

Machine Learning Log

For all the following models, the same model architecture (EfficientNet B5), dataset, and optimizer (Adam) was used. Changes were only made to tune the listed hyperparameters. The source code can be found on my Github page: <https://github.com/isaacC-2B/Rope-Science>

v1Data-Modelv1

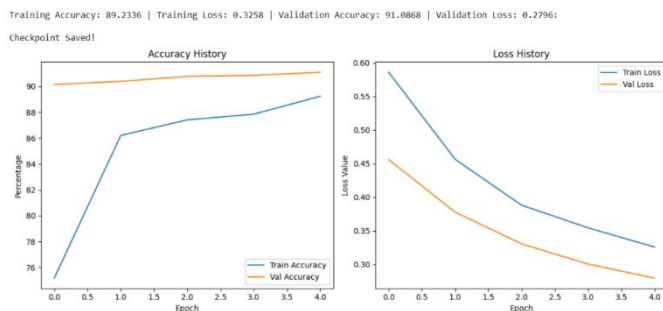
Epochs: 5

Batch size: 32

Learning Rate: 1e-4

Observations – Good accuracy but the loss value remains high and the difference between the training and the validation accuracy and loss indicates underfitting, a term used to describe when the model is too simple to recognize and learn the patterns of the images in the training dataset

Next steps – The future model(s) should be trained for longer, other batch sizes should be tested



v1Data-Modelv2

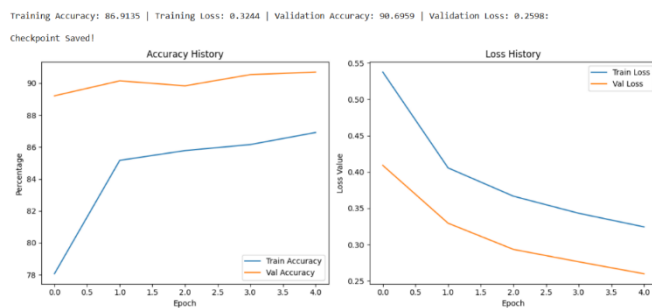
Epochs: 5

Batch size: 8

Learning Rate: 1e-4

Observations – Lower accuracy but similar loss when compared to previous v1 model, underfitting is observed

Next steps – The future model(s) should be trained for longer and have a batch size greater than 8



v1Data-Modelv3

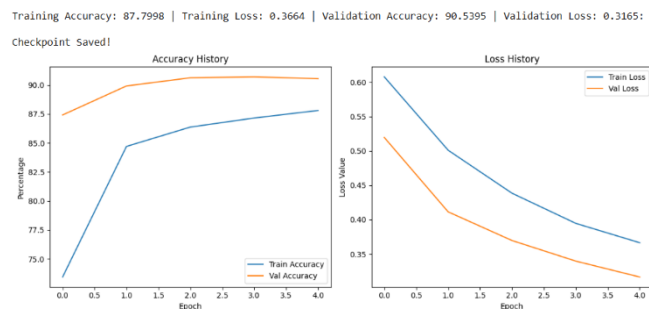
Epochs: 5

Batch size: 64

Learning Rate: 1e-4

Observations – greater accuracy than the v2 model but less than the v1 model, higher loss than both previous models, underfitting is observed

Next steps – The future model(s) should be trained for longer and have a batch size of less than or equal to 32



v1Data-Modelv4**Epochs: 5****Batch size: 16****Learning Rate: 1e-4****Observations –**

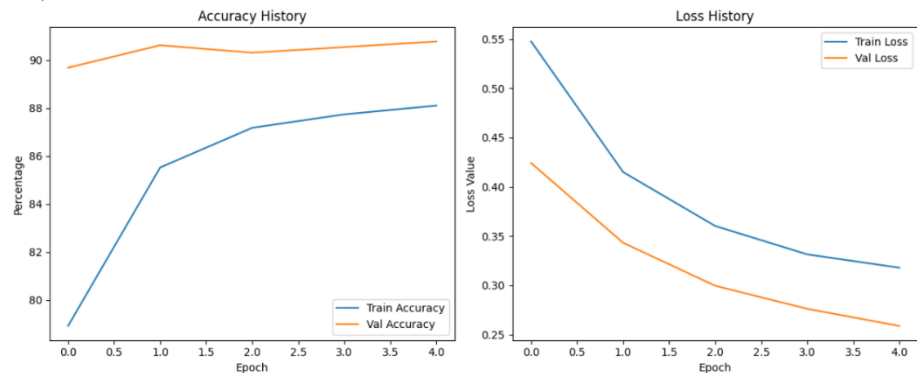
greater accuracy than the v2 and v3 models but less than the v1 model, slightly lower loss than all previous models but a greater

difference between validation loss and training loss can be seen, underfitting is observed

