Transfer Learning Using VGG-16 to Detect Crop Diseases

Eldridge, Ethan
CSCI-P556
Indiana University
Bloomington, IN, USA
etmeldr@iu.edu

***Abstract*—Transfer learning is a useful tool for boosting the usability of a small data set for classification. By leveraging learned features from a classifier trained on another data set, a smaller or more difficult data set can be learned with a high degree of accuracy. This paper applies transfer learning to the problem of crop disease classification, specifically in corn. By experimenting with a variety of different initial weights, the VGG-16 classification model was able to accurately and consistently classify Northern Leaf Blight in images of corn crops.**

I. Introduction

Agriculture is a vitally important industry that generates massive amounts of data each year for analysis. Researchers are making use of this data by applying machine learning techniques to maximize crop yields, prescribe fertilizer supplements, and enhance the overall efficiency of the food production process. One interesting application of machine learning in agriculture is proactive disease detection. If plants in one part of the field contract a disease, it can quickly spread to all nearby plants, and eventually all plants in the field. This can be disastrous for farmers, as many of these diseases can quickly damage or completely destroy the crop. Northern Lead Blight, for example, caused by the fungus *Setosphaeria turcica*, has become a major threat to North American corn crops in recent years. In 2015 it was the most economically damaging maize disease in the US and Ontario, causing roughly $2 billion in damage [1]. For this reason, it is important to detect diseases quickly so that they can be dealt with as soon as possible. Effectively detecting disease outbreaks across an entire farm is very difficult to do manually, however, as the average size of a farm in the U.S. is 444 acres (1.8 km^2) and fields can be difficult to maneuver in without damaging the densely planted crop.

To address this problem, agronomists and other agriculture analysts can use a drone or another UAV to fly over a field and take overhead images of the crops using a high-resolution camera. An image classification model used to detect plant diseases is then applied to the flyover footage to rapidly identify outbreaks. Using this method, farmers can detect disease outbreaks much earlier and in much larger areas than they would otherwise be able to do on their own. For my final project, I will be applying what we have learned so far about machine learning to do just this: identify diseases in images of corn crops. More specifically, I will be training the VGG-16 image classification model from scratch using a large data set of various crop leaf diseases. I will then use transfer learning to take the features learned from the larger crop disease data set and apply them to the problem of Northern Leaf Blight classification.

Fig. 1: Images from each class of PlantVillage dataset.

II. Data Sets

There are two main data sets utilized at different stages of this experiment. The data set utilized in Stage One of this experiment, the stage in which VGG-16 is trained from scratch, is the PlantVillage crop diseases data set [2]. This data set consists of 52,266 images from 36 classes of various diseased and healthy crop leaves. The purpose of such a varied and diverse data set is to expose the VGG-16 model to a variety of plant disease features similar to those of Northern Leaf Blight. The ”healthy corn” class of images was withheld and used in Stage Two of the experiment to classify alongside pictures of Northern Leaf Blight. The PlantVillage data set also contained its own ”northern leaf blight” class, which was disposed of in favor of Stage Two’s primary data set. Example images from the PlantVillage data set can be found in Fig. 1.

The data utilized in stage two of this experiment is the OSF ”Field Images of Maize” data set [1]. This data set consists of several thousand images of Northern Leaf Blight in North

Fig. 2: Example image of Northern Leaf Blight on a leaf of corn.

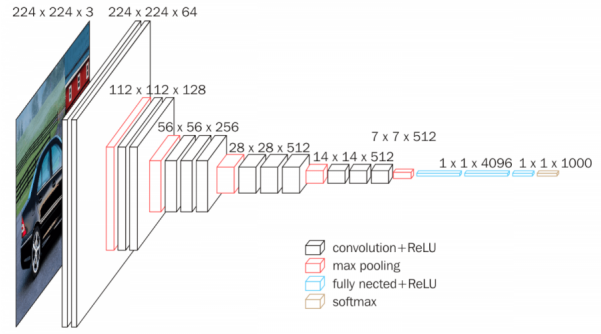


Fig. 3: Architecture of VGG-16.

American corn crops taken from three separate perspectives: a boom rod, a drone, and a handheld camera. From this data set, the project only utilizes images from the handheld category, as these images were the closest in perspective to those of the PlantVillage data set. Between the OSF images of Northern Leaf Blight and the PlantVillage images of healthy corn, Stage Two's data consisted of 7,964 images.

