

Pneumonia Image Classification

Business Understanding:

In this image classification project, I am utilizing neural networks - primarily convolutional neural networks - to create a model that can identify whether or not a patient has pneumonia by analyzing their lung x-rays.

There are two types of pneumonia - bacterial and viral. The image classification system will have to be able to pick out both, while not necessarily knowing which one is which, as the types of pneumonia are not labeled in the dataset.

Data Understanding:

The dataset is organized into 3 folders (train, test, val) and contains subfolders for each image category (Pneumonia/Normal). There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal).

Chest X-ray images (anterior-posterior) were selected from retrospective cohorts of pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou. All chest X-ray imaging was performed as part of patients' routine clinical care.

Due to the relatively small amount of validation data (16 images), as well as test data, I will create my own validation data instead of using the provided split.

```

In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import keras
from keras.models import Sequential
from keras.layers import Dense, Conv2D, BatchNormalization, MaxPooling2D, Dropout, Flatten
from sklearn.preprocessing import StandardScaler, LabelBinarizer
import os
os.environ['KMP_DUPLICATE_LIB_OK']='True'
import tensorflow as tf
from keras import models
from keras import layers
import pathlib
import PIL
import seaborn as sns
import time
import scipy
import numpy as np
from PIL import Image
from scipy import ndimage
from sklearn.model_selection import train_test_split, cross_val_score
from keras.preprocessing.image import ImageDataGenerator, array_to_img, img_to_array, load_img
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.preprocessing import OneHotEncoder
from sklearn.metrics import accuracy_score, confusion_matrix
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import numpy as np
import seaborn as sns
from tensorflow.keras.models import Sequential
from tensorflow.keras.regularizers import l2
from tensorflow.keras.optimizers import SGD
from tensorflow.keras.wrappers import scikit_learn
from tensorflow.keras.callbacks import EarlyStopping
import shutil
import random
import glob
import os
import sys
import itertools
import warnings
import statistics

```

Check Data

```
In [2]: train_norm_size = len(os.listdir('re-split_data/train/normal'))
train_pneum_size = len(os.listdir('re-split_data/train/pneumonia'))
test_norm_size = len(os.listdir('re-split_data/test/normal'))
test_pneum_size = len(os.listdir('re-split_data/test/pneumonia'))
valid_norm_size = len(os.listdir('re-split_data/validation/normal'))
valid_pneum_size = len(os.listdir('re-split_data/validation/pneumonia'))

train_size = train_norm_size + train_pneum_size - 1
test_size = test_norm_size + test_pneum_size - 1
validation_size = valid_norm_size + valid_pneum_size

print(f' There are {train_size} images in the training set, {test_size} in
the test set, and {validation_size} in the validation set')

print(f' train norm is {train_norm_size}')
print(f' train pneum is {train_pneum_size}')
print(f' test norm: {test_norm_size}')
print(f' test pneum: {test_pneum_size}')
print(f' valid norm: {valid_norm_size}')
print(f' valid pneum: {valid_pneum_size}')
```

```
There are 3271 images in the training set, 1751 in the test set, and 816 i
n the validation set
train norm is 882
train pneum is 2390
test norm: 473
test pneum: 1279
valid norm: 220
valid pneum: 596
```

Add Functions

```

In [3]: # Define Result Saving Initial Function
dfcols = ['model_name', 'Train Accuracy', 'Test Accuracy', 'CV1', 'CV2', 'CV3', 'CV4', 'CV5', 'CV_Std', 'CV_avg']
model_summary = pd.DataFrame(columns=dfcols)

def save_result(model_name, Train_Accuracy, Test_Accuracy, cv1, cv2, cv3, cv4, cv5):
    global model_summary
    cv_std = statistics.stdev([cv1, cv2, cv3, cv4, cv5])
    cv_avg = (cv1 + cv2 + cv3 + cv4 + cv5) / 5
    row = [(model_name, Train_Accuracy, Test_Accuracy, cv1, cv2, cv3, cv4, cv5, cv_std, cv_avg)]
    res = pd.DataFrame(columns = dfcols, data = row)
    yeep = [model_summary, res]
    model_summary = pd.concat(yeep)
    model_summary = model_summary.sort_values('CV_avg', ascending = False)
    model_summary = model_summary.drop_duplicates()
    return model_summary.round(3)