Next steps – The next model should be trained for longer and have a batch size of 32

Training Accuracy: 88.0996 | Training Loss: 0.3179 | Validation Accuracy: 90.7740 | Validation Loss: 0.2588:

Checkpoint Saved!

v1Data-Modelv5**Epochs: 10****Batch size: 32****Learning Rate: 1e-4****Observations –**

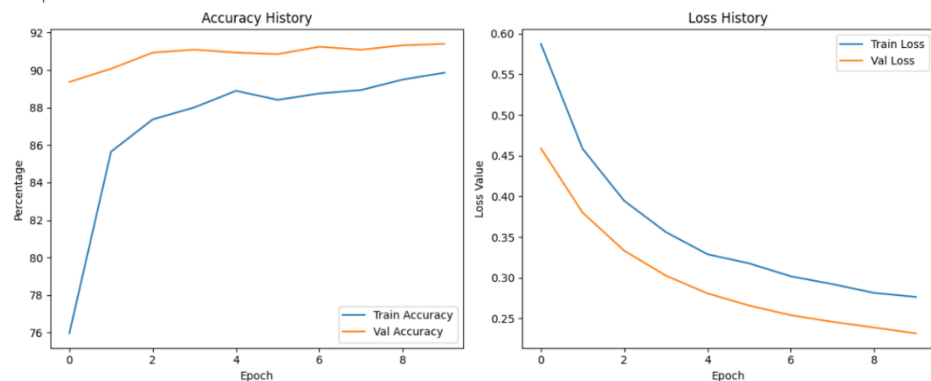
Validation accuracy begins to plateau at around the 4th epoch while the training accuracy continues to climb, both validation

and training loss continues to decrease but at a much slower rate with the gap between the two still indicating underfitting of the training dataset

Next steps – The next model should be trained for a few more epochs and/or blocks of the EfficientNet B5 structure should be unfrozen to further improve the accuracy of the model. The initial learning rate could be increased to 1e-3 or decreased to 1e-5 to potentially improve model training performance

Training Accuracy: 89.8592 | Training Loss: 0.2764 | Validation Accuracy: 91.3995 | Validation Loss: 0.2316:

Checkpoint Saved!



v1Data-Modelv6

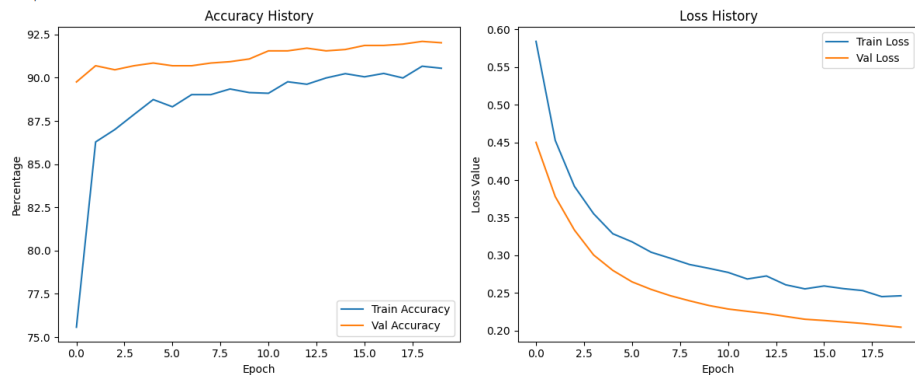
Epochs: 20
Batch size: 32
Learning Rate: 1e-4

Observations –
 Accuracy continues to improve at a slow but linear rate after the 5th epoch, loss continues to

decrease but the gap between validation and training loss and validation and training accuracy remains thus, underfitting is still observed

Next steps – The next model could be further trained for 10 more epochs to provide enough time for the model’s accuracy to rise and the loss to fall