III. System Models

The network architecture used in the following experiments is the popular VGG-16 image classification model, accessed via Keras [3]. This model is consistently ranked as one of the best image classification systems available and was one of the top-performing models in the ImageNet classification competition. The overall structure used in each experiment of the project is shown in Fig. 3. The only differences made to the architecture of the model at each stage was removing the final prediction layer and replacing it with a layer that could classify the required number of classes at each stage (36 in Stage One, 2 in Stage Two).

All computations were performed on an Apple MacBook Pro 16" (2021) with an M1 Max processor.

IV. Methods

A. Data Processing

The data for each stage required a great deal of processing and organization. The data was first downloaded and then organized into directories according to a 70:10:20 train:test:validation split. Once the data was organized into the proper directories, it was inserted into a Keras image generator to be fed into the models. This image generator object also pre-processed the images according to the specifications of the VGG-16 model, which included re-sizing each image to an input size of (224,224), as well as zero-centering each image via mean pixel subtraction.

The data for Stage Two, the OSF data, required extra attention when resizing it. This is because of perspective differences between the PlantVillage and OSF images. In PlantVillage, the camera perspective is very close to each leaf. In OSF on the other hand, although the images were captured using a handheld camera, the camera is still a short distance away from the given leaf. Luckily, the data set came complete with a list of expertly annotated locations for leaf blight instances in each image. Using this information, the specific locations of each leaf blight spot were patched out using a square-cropping algorithm before being used in the experiments.

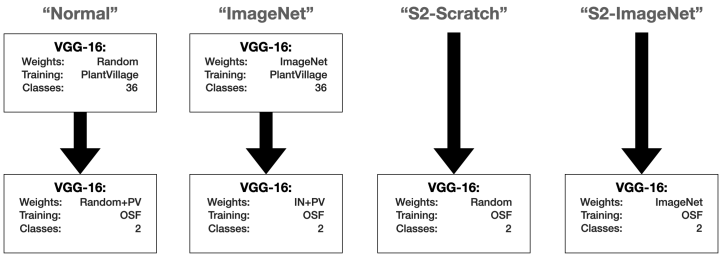


Fig. 4: Summary of Experimental Transfer Models

B. Model Selection and Construction

A handful of variations on the VGG-16 model were constructed in order to test which configuration would be most effective at the leaf blight classification problem. First, an initial transfer learning model was created by training VGG-16 from scratch on the PlantVillage data set, then using these weights to train again on the OSF data. This configuration is referred to as the "Normal" configuration. Another variation involved training VGG-16 not from scratch, but using its original ImageNet weights. This meant that VGG-16, initialized with its ImageNet weights, was trained first on the PlantVillage data set, then trained on the OSF data set and applied to the leaf blight classification problem. This model is referred to as the "ImageNet" configuration. The last two models tested were simpler models used for comparison. These models skipped Stage One entirely and trained the VGG-16 model directly on the OSF data for leaf blight: one from random weights and the other from ImageNet weights. These models are referred to as the "S2-Scratch" and "S2-ImageNet" configurations. A visualization of these configurations can be found in Fig. 4.

Each model followed the VGG-16 architecture exactly, aside from two key changes: prediction layer replacement and weight freezing. At each stage during an experiment, the last

dense layer of VGG-16 was removed and replaced with a new dense layer fit for that stage's classification problem. For example, in Stage One, VGG-16's last layer, with 1000 nodes for 1000 ImageNet classes, was replaced with a new dense layer containing 36 nodes, one for each PlantVillage class. Layer freezing occurred when the weights from a previous training session needed to be used in a transfer stage. For example, in Stage Two of the Normal model's experiments, the weights of every layer but the last layer were frozen so that the model could make use of the weights learned in Stage One.

The Adam optimizer was used to optimize each model, and categorical cross-entropy was used to evaluate loss.

C. Evaluating Models

A variety of experiments were performed to determine the efficacy of VGG-16 and the more general transfer learning method on this problem. These experiments attempted to roughly tune the hyperparameters (training epochs and learning rate) of each experimental model until the most effective configuration became apparent. Instead of doing a simple brute-force grid search of each combination of hyperparameters, which would take up valuable compute and time resources, the models were evaluated attentively at each experiment, with the next experiment for each model being chosen according to the model's past performance. For example, if the Normal configuration saw drastic drops in loss and accuracy across five epochs at a given learning rate, it would not be sensible to evaluate the model for a large number of epochs for that same learning rate.

