TYLCV Identification in Tomato Plant Images using Transfer Learning and InceptionV3

Amit Damri, Yiftach Savransky, Amir Gabay
Ben Gurion University of the Negev
SISE Dept.
Be'er Sheva, Israel
amitdamr@post.bgu.ac.il, yiftachs@post.bgu.ac.il, gabayam@post.bgu.ac.il

1 Introduction

Tomatoes are a very popular food, consumed by many people in the western society. In 2018, world production of tomatoes was 182 million tonnes[1]. Tomato plants might suffer from different diseases which can cause considerable production and economic losses in the agriculture sector. Thus, there is a need to identify these diseases and trying to prevent them.

Recent advances in computer technology have vastly improved image identification abilities. Through the recent advances in Machine Learning, the performance of systems aiming to detect or recognize an object in images have been widely improving. Most notable image identification methods are deep Convolutional Neural Network (CNN) models - which can help in the task of detecting unhealthy plants in images.

One of the most devastating viral diseases of cultivated tomato plants is Tomato Yellow Leaf Curl Virus (TYLCV). This disease decreases the yield of up to 100%. In many regions, TYLCV is the main limiting factor in tomato production. Plants that are infected display common characteristics such as upward curling of leaves and yellowing of young leaves (Moriones and Navas-Castillo 2000). These characteristics can be identified by experts and treated.

In this paper We trained a deep CNN model to identify ill tomato plants using a relatively small dataset of 1000 images labeled by experts. To overcome the limited data challenge, we utilize a pre-trained model and transfer its learned knowledge to our task. This method is referred to as Transfer Learning. We used a model that is based on an Inception(Szegedy et al. 2016) that was pre-trained on the ImageNet dataset (Deng et al. 2009). The InceptionV3 model was adapted and fine tuned to our image classification task.

We compared the performances of our model to ResNet50-based model (He et al. 2016) on the dataset. We also conducted a user study to establish the human level performance and to compare it to our model. Our model outperformed the other model and the user study results.

2 Background

Image Classification is a supervised learning task that attempts to comprehend an entire image. The goal is to classify the image by assigning it to a specific label. Typically, Image Classification refers to images in which only one object appears and is analyzed. One of the most popular models to preform image classification is a variant of Artificial Neural Networks (ANN) called Convolutional Neural Network (CNN).

An ANN has thousands of artificial neurons called processing units, which are interconnected by nodes. These processing units are made up of input units, output units and activation functions. Activation functions are used on the output of neural network units. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function). One example of an activation function is the Sigmoid function. This function is monotonically increasing and maps any real value to the range (0, 1), so that it can be interpreted as a probability. The input units receive various forms and structures of information based on an internal weighting system, and the neural network attempts to learn about the information presented to produce one output report. ANNs use a set of learning rules such as backpropagation, to perfect their output results. An ANN initially goes through a training phase where it learns to recognize patterns in data, whether visually, aurally, or textually.

Deep CNNs are a special type of ANNs, which has shown exemplary performance on several competitions related to Computer Vision and Image Processing. CNN takes its name from mathematical linear operation between matrices called convolution. A convolution extracts windows of the input feature map, and applies filters to them to compute new features, producing an output feature map (which may have a different size and depth than the input feature map). During a convolution, the filters effectively slide over the input feature map's grid horizontally and vertically, one pixel at a time, extracting each corresponding window. CNN have multiple layers; including convolutional layer, non-linearity layer (activation layer), pooling layer and fully connected layer (a layer in which every unit is connected to every unit in the previous layer). The convolutional and fully connected layers have parameters, but pooling and non-linearity layers do not have parameters.

Traditionally, training of CNN models for image clas-

[1] http://www.fao.org/faostat/en/#data/QC

sification requires learning millions of parameters and requires a very large number of annotated image samples. This requirement prevents the application of CNNs to tasks with limited training data. Nowadays, there exist many pretrained CNNs which were trained on large scale data for different tasks. There exists a method of utilizing the parameters that were learned by those models to increase the performance of CNN models on classification tasks with limited labeled samples.

This method of using pre-trained models is referred to as transfer learning. Transfer learning aims to transfer knowledge between related source and target domains. The general steps to perform transfer learning are as follows (Oquab et al. 2014): First, the network is trained on the source task with a large amount of available labelled images. The trained parameters are then transferred to the target task, by replacing the output layer of the trained model. The parameters that were learned in the training process for the source task are kept fixed and only the adaptation layers are trained on the target task training data.