```

```

In [132]: # SOURCE: The origin of this confusion matrix code was found on medium, '
# from https://medium.com/@dtuk81/confusion-matrix-visualization-fc31e3f30f
ea
def make_confusion_matrix(cf,
                           group_names=None,
                           categories='auto',
                           count=True,
                           percent=True,
                           cbar=True,
                           xyticks=True,
                           xyplotlabel s=True,
                           sum_stats=True,
                           figsize=None,
                           cmap='Blues',
                           title=None):

    # CODE TO GENERATE SUMMARY STATISTICS & TEXT FOR SUMMARY STATS
    if sum_stats:
        #Accuracy is sum of diagonal divided by total observations
        accuracy = np.trace(cf) / float(np.sum(cf))

        #if it is a binary confusion matrix, show some more stats
        if len(cf)==2:
            #Metrics for Binary Confusion Matrices
            a = cf[0,0]
            b = cf[0,1]
            c = cf[1,0]
            d = cf[1,1]
            tn = ((a / (a+b))*100).round(2).astype(str) + '%'
            fp = ((b / (a+b))*100).round(2).astype(str) + '%'
            fn = ((c / (c+d))*100).round(2).astype(str) + '%'
            tp = ((d / (c+d))*100).round(2).astype(str) + '%'
            precision = cf[1,1] / sum(cf[:,1])
            recall = cf[1,1] / sum(cf[1,:])
            f1_score = 2*precision*recall / (precision + recall)
            stats_text = "\n\nAccuracy={: 0.3f}\nPrecision={: 0.3f}\nRecall =
{: 0.3f}\nF1 Score={: 0.3f}".format(
                accuracy, precision, recall, f1_score)
        else:
            stats_text = "\n\nAccuracy={: 0.3f}".format(accuracy)
    else:
        stats_text = ""

    # CODE TO GENERATE TEXT INSIDE EACH SQUARE
    blanks = ['' for i in range(cf.size)]

    if group_names and len(group_names)==cf.size:
        group_labels = ["{}\n".format(value) for value in group_names]
    else:
        group_labels = blanks

    if count:
        group_counts = ["{0:0.0f}\n".format(value) for value in cf.flatten
()]
    else:

```

```

group_counts = blanks

if percent:
    group_percentages = [tn, fp, fn, tp]
    # old = group_percentages = ["{0:.2%}".format(value) for value in c
    f.flatten()/np.sum(cf)]
else:
    group_percentages = blanks

box_labels = [f"{v1}{v2}{v3}".strip() for v1, v2, v3 in zip(group_labels, group_counts, group_percentages)]
box_labels = np.asarray(box_labels).reshape(cf.shape[0], cf.shape[1])

# SET FIGURE PARAMETERS ACCORDING TO OTHER ARGUMENTS
if figsize==None:
    #Get default figure size if not set
    figsize = plt.rcParams.get('figure.figsize')

if xyticks==False:
    #Do not show categories if xyticks is False
    categories=False

# MAKE THE HEATMAP VISUALIZATION
plt.figure(figsize=figsize)
sns.heatmap(cf, annot=box_labels, fmt="", cmap=cmap, cbar=cbar, xticklabels=
categories, yticklabels=categories)

if xyplotlabels:
    plt.ylabel('True Label', weight = 'bold')
    plt.xlabel('Predicted Label' + stats_text, weight = 'bold')
else:
    plt.xlabel(stats_text)

if title:
    plt.title(title, size = 20, weight = 'bold')

```

Re-Splitting (Hide)

Check re-aggregated images

Due to the issues with the given train/test split, i re-aggregated the images. From here, I will train/test split the data myself.

```
In [4]: '''
        folder = 're-split_data/NORMAL'
        path = folder

        p = os.listdir(path)
        pf = pd.DataFrame(p)

        norm_tot = len(pf)
        print(f' There are {len(pf[0])} images in the normal folder')
        folder = 're-split_data/PNEUMONIA'
        path = folder
        p = os.listdir(path)
        pf = pd.DataFrame(p)
        pneum_tot = len(pf)
        pneum_weight = len(pf) / 5863
        norm_weight = 1 - pneum_weight
        pf
        print(f' There are {len(pf[0])} images in the pneumonia folder')
        print(f' there are {1576 + len(pf[0])} total images in the dataset')
        print(f' The weight of pneumonia is {round(pneum_weight, 2)}')
        print(f' The weight of normal is {round(norm_weight, 2)}')
        '''
```

```
Out[4]: "\nfolder = 're-split_data/NORMAL'\npath = folder\n\np = os.listdir(path)\n\npf = pd.DataFrame(p)\n\n\nnorm_tot = len(pf)\n\nprint(f' There are {len(pf[0])}\nimages in the normal folder')\n\nfolder = 're-split_data/PNEUMONIA'\n\npath = folder\n\np = os.listdir(path)\n\npf = pd.DataFrame(p)\n\n\npneum_tot = len(pf)\n\npneum_weight = len(pf) / 5863\n\nnorm_weight = 1 - pneum_weight\n\npf\n\nprint(f' There are {len(pf[0])} images in the pneumonia folder')\n\nprint(f' there are {1576 + len(pf[0])} total images in the dataset')\n\nprint(f' The weight of pneumonia is {round(pneum_weight, 2)}')\n\nprint(f' The weight of normal is {round(norm_weight, 2)}')\n\n"
```