Each of the four experimental transfer models (Normal, ImageNet, S2-Scratch, and S2-ImageNet) were first tested on a round that used five training epochs and a learning rate of 0.0001. This round solely served to get a preliminary idea of the performance potential of each classifier. From this first look, it was determined that none of the models made significant progress at any of the tested learning rates past 10 epochs. This is likely due to the Adam optimizer's ability to converge upon loss-function minima in a low number of epochs. For this reason, in addition to the preliminary 5-epoch sweep, each model was only tested at 10 epochs for each of the three learning rates.

Each experiment consisted of two stages: the PlantVillage training stage (Stage One) and the OSF data training stage (Stage Two). The only exceptions to this rule were the experiments performed on the S2 models, which skipped PlantVillage training and instead trained directly on the OSF dataset. Table 1 contains a summary of the experiments performed on each experimental model.

TABLE I: Summary of Experiments for each Transfer Model

Transfer Model	**Evaluation Criteria**	
	Epochs	*Learning Rate*
Normal	5	0.0001
	10	0.01
	10	0.001
	10	0.0001
ImageNet	5	0.0001
	10	0.01
	10	0.001
	10	0.0001
S2-Scratch	5	0.0001
	10	0.01
	10	0.001
	10	0.0001
S2-ImageNet	5	0.0001
	10	0.01
	10	0.001
	10	0.0001

V. Results

Different results were recorded for each stage of the experiments. After Stage One of each experiment, the training and validation accuracy, as well as the categorical cross-entropy loss, were recorded and plotted for a given model. Following stage two, the training and validation accuracy and loss were again plotted for the model. Additionally, a confusion matrix, an ROC curve, and a Precision-Recall curve were all recorded for the final transfer model. Although these visualizations were created for each experimental model, only those visualizations from to the best-performing transfer models in each category (Normal, ImageNet, S2-Scratch, and S2-ImageNet) were included in this section. The remaining plots and visualizations will be available in the submission file for this project. The testing accuracy (calculated using 5-fold cross-validation) and loss from each experiment will be included in Table 2, where the highest accuracy value and the lowest loss value in each column is boldfaced. Additionally, the best-performing hyperparameters for each transfer model are boldfaced in this table.

VI. Discussion

After all experiments were performed and all results were recorded, the ImageNet transfer model ended up being the most effective at classifying images of Northern Leaf Blight. It was hypothesized that from the aforementioned experiments, the "Normal" model would perform the best on Northern Leaf Blight vs Healthy leaf classification, with the ImageNet model being a close second or even better than the Normal model. The Normal model (trained from scratch on PlantVillage, then transferred to OSF) was hypothesized to be the most effective as it had been trained only on images of plant diseases. This was thought to be an advantage that Normal could potentially have over the ImageNet model, as the ImageNet model was initialized using weights from a 1000-class classification problem. With so many classes at play, almost none of which were related to images from the PlantVillage or OSF data sets, it was thought that the ImageNet model might have confounding feature templates or other artifacts from its prior weights that would interfere with a plant-disease-exclusive data set. As it turns out, however, the ImageNet transfer model was able to effectively apply its knowledge of cats, bicycles, and 998 other classes to the problem of Northern Leaf Blight detection.

TABLE II: Summary of Test Results for Each Model

Transfer Model	**Evaluation Criteria**		**Test Results**			
	Epochs	*Learning Rate*	*Stage 1 Acc.*	*Stage 1 Loss*	*Stage 2 Acc.*	*Stage 2 Loss*
Normal	**5**	**0.0001**	0.9247	0.2953	0.95226	0.1142
	10	0.01	0.1058	3.284	0.8543	0.4155
	10	0.001	0.1054	3.284	0.8328	3.165
	10	0.0001	0.9234	0.3153	0.9422	0.1429
ImageNet	5	0.0001	0.9459	0.1663	0.9899	0.03172
	10	0.01	0.9376	4.667	0.9899	0.3899
	10	**0.001**	0.9340	0.5017	**0.9925**	**0.02814**
	10	0.0001	**0.9553**	**0.1341**	0.9874	0.03538
S2-Scratch	5	0.0001	—	—	0.9673	0.1952
	10	0.01	—	—	0.8543	3.882
	10	0.001	—	—	0.8543	0.4153
	10	**0.0001**	—	—	0.9849	0.07688
S2-ImageNet	5	0.0001	—	—	0.9887	0.03575
	10	0.01	—	—	0.9874	0.7633
	10	**0.001**	—	—	0.9899	0.06181
	10	0.0001	—	—	0.9899	0.2811

