lighting and real-world conditions like gardens and open fields and interior places supply the resolution-varied images in this dataset. The dataset poses a great challenge because it is heterogeneous and compels deep models to deal with real-world hurdles like flower-type generalization to environmental factors and visual imprecision. The data set is without a preliminary segmentation of its training and validation and test data sets. For our model evaluations we manually segmented 80% of the data for training and set aside 20% to be used for validation. Validation results were included in model performance evaluations to measure final model performance since an independent test set was excluded in the process.

**III. Models Used**

1. AlexNet is a cornerstone of deep learning due to it being a pioneering convolutional neural network (CNN) for image vision. The network consists of a cascade of five layers of convolutions followed by three fully connected layers with ReLU activation and dropout regularizers for regularization. We have adapted AlexNet's last classification layer to fit into flower classes and supplemented it with dropout of a rate equal to 0.6 to avoid overfitting. Simple nature of AlexNet coupled with its established success in image classifications renders it a suitable benchmarking standard even though it is old [3].
2. MobileNetV2 model is a compact and power-saving CNN design specifically designed for mobile vision and embedded systems. The model retains accuracy through depthwise separable convolutions in collaboration with inverted residual blocks with linear bottlenecks to reduce computations. MobileNetV2 is very efficient and thus is ideal for real-time processing on systems of low resources. The MobileNet involved certain image transformations like 224×224 resizing and also a horizontal flipping and normalization process in our implementation [3].
3. Reg-Net models are a group of convolutional networks generated via neural architecture search to obtain an optimum balance between operational efficiency and quality of performance. The RegNet\_Y\_400MF model is one of the smaller members in a network by utilizing bottleneck blocks and group convolutions in a straightforwardly scalable network model. The model has superior performance in limited computational environments to attain similar accuracy levels such that it is an apt option for real-world mobile plant recognition applications [4].
4. VGG-16 model is The deep convolutional neural network utilizes sets of 3×3 filter convolutions filterd with a uniform 16 weighting layer pattern. The network has repeated convolutions with the ReLU activations and max pooling until it is followed by the 2 fully connected layers with the softmax classifier. Our strategy involved initializing a pre-trained vgg16 and replacing it with a customized prediction head featuring Linear(25088→512) followed by ReLU activation and then 0.4- dropdown before Linear(512→5). The model facilitated full trained by making all the layers need their gradients. Training used Adam as an optimizer while training at a learning rate of 1e-4. Data augmentation comprised image augmentation RandomResizedCrop(224) and RandomHorizontalFlip, RandomRotation and ColorJitter. Training utilized a ReduceLROnPlateau scheduler and early stopping with a patience of 3 epochs throughout learning. The model performed well because it achieved a validation accuracy of about 96.7% when it converged fast when training. [3][6].
5. DenseNet121 is model that puts the image through numerous layers, and each learns from what it knew previously and what it learns thereafter is shared with all the succeeding layers. The model also retains critical information and doesn't go through redundant work. DenseNet121 is then fast in performance, less memory intensive, and performs well in animal, flower, or medical scan image classification due to this sharing. To classify images, I have constructed a data density of a DenseNet121 deep learning model. The data preprocessing used was image transformations such as resizing and normalizing. For assistance to the model to learn and prevent overfitting, dropout layers have been used. For model training, I utilized Adam optimizer with a fixed learning rate and cross-entropy loss to monitor how well the model is performing. Also, early-stopping was used to prevent overtraining when training is stopped when the model is not improving for a while. This configuration helps to learn efficiently and generalize well to test data.[3][5]
6. ResNet-18 is a residual network that The network uses shortcut connections to prevent gradient values from disappearing during deep learning operations. Through residual blocks the input gets connected to successive layers of stacked blocks that enables deep networks with normal gradient flow during training. The procedure for our case involved reprogrammed ResNet-18 through nn.Linear(f\_in, 5) which modified its final fully-connected layer against the number of input features f\_in. The training utilized Adam optimization with 1e-3 learning rate and added StepLR rate adjustment (gamma=0.5, step\_size=3) during the training process. The preprocessing steps involved turning images into 224×224 size while executing RandomHorizontalFlip. The training lasted 15 epochs and validation accuracy reached its maximum of 95.8% during this time period. LeViT-384 demonstrated superior overfitting results compared to the faster but less efficient model although it reached convergence quickly. [6].
