REPORT

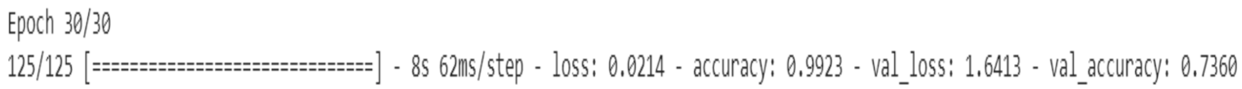
ADVANCED MACHINE LEARNING

ASSIGNMENT-03

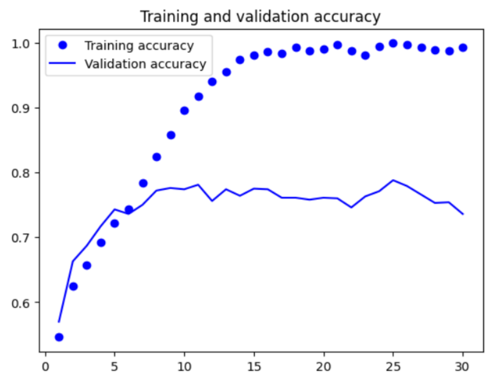
This code defines a convolutional neural network (CNN) using the Keras API in TensorFlow.In this report the investigation of the connection between training samples and the model training methods can observed.

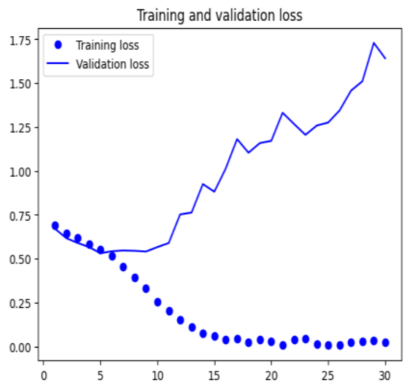
Starting up with the convolution method with the optimization technique along with the pretrained network.

BEGINNING WITH THE TRAINING SAMPLE OF 1000:

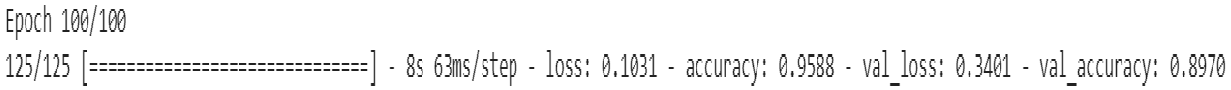


```
Epoch 30/30
125/125 [==============================] - 8s 62ms/step - loss: 0.0214 - accuracy: 0.9923 - val_loss: 1.6413 - val_accuracy: 0.7360
```

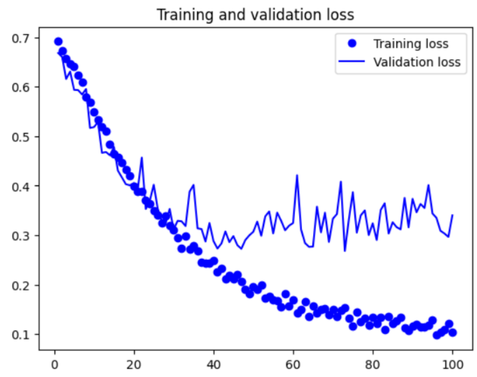


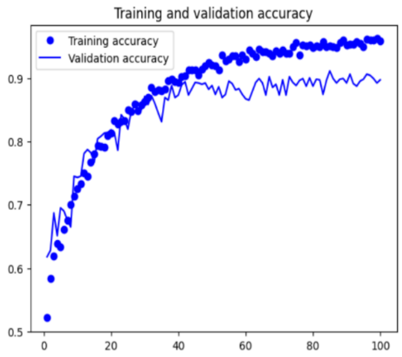


However, the validation loss is 1.6413 and the validation accuracy is 0.7360, which point to overfitting.