I hypothesize that this result is mostly due to the massive amount of training data that the ImageNet weights are informed on. The VGG-16 model is most popular for its performance on the ImageNet data set, which contains more than 14 million images of 1000 classes. VGG-16's ability to retain mostly relevant features after so many different training examples is a testament to its robust architecture, and the VGG-16 model performed the best in these experiments when its weights were informed by large amounts of training data. This trend can be seen in Table II, where the 5-fold cross-validation accuracy was noticeably higher for the ImageNet-informed transfer models (ImageNet and S2-ImageNet) than in the models whose weights were initialized randomly. With this being said, it is also worth noting that the ImageNet model, which had been trained on both the ImageNet and PlantVillage data sets, was trained on the most data and also also featured the highest accuracy across all transfer models.

Another interesting observation is that although the ImageNet transfer model performed the best overall, the S2-ImageNet performed nearly as well without being trained on the PlantVillage data set at all. This is likely due to the fact the ImageNet weights themselves were responsible for the performance of the models. Given that the PlantVillage data set is so small (approx. 52,000 images) in comparison to the ImageNet data set (>14,000,000), it would be reasonable to conclude that the original ImageNet weights were not very heavily modified in any of the experiments they were involved in. It would be interesting to repeat this experiment with a much larger plant disease data set (with several million images) to compare the performance of a model trained solely on millions of plant disease images and a model trained on the same number of images, but on the extremely diverse ImageNet data set instead. In the meantime, it seems that using solely ImageNet weights in VGG-16 will cut down on a great deal of training time overall and still achieve accurate results on the Northern Leaf Blight classification problem.

Effects of the class distribution of the data can be seen in the final results as well. Looking at the confusion matrices for each model (Figs. 5c-8c), one can observe a consistent amount of errors in the lower-left quadrant, where healthy images were misclassified as having Northern Leaf Blight. The same trend can also be seen in the ROC and Precision-Recall curves for each model, where the ROC True Positive rate, $\frac{TP}{TP+FN}$, was visibly lower than its counterpart (Figs. 5d-8d) and the Recall was consistently lower than the Precision (Figs. 5e-8e). This was likely due to an overabundance of blight images in the data, where the blight class outnumbered the healthy class at a ratio of 6:1. To combat this issue, data augmentation was performed on the OSF images, but the issue still persisted in some of the experimental results. Cutting down on the number of blight images could be a possible solution, but given that the data set was already very small, it was thought that having more data with a skewed distribution would be better than no data at all. Although it was not possible to obtain more images for this project, it is recommended that in future applications more images be obtained of healthy corn, thereby evening out the model's understanding of each class.

VII. Conclusion

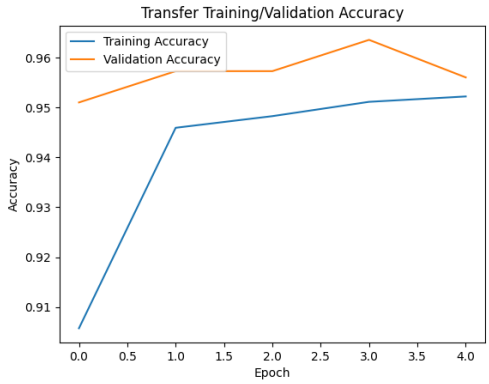
The proposed methods were able to classify images of Northern Leaf Blight with a high degree of accuracy. Though improvements to class distribution and data availability could make the models more robust, the transfer learning method was successful in using the given data to solve the problem.