From here, there is some code which I used to re-split the data.

```
In [5]: # define test and train split percentages
'''
norm_train = norm_tot * .7
norm_test = norm_tot * .3
pneum_train = pneum_tot * .7
pneum_test = pneum_tot * .3
'''

'''
pf = os.listdir('re-split_data/NORMAL')
rand_norm_files = random.sample(pf, int(norm_train))
for file in rand_norm_files:
    shutil.copy('re-split_data/NORMAL/' + file, 're-split_data/train/normal')
'''
```

```
Out[5]: "\npf = os.listdir('re-split_data/NORMAL')\nrand_norm_files = random.sample\n\n(pf, int(norm_train))\n\nfor file in rand_norm_files:\n\n    shutil.copy('re-split_data/NORMAL/' + file, 're-split_data/train/normal')\n\n"
```

```
In [6]: '''
p1 = os.listdir('re-split_data/train/normal')
p1 = pd.DataFrame(p1)

p2 = os.listdir('re-split_data/NORMAL')
p2 = pd.DataFrame(p2)

tester_files = pd.concat([p1[0], p2[0]]).drop_duplicates(keep=False)
'''
```

```
Out[6]: "\np1 = os.listdir('re-split_data/train/normal')\np1 = pd.DataFrame(p1)\n\np2 = os.listdir('re-split_data/NORMAL')\np2 = pd.DataFrame(p2)\n\ntester_files = pd.concat([p1[0], p2[0]]).drop_duplicates(keep=False)\n\n"
```

```
In [7]: #for file in tester_files:
        #shutil.copy('re-split_data/NORMAL/' + file, 're-split_data/test/normal')
        '
```



```
In [8]: '''
# The pneumonia files
pf = os.listdir('re-split_data/PNEUMONIA')

rand_Pfiles = random.sample(pf, int(pneum_train))

for file in rand_Pfiles:
    shutil.copy('re-split_data/PNEUMONIA/' + file, 're-split_data/train/pneumonia')

p3 = os.listdir('re-split_data/train/pneumonia')
p3 = pd.DataFrame(p3)

p4 = os.listdir('re-split_data/PNEUMONIA')
p4 = pd.DataFrame(p4)

tester_p = pd.concat([p3[0], p4[0]]).drop_duplicates(keep=False)

for file in tester_p:
    shutil.copy('re-split_data/PNEUMONIA/' + file, 're-split_data/test/pneumonia')
'''
```

```
Out[8]: "\n# The pneumonia files\npf = os.listdir('re-split_data/PNEUMONIA')\n\nrand_Pfiles = random.sample(pf, int(pneum_train))\n\nfor file in rand_Pfiles:\n    n    shutil.copy('re-split_data/PNEUMONIA/' + file, 're-split_data/train/pneumonia')\n\nnp3 = os.listdir('re-split_data/train/pneumonia')\nnp3 = pd.DataFrame(p3)\n\nnp4 = os.listdir('re-split_data/PNEUMONIA')\nnp4 = pd.DataFrame(p4)\n\ntester_p = pd.concat([p3[0], p4[0]]).drop_duplicates(keep=False)\n\nfor file in tester_p:\n    n    shutil.copy('re-split_data/PNEUMONIA/' + file, 're-split_data/test/pneumonia')\n\n"
```