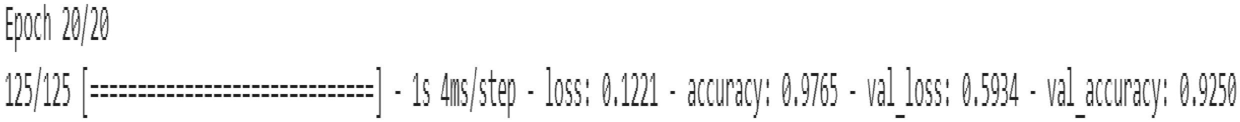


```
Epoch 100/100
125/125 [==============================] - 8s 63ms/step - loss: 0.1031 - accuracy: 0.9588 - val_loss: 0.3401 - val_accuracy: 0.8970
```

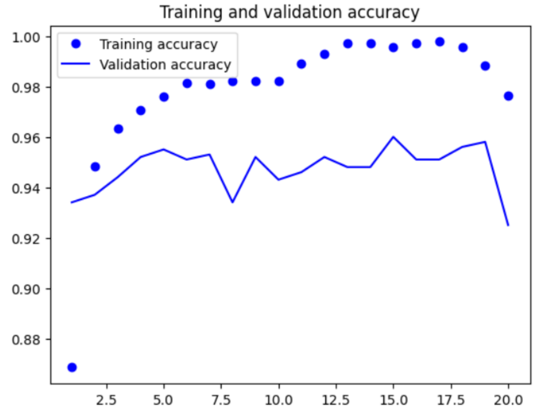


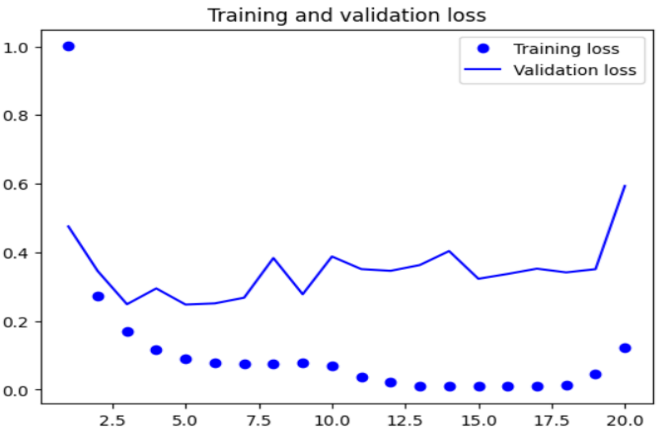


For a classification problem, the given code trains a neural network model. Although the model performs poorly on the validation data, it performs well on the training data, suggesting overfitting. To enhance the model's performance, additional research and model tuning may be required.

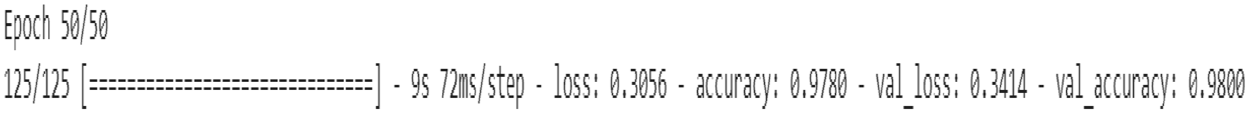


```
Epoch 20/20
125/125 [==============================] - 1s 4ms/step - loss: 0.1221 - accuracy: 0.9765 - val_loss: 0.5934 - val_accuracy: 0.9250
```

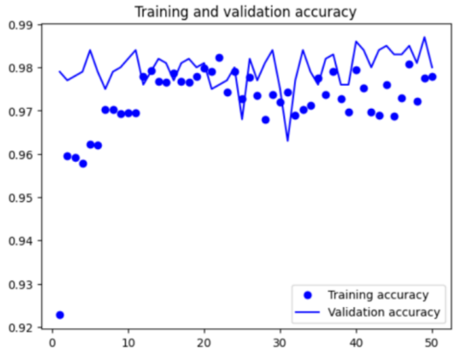


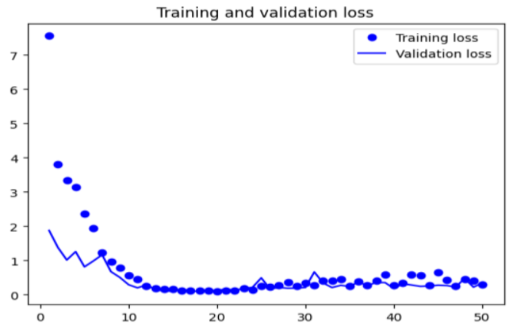


The model obtained a training accuracy of 0.9588 and a training loss of 0.1031 towards the conclusion of the 100th epoch. The accuracy of the validation was 0.8970, while the validation loss was 0.3401. These findings imply that the model has improved its ability to categorize training data with some degree of accuracy and that it is generalizing to validation data with some degree of proficiency. However, since the validation accuracy is marginally worse than the training accuracy, overfitting may be taking place.