References

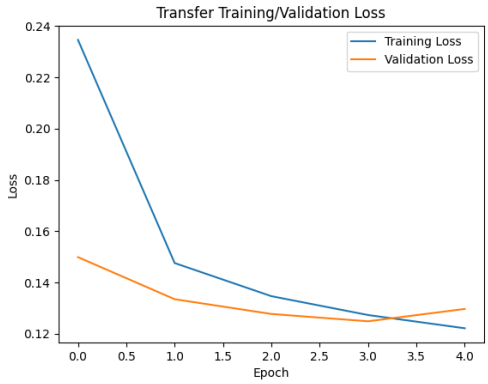
[1] T. Wiesner-Hanks and M. Brahimi, "Image set for Deep Learning: Field images of maize annotated with disease symptoms," OSF, 28-Mar-2018. [Online]. Available: https://osf.io/p67rz/wiki/home/. [Accessed: 15-Dec-2021].

[2] M. Salathe, "An open access repository of images on plant health to enable the development of Mobile Disease Diagnostics," arXiv.org, 12-Apr-2016. [Online]. Available: https://arxiv.org/abs/1511.08060. [Accessed: 15-Dec-2021].

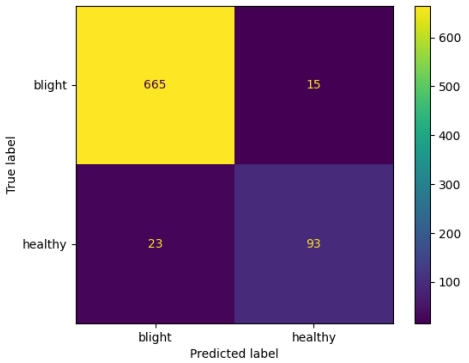
[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv.org, 10-Apr-2015. [Online]. Available: https://arxiv.org/abs/1409.1556. [Accessed: 15-Dec-2021].



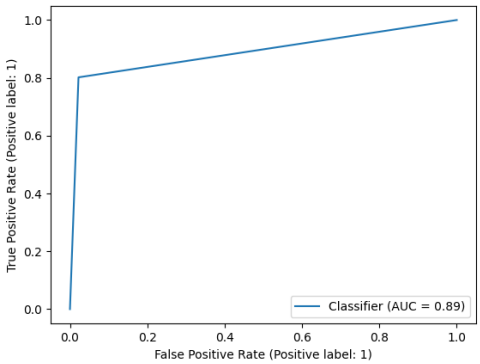
(a) Accuracy



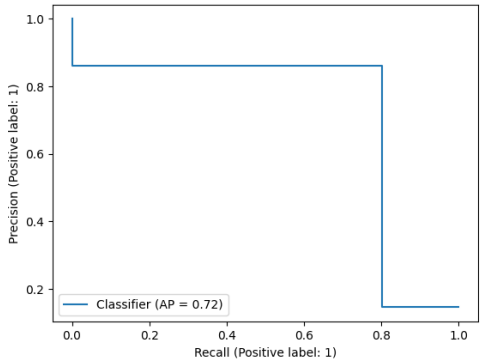
(b) Loss



(c) Confusion Matrix

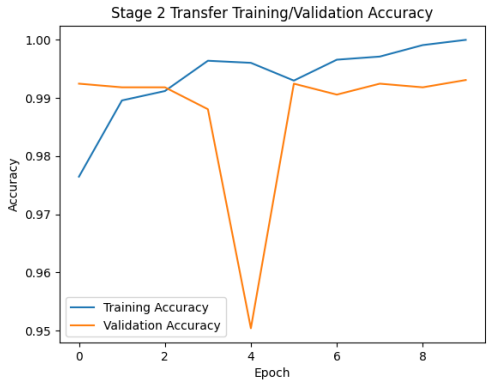


(d) ROC Curve

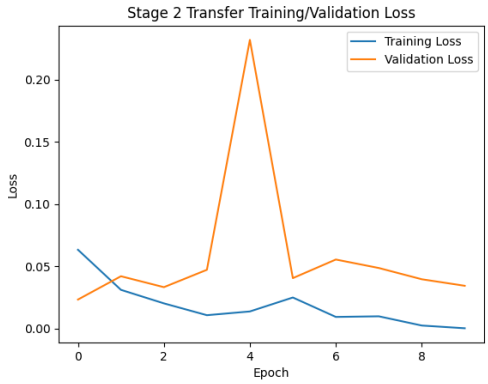


(e) Precision-Recall Curve

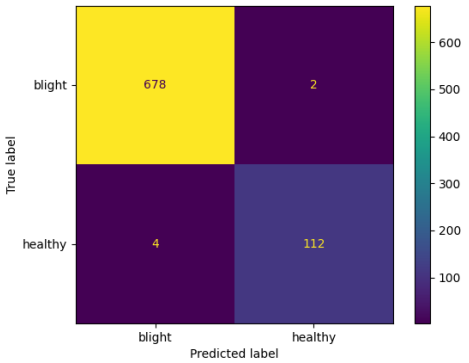
Fig. 5: Results of Top-performing Normal model (5 Epochs, LR=0.0001)