Validation Files

Make a validation set from the train set

```
In [9]: '''
pf = os.listdir('re-split_data/train/normal')
norm_tot = len(pf)
pf1 = os.listdir('re-split_data/train/pneumonia')
pneum_tot = len(pf1)

print(f' There are {norm_tot} images in the normal training folder and {pneum_tot} in the pneumonia training folder')
'''
```

```
Out[9]: "\n\npf = os.listdir('re-split_data/train/normal')\n\nnorm_tot = len(pf)\n\npf1 = os.listdir('re-split_data/train/pneumonia')\n\npneum_tot = len(pf1)\n\n\nprint(f' There are {norm_tot} images in the normal training folder and {pneum_tot} in the pneumonia training folder')\n"
```

```
In [10]: # I will take 110 images from normal and 298 from pneumonia to create validation set

#rand_sample_norm = random.sample(pf, 110)

#for file in rand_sample_norm:
    #shutil.move('re-split_data/train/normal/' + file, 're-split_data/validation/normal')

#rand_sample_pneum = random.sample(pf1, 298)

#for file in rand_sample_pneum:
    #shutil.move('re-split_data/train/pneumonia/' + file, 're-split_data/validation/pneumonia')

'''
pf2 = os.listdir('re-split_data/validation/normal')
valid_norm_tot = len(pf2)
pf3 = os.listdir('re-split_data/validation/pneumonia')
valid_pneum_tot = len(pf3)

rand_sample_valid_norm = random.sample(pf2, 110)
rand_sample_valid_pneum = random.sample(pf3, 298)

for file in rand_sample_valid_norm:
    shutil.move('re-split_data/validation/normal/' + file, 're-split_data/train/normal')

for file in rand_sample_valid_pneum:
    shutil.move('re-split_data/validation/pneumonia/' + file, 're-split_data/train/pneumonia')

'''
```

```
Out[10]: "\npf2 = os.listdir('re-split_data/validation/normal')\nvalid_norm_tot = len(pf2)\nnpf3 = os.listdir('re-split_data/validation/pneumonia')\nvalid_pneum_tot = len(pf3)\n\nrand_sample_valid_norm = random.sample(pf2, 110)\nrand_sample_valid_pneum = random.sample(pf3, 298)\n\nfor file in rand_sample_valid_norm:\n    shutil.move('re-split_data/validation/normal/' + file, 're-split_data/train/normal')\n\nfor file in rand_sample_valid_pneum:\n    shutil.move('re-split_data/validation/pneumonia/' + file, 're-split_data/train/pneumonia')\n\n\n"
```

Generate Test and Train Images

```
In [11]: # get all the data in the directory split/test , and reshape them
train_generator = ImageDataGenerator(rescale=1./255).flow_from_directory('re-split_data/train',
                                target_size=(64, 64), batch_size = train_size)

test_generator = ImageDataGenerator(rescale=1./255).flow_from_directory('re-split_data/test',
                                target_size=(64, 64), batch_size = test_size, shuffle= False)

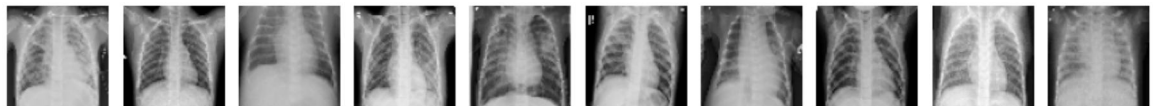
valid_generator = ImageDataGenerator(rescale=1./255).flow_from_directory('re-split_data/validation',
                                target_size=(64, 64), batch_size = validation_size)
```

Found 3272 images belonging to 2 classes.
Found 1752 images belonging to 2 classes.
Found 816 images belonging to 2 classes.

```
In [12]: # create the data sets
train_images, train_labels = next(train_generator)
test_images, test_labels = next(test_generator)
valid_images, valid_labels = next(valid_generator)
```

```
In [13]: def show_images(images):
    fig, axes = plt.subplots(1, 10, figsize=(12,12))
    axes = axes.flatten()
    for img, ax in zip(images, axes):
        ax.imshow(img)
        ax.axis('off')
    plt.tight_layout()
    plt.show()
```

```
In [14]: show_images(train_images)
```



```
In [15]: train_img = train_images.reshape(train_images.shape[0], -1)
test_img = test_images.reshape(test_images.shape[0], -1)
valid_img = valid_images.reshape(valid_images.shape[0], -1)

print(train_img.shape)
print(test_img.shape)
print(valid_img.shape)
```

(3271, 12288)
(1751, 12288)
(816, 12288)

```
In [16]: train_y = np.reshape(train_labels[:,0], (train_size,1))
test_y = np.reshape(test_labels[:,0], (test_size,1))
valid_y = np.reshape(valid_labels[:,0], (validation_size,1))

print(train_y.shape)
print(test_y.shape)
print(valid_y.shape)

(3271, 1)
(1751, 1)
(816, 1)
```