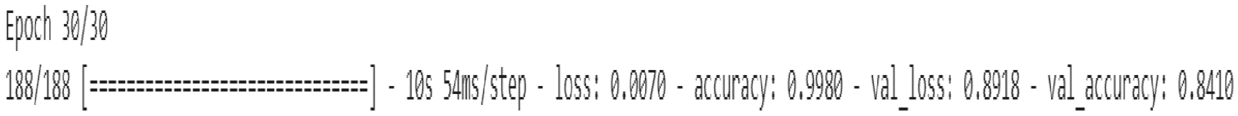
```
Epoch 50/50
125/125 [==============================] - 9s 72ms/step - loss: 0.3056 - accuracy: 0.9780 - val_loss: 0.3414 - val_accuracy: 0.9800
```



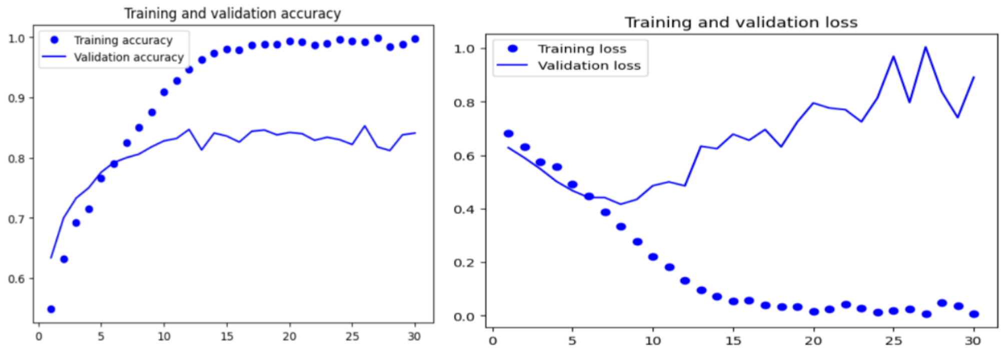


The model had a training loss of 0.1221 and a training accuracy of 0.9765 at the conclusion of the 20th epoch. The validation accuracy was 0.9250 and the validation loss was 0.5934.These findings imply that the model is performing well in terms of generalization to the validation data and has learned to accurately classify the training data. Performance of the model seems to be fairly decent overall.

USIMG THE SAMPLE OF 4000: The following are the outcomes and comparisions

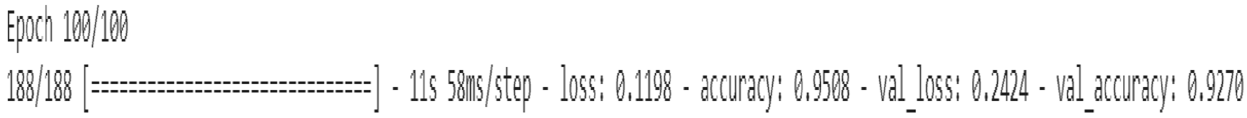


```
Epoch 30/30
188/188 [==============================] - 10s 54ms/step - loss: 0.0070 - accuracy: 0.9980 - val_loss: 0.8918 - val_accuracy: 0.8410
```