Training Accuracy: 90.5501 | Training Loss: 0.2463 | Validation Accuracy: 92.0250 | Validation Loss: 0.2046:
 Checkpoint Saved!



v1Data-Modelv7

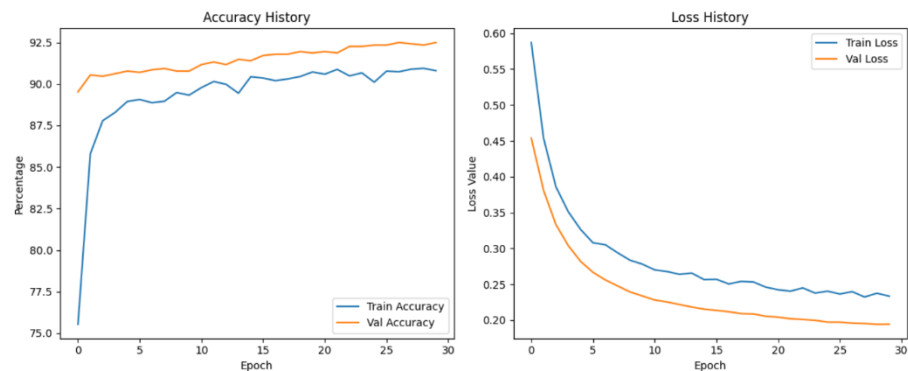
Epochs: 30
Batch size: 32
Learning Rate: 1e-4

Observations –
 Accuracy and loss improvements over the v6 model are minimal, the loss and accuracy

improvements begin to plateau at around the 20th epoch, and the gap between the accuracies and losses has barely shrunk thus underfitting is still observed

Next steps – Around 20 epochs is the ideal training length if all but the final layer of the model is frozen, unfreezing blocks of the EfficientNet B5 structure and adjusting the initial learning rate should be attempted to further improve the accuracy of the model.

Training Accuracy: 90.7977 | Training Loss: 0.2335 | Validation Accuracy: 92.4941 | Validation Loss: 0.1944:



v1Data-Modelv8

Epochs: 10

Batch size: 32

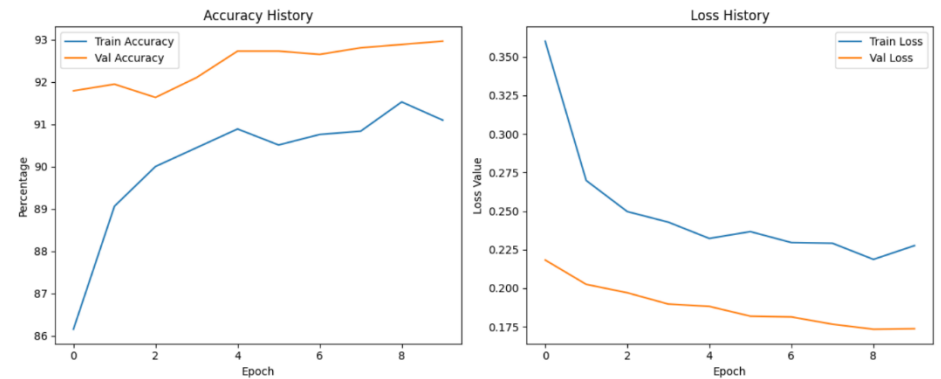
Learning Rate: 1e-3

Observations – The faster learning rate resulted in a higher accuracy and lower loss than any other model, accuracy and loss between epochs were not as smooth due to the faster learning rate and the gap between the validation and training loss still exists showing underfitting

loss between epochs were not as smooth due to the faster learning rate and the gap between the validation and training loss still exists showing underfitting

Next steps – Train the model for longer to allow the loss to further decrease and the accuracy to further increase, implement unfreezing blocks