7. The EfficientNetV2 image classification model is incredibly powerful because it is able to classify in seconds and with accuracy the objects in an image. It’s so much smarter than a camera brain because it has learned: a cat, or a bouquet of flowers, or a vehicle. EfficientNetV2 is unique in that it learns so much faster and with less computational resources than older models. It is intelligent, and it is also very efficient. It is utilized extensively in apps and tools that must recognize pictures in a flash and accurately. I used EfficientNetV2 model which is a very strong image classifier. Some of the images were also prepared and transformed and to enable the model to learn optimally, some data was applied to it. Later I assisted in training the model in a better manner by incorporating weight decay to reduce overfitting and also used the Adam optimizer and used a specific learning rate. Besides this, CrossEntropyLoss was also used to enable the model to know how much it was deviating. I have also introduced dropout layers in the model in which some neurons are disabled randomly in training to enable generalizability of the model. Finally, early stopping was used to terminate training when it saturates and to avoid training for a long time and losing accuracy on new data.[4]
8. Data efficient image transformer is a type of model that can view images and recognize what is in them. The unique thing about this model is that it learns to do this without requiring large numbers of images like other models. It is trained through intelligent approaches like “knowledge distillation” where it learns from an effectively trained mannequin (think of a teacher teaching a pupil). This allows DeIT to train much faster with fewer data. I built a model capable of image classification with the DeiT structure. Images were transformed into training prior to passing them down to a model with resizing, normalization etc. For learning, I utilized Adam optimizer that adjusts learning in a smart manner and introduced weight decay so that models don’t tend to memorize training data by a lot (oversorting). The loss function utilized was cross-entropy loss function, which is standard for classification problems. I utilized a dropout to model also that randomly disables some neurons that occur in training to make model resilient. I utilized early stopping, meaning when model is not anymore enhancing on validation data, training was automatically stopped. Such decisions enabled learning of a model in an efficient manner and making it perform well on new unseen images.[1]
9. Vision Transformer model that corpus and have knowledge about how to classify based on the image. Rather than an image, a ViT consumes a sequence of patches, converts them into sequences (similar to a sentence), and a transformer (that was used originally for language) to comprehend how to put things together. I make the images slightly less complex for the model to process by transforming beforehand such as resizing, normalization. For training the model well and generalizing well to new images, I train the model with the Adam optimizer taking responsibility for making necessary adjustments to model’s weights. Additionally, I also establish a learning rate and counter overfitting by incorporating weight decay which is discouraging the model to rely on a certain pattern. It also has a dropout layer to randomize what part of the model becomes stronger. CrossEntropyLoss is how incorrect the model’s prediction is, and how to direct learning. I also employ early stopping to terminate training when model improves less so that my model will not overfit. Individually, these elements help in a ViT model learning necessary information about images and then classifying accurately.[1]
10. The Swin Transformer model integrates hierarchical representation along with shifted windows in a way to obtain reduced levels of computer complexity. The self-attention mechanism allows it to find long range relationships in data and meanwhile retains superior performance against complex image recognition tasks. The flower classification task used the Swin Transformer as its new benchmark since it is superior in processing global and local features simulaneously [2].
11. LeViT-384 is a lightweight Vision Transformer Researchers developed this model to combine the beneficial features between both convolutional networks and attention mechanisms. LeViT integrates convolutional downsampling steps with the pooling between transformer blocks that give the model higher speed at the inference time and aslo the more efficient parameters. The network applies batch normalization instead of LayerNorm while using the HardSwish activations together with the attention blocks on the patch embeddings. To initiate the implementation we invoked the timm. create\_model("levit\_384", pretrained=True, num\_classes=5) function to establish the pretrained model with the 5 head output parameters. AdamW is selected as the optimizer during the training, which used a learning rate = 2e-4 until the early stopping with the patience value set to 3 epochs occurred. Because the preprocessing included only dimension resize to be 224 and also normalizing steps. Across all models the learning process isinitially slow but the model displayed the peak generalization at epoch 8 when is is reached 97.4% validation accuracy so showcasing the top performance for flower classification[1][2].