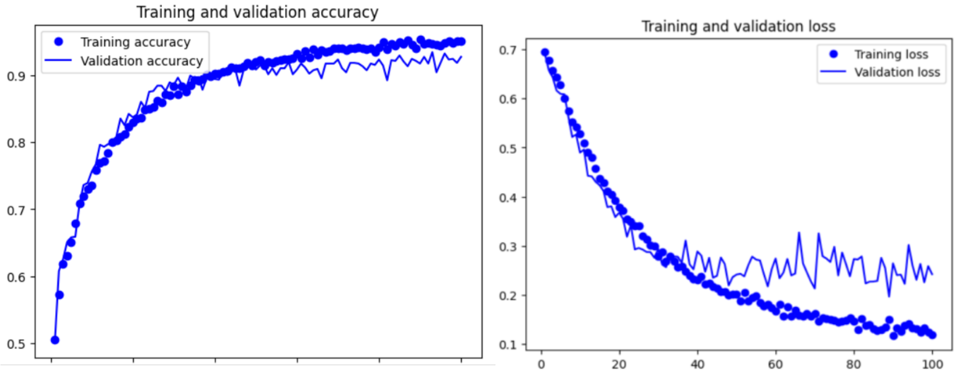


The model obtained a training accuracy of 0.9980 and a training loss of 0.0070 by the conclusion of the 30th epoch. However, the accuracy of the validation was 0.8410, and the validation loss was 0.8918.

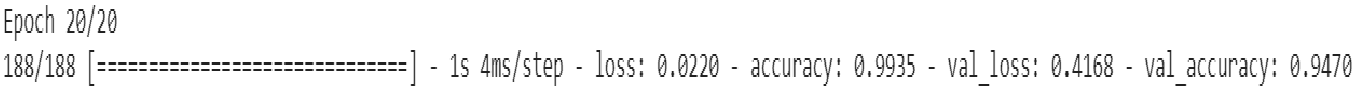
According to these findings, the model has mastered the art of accurately classifying training data but struggling to apply that knowledge to the validation data.



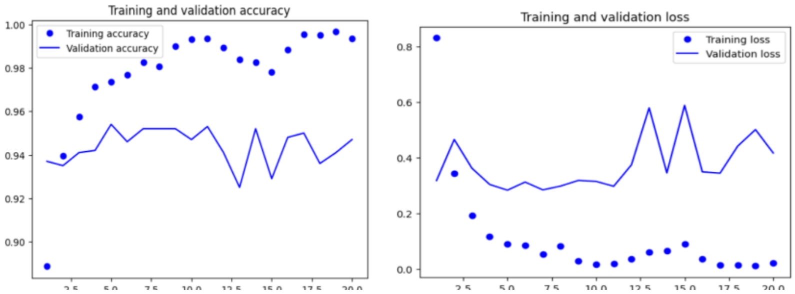
```
Epoch 100/100
188/188 [==============================] - 11s 58ms/step - loss: 0.1198 - accuracy: 0.9508 - val_loss: 0.2424 - val_accuracy: 0.9270
```



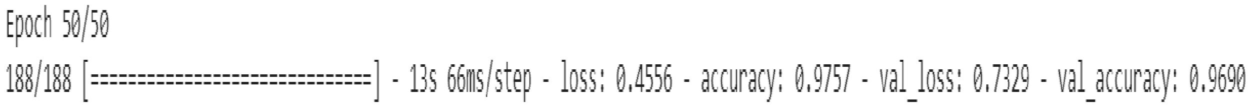
A training loss of 0.1198 and a training accuracy of 0.9508 were attained by the model by the conclusion of the 100th epoch. Validation accuracy was 0.9270 and validation accuracy was 0.2424.These outcomes indicate that the model is generalizing effectively to the validation data and has learned to classify the training data with a fair degree of accuracy. The validation accuracy is nearly identical to the training accuracy, demonstrating that the model did not overfit the training set of data.



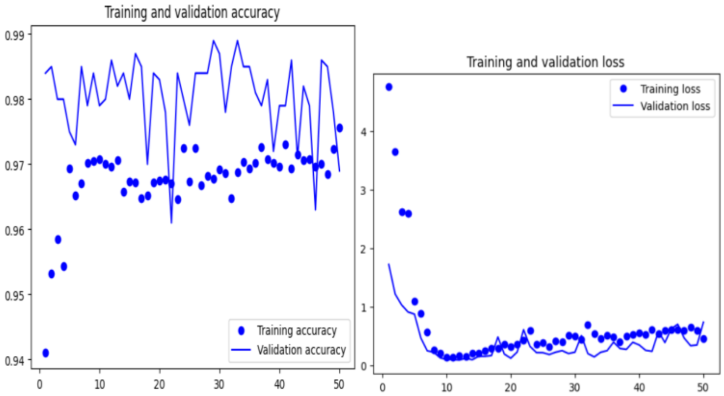
```
Epoch 20/20
188/188 [==============================] - 1s 4ms/step - loss: 0.0220 - accuracy: 0.9935 - val_loss: 0.4168 - val_accuracy: 0.9470
```



The model attained a training accuracy of 0.9935 and a training loss of 0.0220 by the conclusion of the 20th epoch. The accuracy of the validation was 0.9470, while the validation loss was 0.4168. These results indicate that the model is performing well in terms of generalization to the validation data and has learned to accurately classify the training data. Overall, it seems like the model is performing fairly well.