Training Accuracy: 91.0975 | Training Loss: 0.2276 | Validation Accuracy: 92.9633 | Validation Loss: 0.1737:



v1Data-Modelv9

Epochs: 10

Batch size: 32

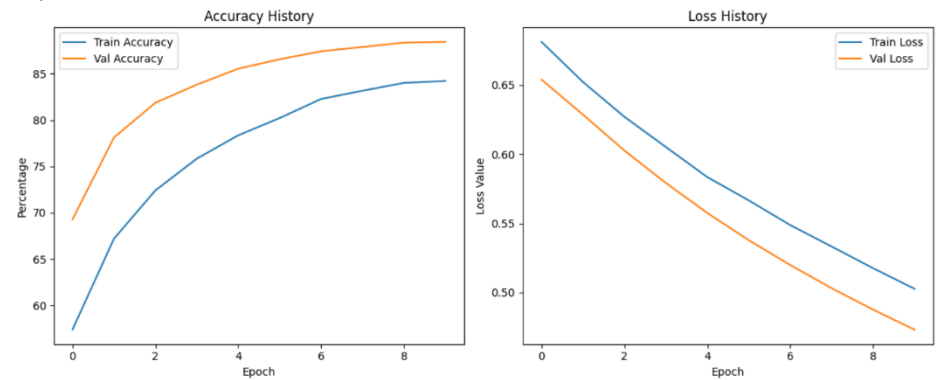
Learning Rate: 1e-5

Observations – The slower learning rate resulted in lower accuracy and loss as the model trained too slowly

Next steps – Return back to the initial 1e-3 learning rate, implement unfreezing blocks

Training Accuracy: 84.2153 | Training Loss: 0.5027 | Validation Accuracy: 88.4285 | Validation Loss: 0.4732:

Checkpoint Saved!



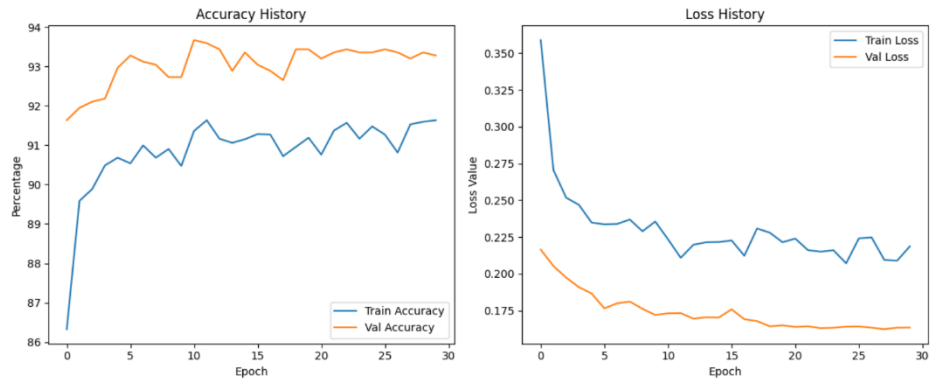
v1Data-Modelv10

Epochs: 30 (Block 8 unfrozen at Epoch 20)

Batch size: 32

Learning Rate: 1e-3 – 1e-8 (decreased by a multiplier of 0.1 based on the learning rate scheduler function with a *patience* set for 3 epochs)

Training Accuracy: 91.6319 | Training Loss: 0.2186 | Validation Accuracy: 93.2760 | Validation Loss: 0.1634



Observations – The model struggled to improve making the *patience* function continually decrease the learning rate. The accuracy peaked and the training loss was at its least during epochs 11-12, the accuracy and loss barely improved from the unfreezing of the 8th block likely due to learning rate (1e-6) becoming too low during that period of time, the overall pattern of training is full of fluctuation and appears to be almost random after the 10th epoch

Next steps – Remove the learning rate scheduler’s *patience* function and manually adjust the learning rate between epochs to prevent the learning rate from decreasing; begin unfreezing block 8 sooner and try unfreezing block 7, train for less time in total as the loss and accuracy plateaued after the 20th epoch

v1Data-Modelv11

Epochs: 20 (Block 8 unfrozen at Epoch 10, Block 7 unfrozen at Epoch 15)