The transfer learning application requires a trained model. Deep CNN models were relatively successful in performing classification tasks on the ImageNet dataset (ILSVRC) as shown in (Krizhevsky, Sutskever, and Hinton 2012). The ImageNet dataset (Deng et al. 2009) consists of about 1.2 million images for training, 50,000 for validation and 100,000 images for testing. Each image is associated with one ground truth category (out of 1000). There are various widely known pre-trained deep CNN models that are publicly available such as InceptionV3 and ResNet50 that have been proven to perform very well in classification of the ImageNet dataset.

The Inception model (GoogLeNet architecture) suggested by (Szegedy et al. 2015), introduced a new architecture and techniques that increased the performance on the ImageNet classification task. This model utilizes a new type of network-in-network blocks as a part of the network's architecture to increase the representational power of neural networks. A subsequent improvement to the Inception model was represented in (Szegedy et al. 2016) with the InceptionV3 model. This model used a new architecture which includes convolutional layers and Inception blocks that are different than the original GoogLeNet architecture. This model is much cheaper computationally and outperforms the results reported by (Szegedy et al. 2015).

Another widely used model is ResNet (He et al. 2016). This model was designed to overcome the difficultly of training meaningful features in deep neural networks. The model enables stacked layers to use "shortcut connections" (skipping one or more layers) by adding the output of a layer to the input of a deeper layer. These shortcut connections add neither extra parameter nor computational complexity. These residual networks can gain accuracy from considerably increased depth. ResNet50 is an implementation of ResNet architecture with 50 layers.

3 Related Work

Transfer learning methods were implemented in various fields and on a variety of tasks. Using this concept, we will show how we can overcome the deficit of training samples for a specific image classification task by adapting classifiers that were trained for other image classification tasks.

Transfer learning utilization techniques were presented in (Aytar and Zisserman 2011), in which SVM models were transferred from one category to another similar category detection task. The models performed well on the category detection task even though they had limited available samples of the target category. The capabilities of transfer learning were also shown in previous works such as (Donahue et al. 2014).

A transfer learning method for object detection and classification using CNN was suggested in (Oquab et al. 2014). This paper describes a method of reusing CNN layers that were trained on the ImageNet dataset as mid-level image representations that can be transferred to new tasks. Results of experiments on models that were trained using this method show that despite of differences in image statistics and tasks in the two datasets, the transferred representation leads to significantly improved results for object and action classification. On the same classification task, a pre-trained network outpreformed the same network that was not initialized with pre-trained weights by between 8% to 20% (average precision).

Another work of Bharadwaj and Juliet (Reddy and Juliet 2019) uses transfer learning to classify Malaria Cell images. Using transfer learning technique, they were able to use a limited dataset of malaria cell images in order to classify infected and non-infected malaria cells. This improved the diagnostic accuracy of microscopists, especially in limited resources areas. Their method used a pre-trained ResNet50 model and right after the ResNet50 layers they used a sigmoid dense layer. All of the ResNet50 layers were frozen through the learning process, except of a few batch normalization layers that were autotuned. The only layer that is trained based on back propagation is the last fully connected layer. Using the described method, they achieved 95.4% accuracy.

In addition, another transfer learning example was described by Hentschel et al. (Hentschel, Wiradarma, and Sack 2016). They used a dataset of 85,000 paintings manually annotated with 25 genre labels. Since the data provided in the dataset is rather small, when compared to ImageNet, they presented results for a CNN that was pre-trained on the entire ImageNet training data and fine-tuned it using their dataset. In order to adapt the CNN to a new target data, they replaced the last layer with randomly initialized weights for all of the target outputs. They found out that adding more training data helps to increase the classifier accuracy, and only little data is required for the CNN to adapt to the new target task of the painting classification.

In this paper we try to utilize transfer learning techniques on pre-trained image classification CNN models (InceptionV3 and ResNet50) in order to recognize disease in tomato plants from a small, labeled dataset of tomato plant images. The fine-tuning process that we use includes replacing the last layer of the pre-trained CNN models with two fully connected layers with a sigmoid output function. Only the layers that were added are trained in the training process while the others are kept frozen. The output indicates whether a disease was detected in the plant or not.

4 Experiments

We conducted offline and online experiments to test our developed method.