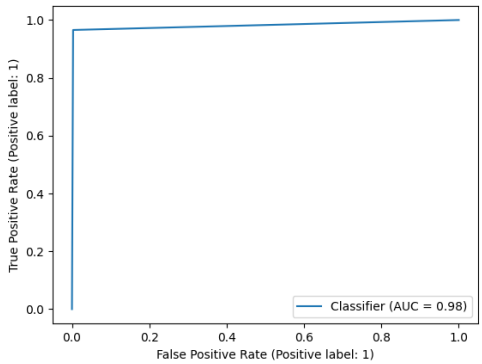
(a) Accuracy



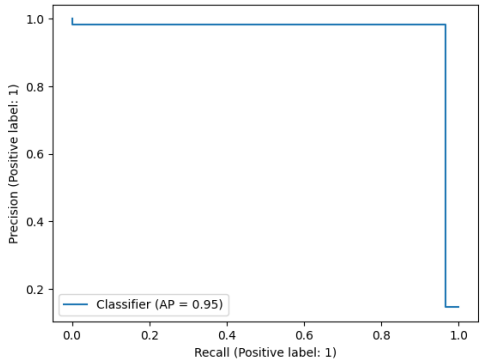
(b) Loss



(c) Confusion Matrix

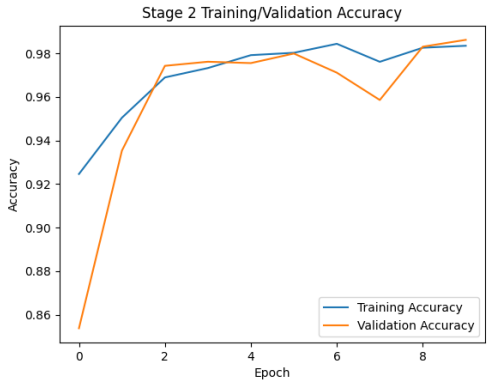


(d) ROC Curve

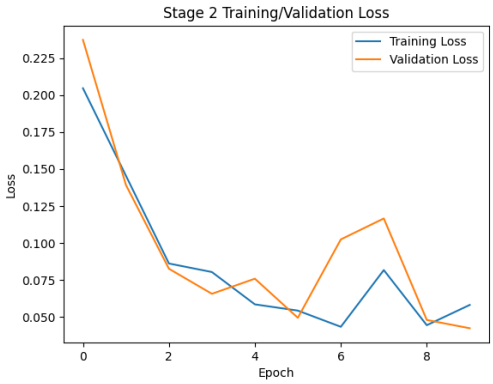


(e) Precision-Recall Curve

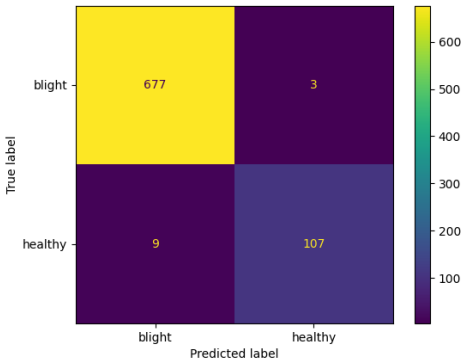
Fig. 6: Results of Top-performing ImageNet model (10 Epochs, LR=0.001)