Batch size: 32

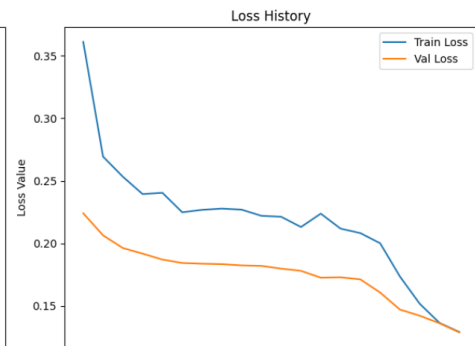
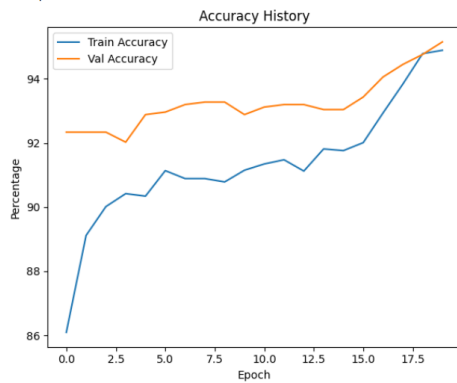
Learning Rate: 1e-3 (1e-4 at Epoch 5, 1e-5 at Epoch 10, 1e-6 at Epoch 15)

Observations – The unfreezing of the 7th block resulted in higher accuracy and lower loss on both the training and validation datasets. Underfitting was also eliminated with the unfreezing of block 7 and it appears that the model could likely further improve if the unfreezing began earlier

Next steps – Shift the unfreezing of blocks 7 and 8 a few epochs earlier to allow for the model to achieve better results in a lower number of epochs, attempt to unfreeze block 6 in the later stages of training

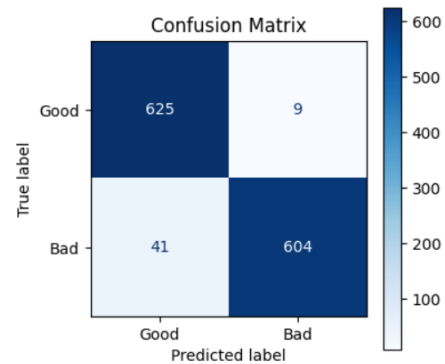
Training Accuracy: 94.8905 | Training Loss: 0.1290 | Validation Accuracy: 95.1525 | Validation Loss: 0.1287:

Checkpoint Saved!



Accuracy: 96.09%
precision recall f1-score support

Good	0.94	0.99	0.96	634
Bad	0.99	0.94	0.96	645
accuracy			0.96	1279
macro avg	0.96	0.96	0.96	1279
weighted avg	0.96	0.96	0.96	1279



v1Data-Modelv12

Training Accuracy: 96.7414 | Training Loss: 0.0862 | Validation Accuracy: 95.4652 | Validation Loss: 0.1249:
Checkpoint Saved!

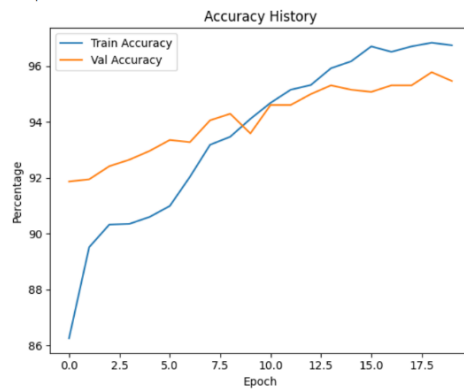
Epochs: 20 (Block 8 unfrozen at Epoch 5, Block 7 unfrozen at Epoch 10, Block 6 unfrozen at Epoch 15)

Batch size: 32

Learning Rate: 1e-3 (1e-4 at Epoch 5, 1e-5 at Epoch 10, 1e-6 at Epoch 15)

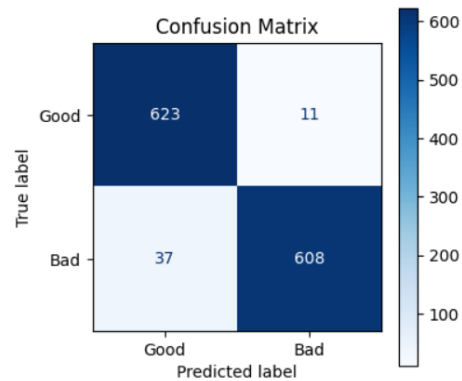
Observations – Overfitting (when the model begins learning the specific features of images in the data rather than identifying common collective features resulting in higher training accuracy and lower loss than on the validation dataset) began at epoch 12 due to the faster learning rate (compared to the previous v11 model) when unfreezing the 7th block, unfreezing the 6th block resulted in little to no improvement and barely added to the overfitting of the training data