4.1 Dataset

Our dataset includes 1000 images of diseased and healthy tomato plants. Each image was rated by an expert to determine the disease level of the plant. The ratings range is between 0 (highly diseased), to 5 (Healthy). We discretized the data labels by the threshold of 2.5 as follows: ratings above 2.5 were considered as healthy, otherwise they were considered as diseased. In addition, the dataset has additional rating values: 6 – indicates a diseased plants of unknown severity, which were considered as diseased. The rating -1 indicates unknown condition which were removed from the database. With this pre-processing the data contains 462 images of healthy plants, and 373 images of diseased plants.

4.2 Metrics

The cost of missing a diseased plant is equal to the cost of identifying a healthy plant as diseased. The outcome of missing a diseased plant is that the plant dies from the disease without care and we lose X amount of yield. The outcome of identifying a healthy plant as diseased is the removal of that plant and a loss of X amount of yield. Therefore, we seek to minimize the error rate regardless of the error type, in other words maximize the accuracy. We also considered the AUC metric which indicates that the model achieves false positive and true positive rates that are significantly above random chance, which is not guaranteed for accuracy.

4.3 Offline Experiment Process

We conducted an offline experiment to test the following hypothesis:
H_0: *Error rate of InceptionV3-based model = error rate of Resnet50=based model*
H_1: *Error rate of InceptionV3-based model < error rate of Resnet50=based model*

We split the data to 70% train set, and 30% test set. Various image augmentation techniques were used as well as featurewise normalization. We applied random crops, horizontal flips and random rotations of the train set images.

We conducted the experiment in a Python environment using pre-trained InceptionV3 and ResNet50 models from the Keras framework. The last layer of each model was removed and replaced by 2 hidden fully connected layers with an activation function of ReLU. The first added layer is in the size of 2048 and the second hidden layer is in the size of 1024. The last output layer includes one processing unit with an activation function of sigmoid. To train the model we used the binary cross entropy loss function and Adam optimizer (using $\beta_1 = 0.9$, $\beta_2 = 0.999$).

Model	Accuracy	AUC
ResNet50 based	0.69	0.71
InceptionV3 based	**0.785**	**0.782**

Table 1: ResNet50 based and InceptionV3 based models' results on the test set

The hyper parameters that were used, were found using grid search. The best found and used parameters were learning rate of 0.0001, batch size of 32, epochs count of 20. Using the chosen combination of hyper parameters, we trained the models and evaluated the accuracy and AUC scores. The loss metric over the epochs for the InceptionV3 based model is presented in figure 1 and for the ResNet50 based model in figure 2.

Offline Experiment Results The accuracy and AUC metrics of both models on the test set are presented in table 1. The results show that the InceptionV3 based model's accuracy is greater than the ResNet50 based model's accuracy. We conducted McNemar's test to check if there is a statistically significant difference of the proportion of errors between the models. The test's p-value that was obtained is 0.003 therefore, the null hypothesis is rejected with $\alpha = 0.05$. The combination of the InceptionV3 based model's better performances and the rejection of the null hypothesis in the McNemar's test indicates the rejection of our null hypothesis and that the InceptionV3 model has a lower error rate than the ResNet50 based model's error rate.

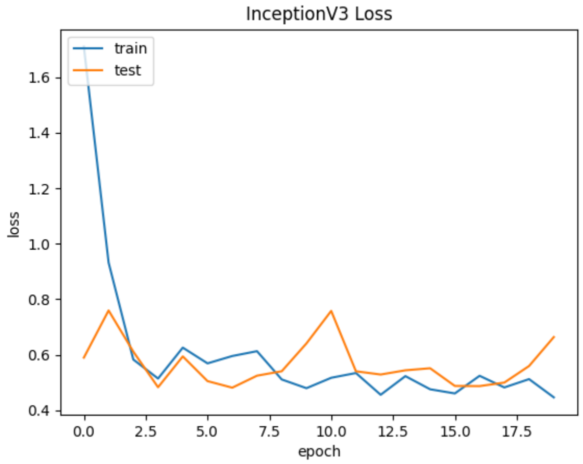

Figure 1: InceptionV3 based model loss metric in the learning phase over epochs

4.4 User Study

We conducted a user study to test the following hypothesis:
H_0: *Error rate of InceptionV3-based model = Human-level error rate*
H_1: *Error rate of InceptionV3-based model < Human-level error rate*