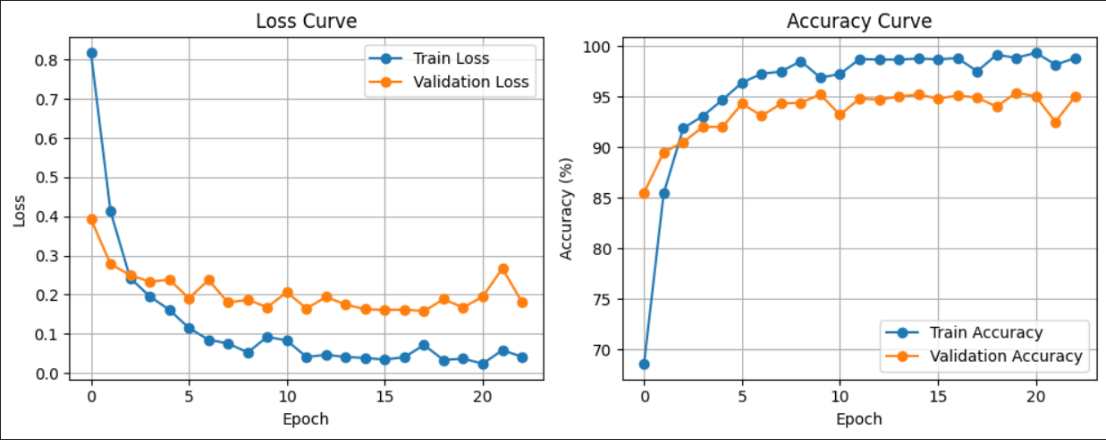
**IV. Implementation**

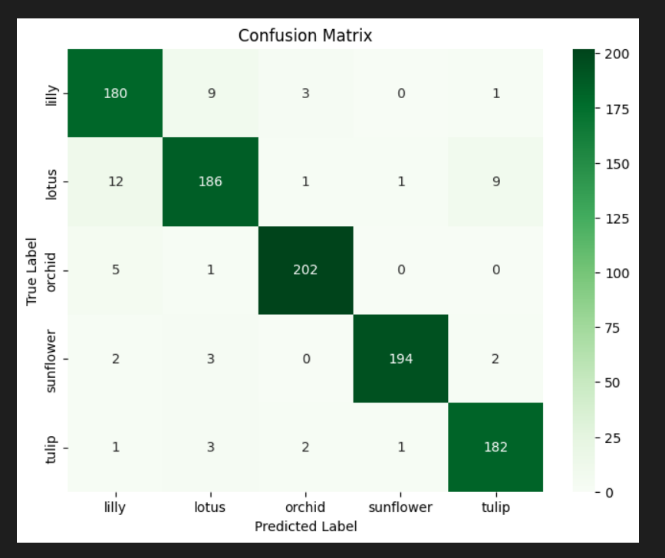
The models were provided with images resized to 224×224 pixels in size for uniform input dimensions. The preprocessing pipeline also improved image quality through image sharpness and contrast and brightness adjustments even though routine. All the enhanced images are then placed in one directory to be ready to be used. The data normalization was carried out utilizing Different statistical operations for each model were employed to prepare the data to be compatible in a pretrained model environment. The model training utilized Different batch sizes for each model on the dataset.

**V. Comparison**

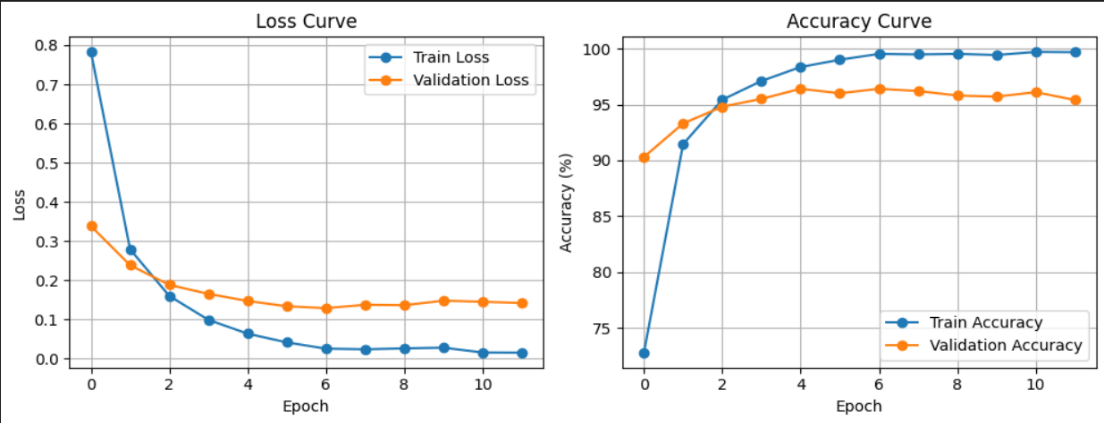
|  |  |
| --- | --- |
| Model | Validation Accuracy |
| AlexNet | 95.40% |
| MobileNetV2 | 96.40% |
| RegNet | 96.60% |
| VGG16 | 96.70% |
| DenseNet121 | 96.50% |
| ResNet18 | 96.10% |
| EfficientNetV2 | 97.30% |
| DEIT | 98.80% |
| Vision Transformer | 98.20% |
| Swin Transformer | 97.10% |
| LeVIT | 97.04% |