Next steps – reduce the learning rate down to 1e-6 when unfreezing the 7th block



Accuracy: 96.25%

	precision	recall	f1-score	support
Good	0.94	0.98	0.96	634
Bad	0.98	0.94	0.96	645
accuracy			0.96	1279
macro avg	0.96	0.96	0.96	1279
weighted avg	0.96	0.96	0.96	1279



v1Data-Modelv13

Epochs: 20 (Block 8 unfrozen at Epoch 5, Block 7 unfrozen at Epoch 10, Block 6 unfrozen at Epoch 15)

Batch size: 32

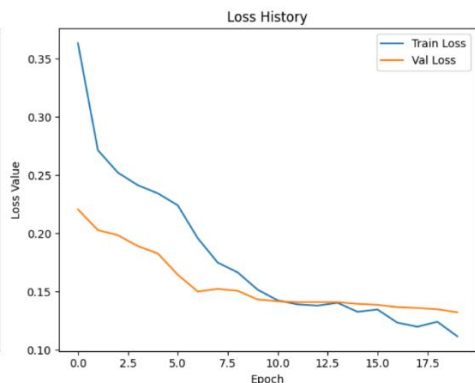
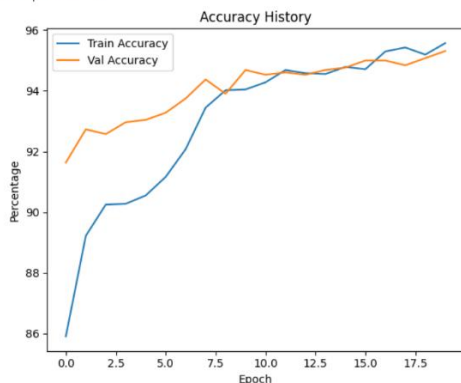
Learning Rate: 1e-3 (1e-4 at Epoch 5, 1e-6 at Epoch 10, 1e-6 at Epoch 15)

Observations – With a lower learning rate of 1e-6 at epoch 10, the amount of overfitting decreased by a large margin resulting in more consistent accuracy and loss between the training and validation datasets, although this change improved the accuracy, a small gap between the validation and training loss still exists; both the training and validation accuracies were also lower than on the previous v12 model

Next steps – Using observations from models v11-13 the learning rate when unfreezing block 8 should be set to 1e-4, as in previous models the 1e-5 learning rate allowed for only minimal changes, for the unfreezing of block 7 the learning rate should decrease to 1e-6 in order to prevent the model from overfitting as seen in the v12 model; finally the 6th block should not be unfrozen as it results in little to no increase in performance

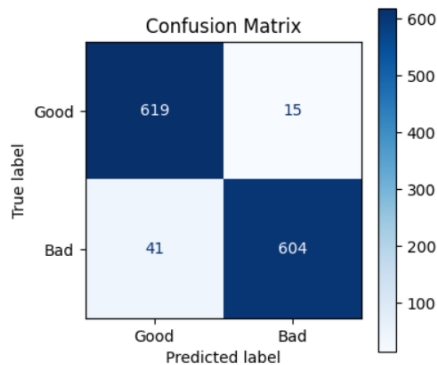
Training Accuracy: 95.5683 | Training Loss: 0.1115 | Validation Accuracy: 95.3088 | Validation Loss: 0.1321:

Checkpoint Saved!



Accuracy: 95.62%

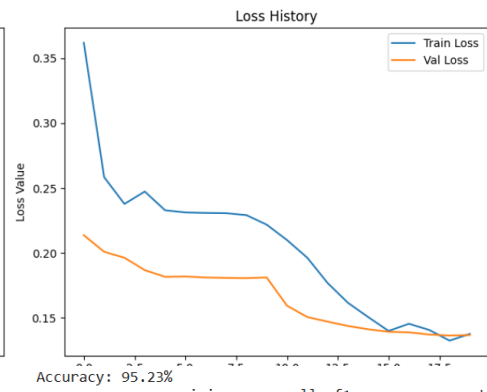
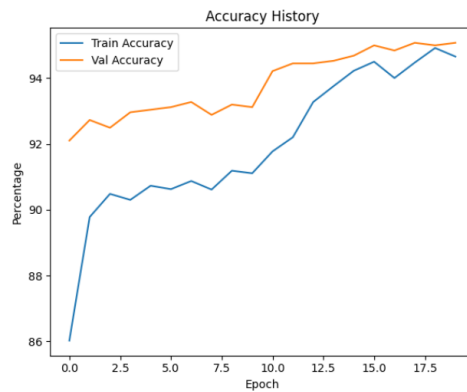
	precision	recall	f1-score	support
Good	0.94	0.98	0.96	634
Bad	0.98	0.94	0.96	645
accuracy			0.96	1279
macro avg	0.96	0.96	0.96	1279
weighted avg	0.96	0.96	0.96	1279