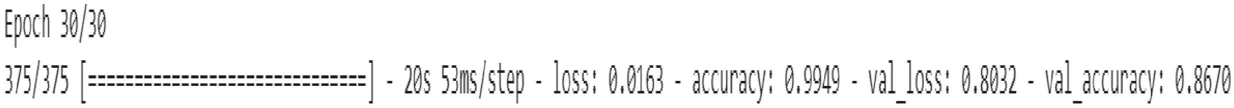


```
Epoch 50/50
188/188 [==============================] - 13s 66ms/step - loss: 0.4556 - accuracy: 0.9757 - val_loss: 0.7329 - val_accuracy: 0.9690
```

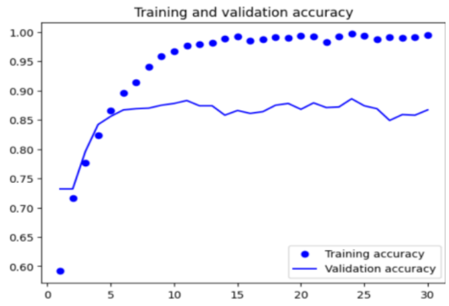


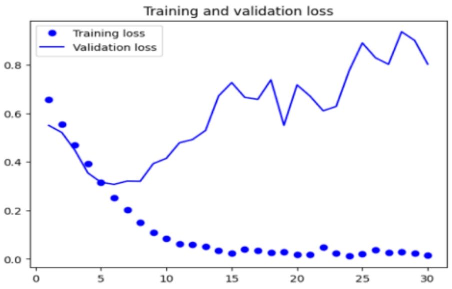
By the end of the 50th epoch, the model had a training accuracy of 0.9757 and a training loss of 0.4556. Validation accuracy was 0.9690, with a 0.7329 validation loss. These findings imply that the model is performing well in terms of generalization to the validation data and has learned to categorize the training data with an extremely high degree of accuracy. Performance of the model seems to be excellent overall.

EXAMINING THE SAMPLE OF 7000: The following are the outcomes and comparisions.



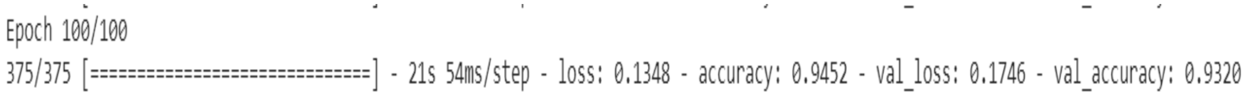
```
Epoch 30/30
375/375 [==============================] - 20s 53ms/step - loss: 0.0163 - accuracy: 0.9949 - val_loss: 0.8032 - val_accuracy: 0.8670
```



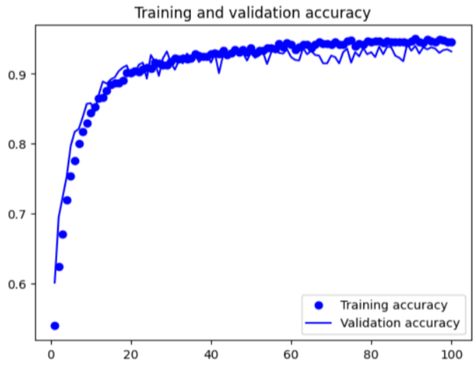


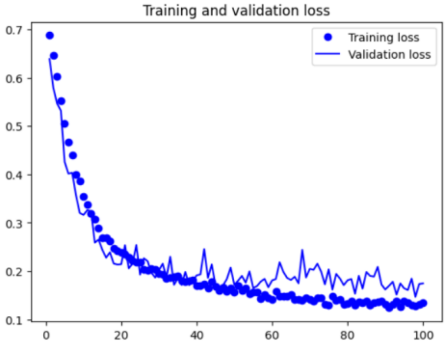
At the end of the 30th epoch, the model achieved a training loss of 0.0163 and a training accuracy of 0.9949. The validation loss was 0.8032 and the validation accuracy was 0.8670.