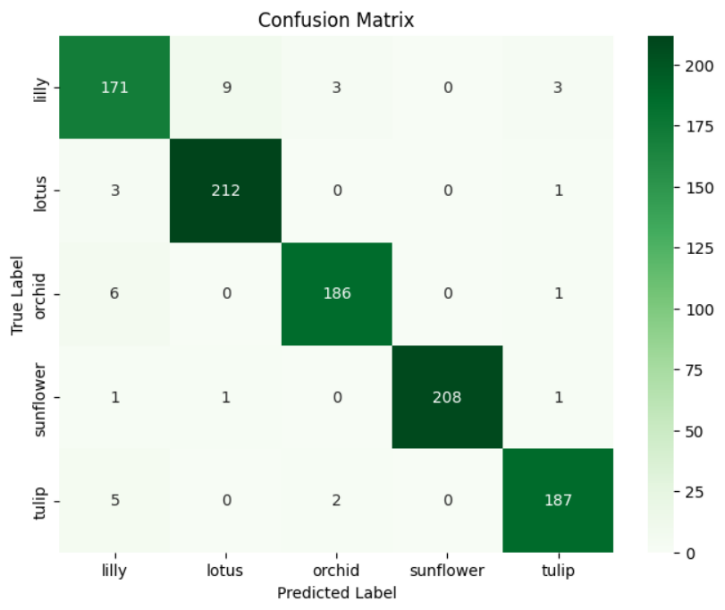
AlexNet





MobileNetV2





RegNet

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

A diagram of a number of flowers

AI-generated content may be incorrect.

VGG16

A graph with a line graph

AI-generated content may be incorrect.A graph with lines and numbers

AI-generated content may be incorrect.A diagram of different plants

AI-generated content may be incorrect.

DenseNet121

A graph of a number of different colored lines

AI-generated content may be incorrect.A diagram of a number of flowers

AI-generated content may be incorrect.

ResNet18:

A graph with lines and numbers

AI-generated content may be incorrect. A graph with a line graph

AI-generated content may be incorrect.

A diagram of different plants

AI-generated content may be incorrect.

EfficientNetV2

A comparison of a graph

AI-generated content may be incorrect.A graph of numbers and a diagram

AI-generated content may be incorrect.

DEIT

A screenshot of a graph

AI-generated content may be incorrect.A graph of a number of different colored squares

AI-generated content may be incorrect.