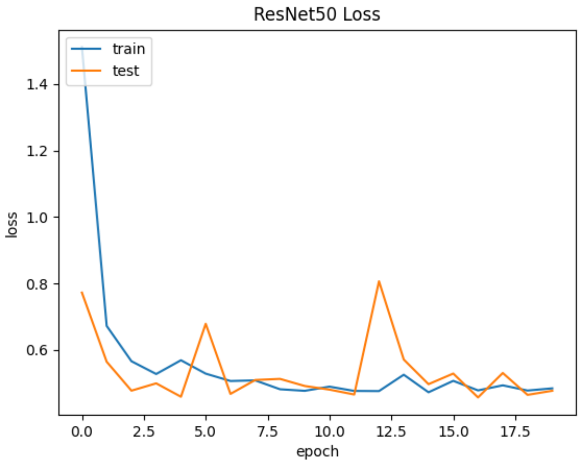

Figure 2: ResNet50 based model loss metric in the learning phase over epochs

The user study included 7 graduate students as participants. Our questionnaire was based on 8 different images per participant, chosen from the offline test set. Every participant was asked to classify whether the presented plant is healthy or not. This way the participants tagged 56 pictures in total that were used as a test set for our model to compare their results.

The questionnaire structure was as follows:

- **Preliminary questionnaire** - Was issued to segment the participants in the experiment. We asked the participants demographically related questions about their age group, their educational level and their level of familiarity with this branch of agriculture.
- **Explanation** - We explained the technical aspects of using the system i.e. how to classify the plant images.
- **Demonstration of the task** - We showed 2 images of ill tomato plants and 2 images of healthy tomato plants so that the participants would be able to conduct the classification task.
- **Experiment Outline** - Each participant was asked to classify 8 images of tomato plants to ill or healthy.
- **Summary** - We asked the users how certain they were when classifying the images and what affected their decision the most.

The preliminary questionnaire shows that 71% of the participants in the study did not have any previous experience, the participants' ages were between 23 and 50 and there were 6 men and 1 woman.

User Study Results The accuracy and AUC metrics of the participants and of our InceptionV3 based model on the user study images set are presented in table 2. The results show that the InceptionV3 based model accuracy is greater than the participants' accuracy. We conducted McNemar's test to check if there is a statistically significant difference of the proportion of errors between the model and the participants.

Model	Accuracy	AUC
User Study	0.64	0.64
InceptionV3 based	**0.8**	**0.81**

Table 2: User study and InceptionV3 based model results on the user study image set

The test's p-value that was obtained is 0.049 therefore, the null hypothesis is rejected with $\alpha = 0.05$. The combination of the model's better performances and the rejection of the null hypothesis in the McNemar's test indicates the rejection of our null hypothesis and that the InceptionV3 model has a lower error rate than the participants' error rate.

5 Conclusions and Future Work

TYLCV is one of the most devastating viral diseases of cultivated tomato plants which causes the loss of significant amount of yield worldwide, therefore There is a need to develop accurate methods to identify infected plants. In this paper we presented a novel method to identify this disease in tomato plant images. Our method use pre-trained InceptionV3 model on the ImageNet dataset that was fine tuned on a dataset of tomato plant images.

Our method was proved to have a statistically significant lower error rate than a ResNet50 based model. To our understanding our model performed better because it reached higher results on the ImageNet classification task, which was the source task of our transfer learning method.

Moreover, our method was also proved to have significantly lower error rate than a human-level classifier through a user study. We assume this is because our model is able to recognize patterns better than this limited ability of human.

Future Work For future work we would recommend to optimize the model's parameters to get better performances. Furthermore, we think that additional deep CNN models should also be tested to find a better performing model. In addition, as a consequence of the good results of our research, we would recommend to apply the suggested method on other plant diseases and domains.

References

Aytar, Y.; and Zisserman, A. 2011. Tabula rasa: Model transfer for object category detection. In *2011 international conference on computer vision*, 2252–2259. IEEE.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; and Darrell, T. 2014. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, 647–655. PMLR.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hentschel, C.; Wiradarma, T. P.; and Sack, H. 2016. Fine tuning CNNS with scarce training data—Adapting ImageNet to art epoch classification. In *2016 IEEE International Conference on Image Processing (ICIP)*, 3693–3697. IEEE.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25: 1097–1105.

Moriones, E.; and Navas-Castillo, J. 2000. Tomato yellow leaf curl virus, an emerging virus complex causing epidemics worldwide. *Virus research* 71(1-2): 123–134.

Oquab, M.; Bottou, L.; Laptev, I.; and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1717–1724.

Reddy, A. S. B.; and Juliet, D. S. 2019. Transfer learning with ResNet-50 for malaria cell-image classification. In *2019 International Conference on Communication and Signal Processing (ICCSP)*, 0945–0949. IEEE.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.