These results suggest that the model has learned to classify the training data with high accuracy, but is not generalizing as well to the validation data. Further improvements may be necessary to improve the model's performance on the validation dataset.

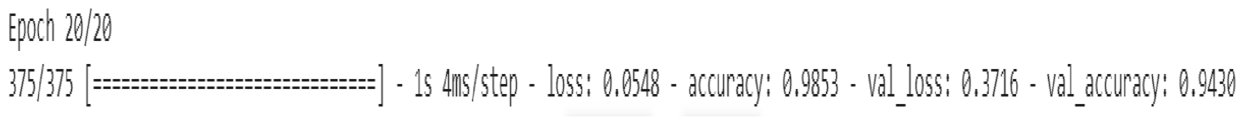


```
Epoch 100/100
375/375 [==============================] - 21s 54ms/step - loss: 0.1348 - accuracy: 0.9452 - val_loss: 0.1746 - val_accuracy: 0.9320
```

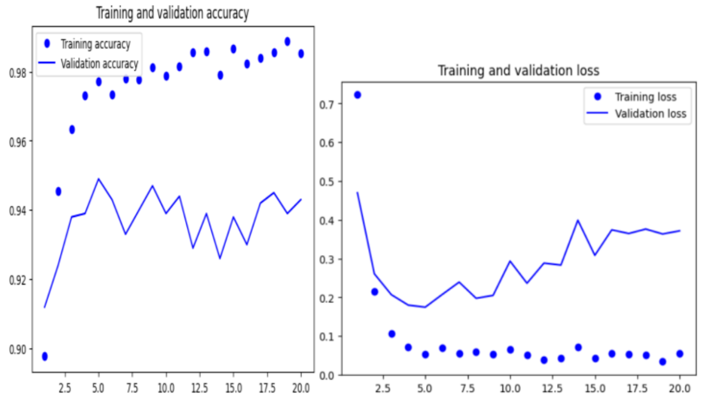




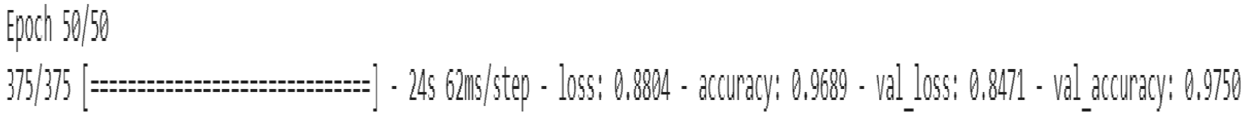
The model finished the 100th epoch with a training loss of 0.1348 and an accuracy score of 0.9452. The accuracy of the validation was 0.9320, while the validation loss was 0.1746. The model appears to have learnt to classify the training data with some accuracy, according to these results, but there is still potential for improvement given that the validation accuracy is lower. The model is not overfitting to the training data, as seen by the comparatively low validation loss.



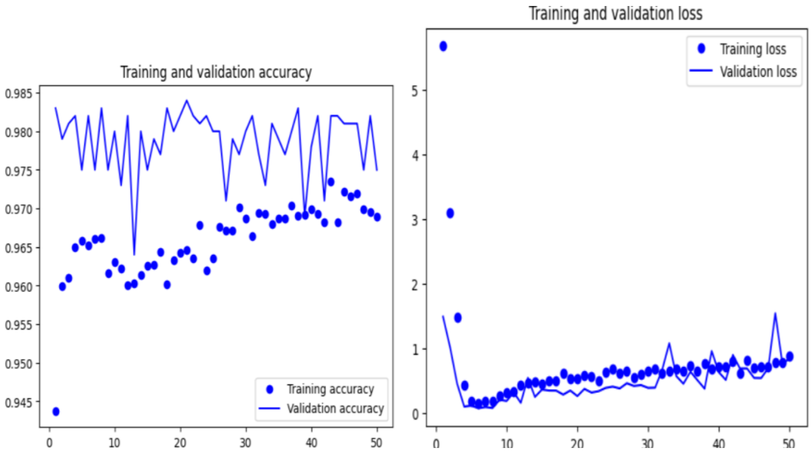
```
Epoch 20/20
375/375 [==============================] - 1s 4ms/step - loss: 0.0548 - accuracy: 0.9853 - val_loss: 0.3716 - val_accuracy: 0.9430
```