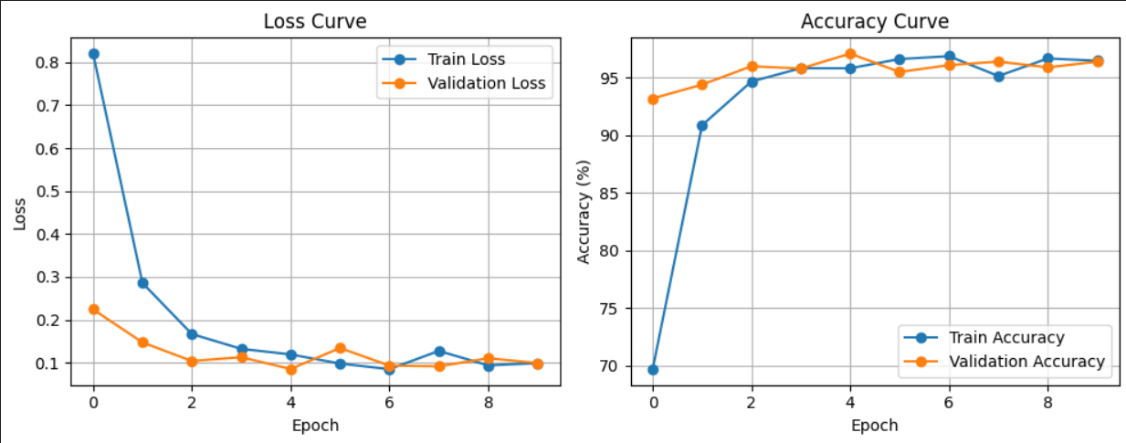
Vision Transformer

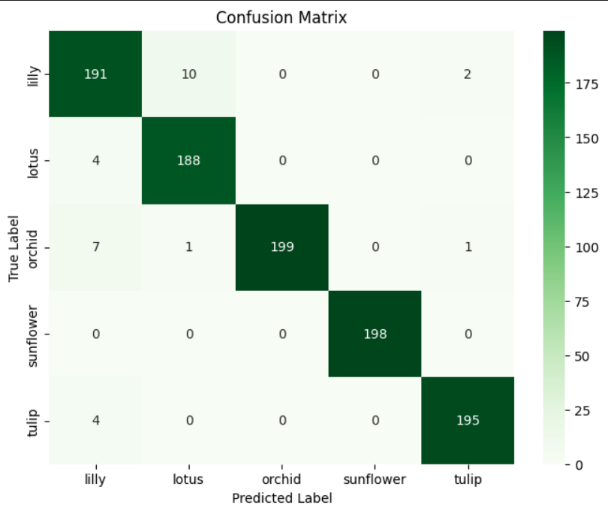
A graph of different colored lines

AI-generated content may be incorrect.A green squares with white text

AI-generated content may be incorrect.

Swin Transformer





LeVIT:

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

A diagram of different plants

AI-generated content may be incorrect.

**VI. Conclusion:**

The various deep learning network models range from traditional convolutional neural networks to today's transformer models to evaluate flower image classification performance. The different models were evaluated through validation accuracy testing with standardized experimental parameters and dataset across models.

The validation accuracy was achieved by EfficientNetV2 to an extent of 97.30%, VGG16 to an extent of 96.70%, RegNet to an extent of 96.60%, and DenseNet121 to an extent of 96.50% in the CNN-based models. The study indicates that well-arranged efficient convolutionary frameworks have a good performance level. DEIT was the most accurate model following its transformer-based design implementation since it attained a very good accuracy rate of 98.80% whereas Vision Transformer achieved a rate of accuracy of 98.20% and Swin Transformer to a rate of accuracy of 97.10%. LeViT attained good results of 97.04% despite it being a hybrid model.

The study establishes that transformers are a novel solution to image classification in which global attention and feature learning require higher quality in class discrimination. The transformer model yields superior performance and wider usability in intricate visual issues compared to what CNN models have to offer. Models based on transformers are now the state-of-the-art method for fine-grained classification of flowers.

**VII. References:**

[1] Ermis, B., Zappella, G., Wistuba, M., Rawal, A., & Archambeau, C. (2022). Continual learning with transformers for image classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 3774-3781).

[2] Dihin, R. A., Al Shemmary, E. N., & Al-Jawher, W. A. M. (2023). Implementation Of The Swin Transformer and Its Application In Image Classification. Journal Port Science Research, 6(4), 318-331.

[3] Narvekar, C., & Rao, M. (2020, December). Flower classification using CNN and transfer learning in CNN-Agriculture Perspective. In 2020 3rd international conference on intelligent sustainable systems (ICISS) (pp. 660-664). IEEE.

[4] Kim, H., & Kim, D. (2023). Deep-Learning-Based Strawberry Leaf Pest Classification for Sustainable Smart Farms. Sustainability, 15(10), 7931.

[5] Tian, M., Chen, H., & Wang, Q. (2019, June). Flower identification based on Deep Learning. In Journal of physics: conference series (Vol. 1237, No. 2, p. 022060). IOP Publishing.

[6] Burhan, I. M., Ali, Q. A., Hussein, I. S., & Jaleel, R. A. (2023). Mobile-computer Vision Model with Deep Learning for Testing Classification and Status of Flowers Images by using IoTs Devices. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, *14*(1), 82-94.