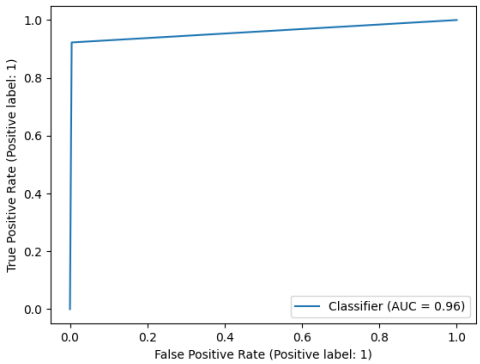
(a) Accuracy



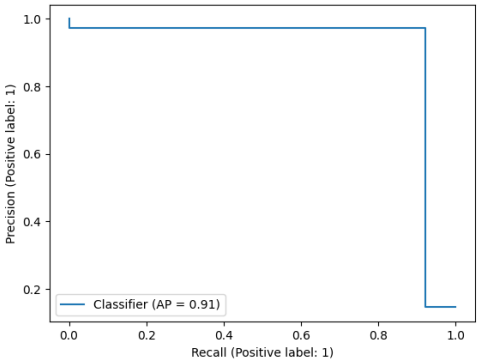
(b) Loss



(c) Confusion Matrix

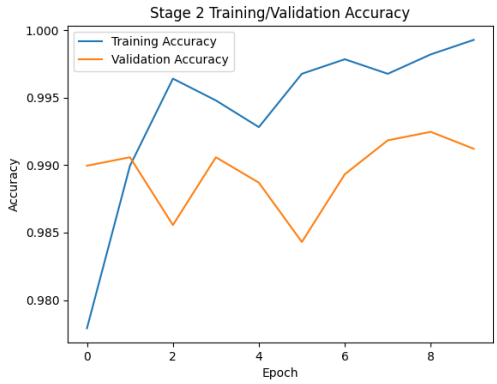


(d) ROC Curve

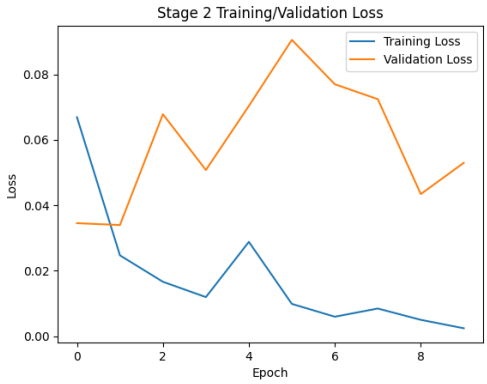


(e) Precision-Recall Curve

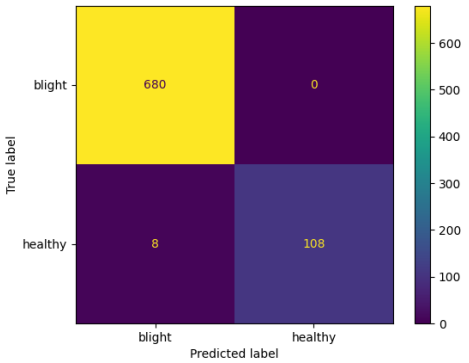
Fig. 7: Results of Top-performing S2-Scratch model (10 Epochs, LR=0.0001)



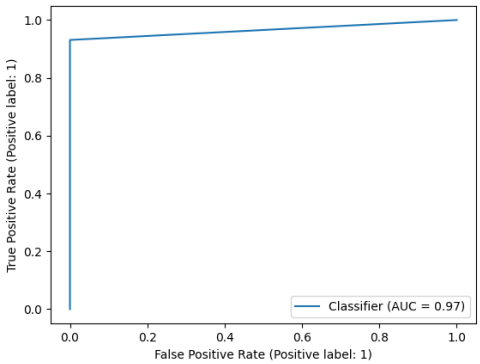
(a) Accuracy



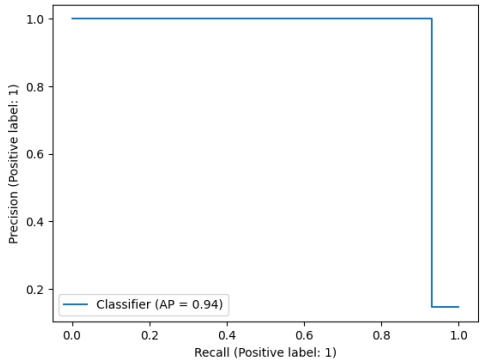
(b) Loss



(c) Confusion Matrix



(d) ROC Curve



(e) Precision-Recall Curve

Fig. 8: Results of Top-performing S2-ImageNet model (10 Epochs, LR=0.001)