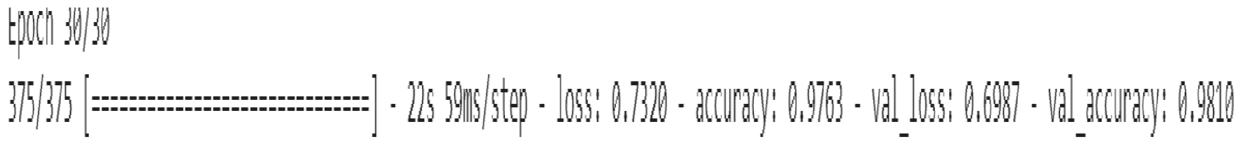
The model attained a training accuracy of 0.9853 and a training loss of 0.0548 by the conclusion of the 20th epoch. The accuracy of the validation was 0.9430, and the validation loss was 0.3716. These findings imply that the model has mastered the art of accurately classifying training data, and the validation accuracy is likewise rather good. Additionally, the validation loss is modest, demonstrating that the model is not overfit to the training set of data.



```
Epoch 50/50
375/375 [==============================] - 24s 62ms/step - loss: 0.8804 - accuracy: 0.9689 - val_loss: 0.8471 - val_accuracy: 0.9750
```

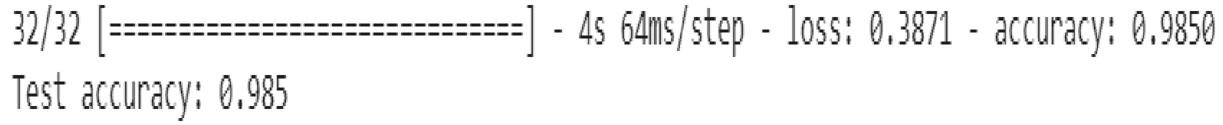


The validation loss was 0.85 and the validation accuracy was 0.97 in the most recent epoch, whereas the training loss was 0.88 and the training accuracy was 0.97. This shows that the model is functioning effectively and generalizing to new data with ease. While the validation loss and accuracy show how well the model generalizes to new data, the training loss and accuracy show how well the model performs on the training dataset.



```
Epoch 30/30
375/375 [==============================] - 22s 59ms/step - loss: 0.7320 - accuracy: 0.9763 - val_loss: 0.6987 - val_accuracy: 0.9810
```

The model achieved a validation accuracy of 98.1% after 30 epochs with a validation loss of 0.6987.



```
32/32 [==============================] - 4s 64ms/step - loss: 0.3871 - accuracy: 0.9850
Test accuracy: 0.985
```

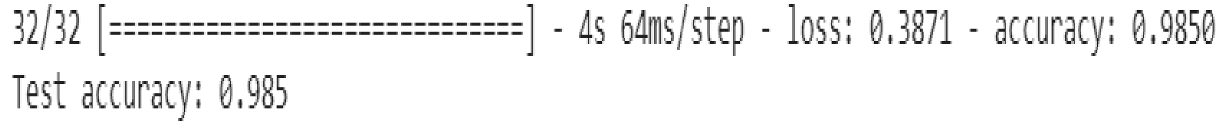
The test accuracy of the model is 0.985.

Here are also used the techniques like Optimization, Dropouts, Data Augmentation.

Finding the optimal collection of weights for a model to minimize a specific loss function is referred to as optimization. Gradient descent is a common method for accomplishing this, in which the parameters of the model are updated in the direction of steepest descent after calculating the gradient of the loss with respect to the parameters of the model.

A specific percentage of the nodes in a neural network are randomly dropped out (or set to zero) as part of the regularization process known as dropouts. By making the network learn more reliable properties that are independent of particular nodes, this helps prevent overfitting.

By applying random modifications, such flipping, rotating, and scaling, to existing data, data augmentation entails producing new training data. This broadens the training set's diversity and improves the model's ability to generalize to brand-new, unexplored data.



```
32/32 [==============================] - 4s 64ms/step - loss: 0.3871 - accuracy: 0.9850
Test accuracy: 0.985
```

The test accuracy of the model is 0.985.