v1Data-Modelv14

Epochs: 20 (Block 8 unfrozen at Epoch 10, Block 7 unfrozen at Epoch 15)
Batch size: 32
Learning Rate: 1e-3 (1e-4 at Epoch 5, 1e-4 at Epoch 10, 1e-6 at Epoch 15)

Training Accuracy: 94.6559 | Training Loss: 0.1376 | Validation Accuracy: 95.0743 | Validation Loss: 0.1368:



Accuracy: 95.23%

	precision	recall	f1-score	support
Good	0.93	0.97	0.95	634
Bad	0.97	0.93	0.95	645
accuracy			0.95	1279
macro avg	0.95	0.95	0.95	1279
weighted avg	0.95	0.95	0.95	1279

Observations – The unfreezing of the 8th block with a learning rate of 1e-4 was highly effective at bringing up the training accuracy while avoiding overfitting, the overall training and validation accuracy was the lowest of any of the models trained with the unfreezing of block, virtually no overfitting of the training dataset was observed, observations from previous models are now providing insubstantial gains in performance as fluctuations in the model training process make it difficult to further fine tune the training hyperparameters

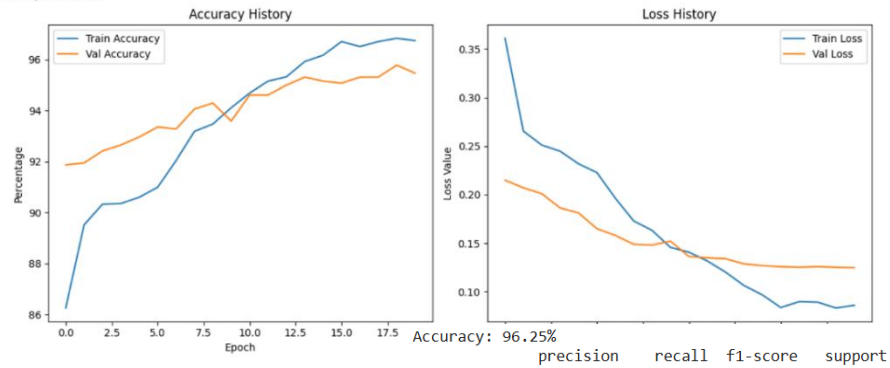
Next steps – Select one of four latest models (v11-14) based on the model’s performance on the test dataset shown in the classification reports and confusion matrixes as the performance of the models would be incredibly difficult to further improve



Final selected model - v1Data-Modelv12

Epochs: 20 (Block 8 unfrozen at Epoch 5, Block 7 unfrozen at Epoch 10, Block 6 unfrozen at Epoch 15)
Batch size: 32
Learning Rate: 1e-3 (1e-4 at Epoch 5, 1e-5 at Epoch 10, 1e-6 at Epoch 15)

Training Accuracy: 96.7414 | Training Loss: 0.0862 | Validation Accuracy: 95.4652 | Validation Loss: 0.1249:
 Checkpoint Saved!



Why this model was selected – This version 12 model was selected as it had the highest validation and training accuracy and lowest loss of any other model. Additionally when looking at the confusion matrix, this model incorrectly predicted 37 images that were labeled as “bad” to be “good”, this was the least of any model which is ideal given the purpose of the device is to provide an additional layer of security, thus incorrectly identifying true positives (instances where the rope is damaged) would potentially be much more costly than incorrectly identifying true negatives (instances where the rope is not damaged).

	precision	recall	f1-score	support
Good	0.94	0.98	0.96	634
Bad	0.98	0.94	0.96	645
accuracy			0.96	1279
macro avg	0.96	0.96	0.96	1279
weighted avg	0.96	0.96	0.96	1279