Baseline Model

```
In [17]: # Build a baseline fully connected model
model = models.Sequential()
model.add(layers.Dense(20, activation='relu', input_shape=(12288,))) # 2 hidden layers
model.add(layers.Dense(7, activation='relu'))
model.add(layers.Dense(5, activation='relu'))
model.add(layers.Dense(1, activation='sigmoid'))
```

```
In [18]: model.compile(optimizer='sgd',
                        loss='binary_crossentropy',
                        metrics=['accuracy'])

baseline = model.fit(train_img, train_y, epochs=15, batch_size=32)

train_loss = baseline.history['loss']
train_acc = baseline.history['accuracy']

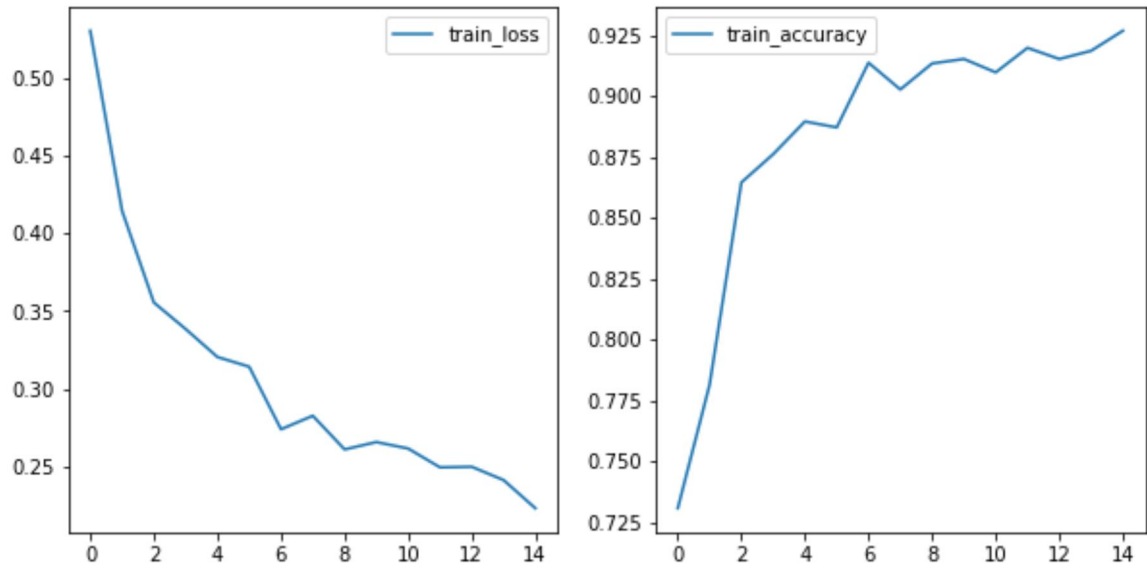
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=baseline.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=baseline.epoch, y=train_acc, ax=ax2, label='train_accuracy')
```

```

Epoch 1/15
103/103 [=====] - 1s 3ms/step - loss: 0.5303 - accuracy: 0.7307
Epoch 2/15
103/103 [=====] - 2s 15ms/step - loss: 0.4145 - accuracy: 0.7814
Epoch 3/15
103/103 [=====] - 2s 18ms/step - loss: 0.3554 - accuracy: 0.8646
Epoch 4/15
103/103 [=====] - 2s 20ms/step - loss: 0.3384 - accuracy: 0.8762
Epoch 5/15
103/103 [=====] - 2s 20ms/step - loss: 0.3204 - accuracy: 0.8896
Epoch 6/15
103/103 [=====] - 2s 21ms/step - loss: 0.3141 - accuracy: 0.8872
Epoch 7/15
103/103 [=====] - 0s 3ms/step - loss: 0.2739 - accuracy: 0.9138
Epoch 8/15
103/103 [=====] - 1s 13ms/step - loss: 0.2825 - accuracy: 0.9028
0s - loss: 0.2740 - accuracy: 0.9153
Epoch 9/15
103/103 [=====] - 3s 29ms/step - loss: 0.2609 - accuracy: 0.9135
Epoch 10/15
103/103 [=====] - 3s 28ms/step - loss: 0.2656 - accuracy: 0.9153
Epoch 11/15
103/103 [=====] - 2s 21ms/step - loss: 0.2615 - accuracy: 0.9098
Epoch 12/15
103/103 [=====] - 2s 23ms/step - loss: 0.2494 - accuracy: 0.9199
Epoch 13/15
103/103 [=====] - 2s 18ms/step - loss: 0.2497 - accuracy: 0.9153
Epoch 14/15
103/103 [=====] - 3s 30ms/step - loss: 0.2412 - accuracy: 0.9187
Epoch 15/15
103/103 [=====] - 3s 26ms/step - loss: 0.2230 - accuracy: 0.9269

```

Out[18]: <AxesSubplot: >



SkLearn Version for Cross-Validation

```
In [19]: # Build function that builds the model so we can evaluate in sklearn
def build_model():
    model.add(layers.Dense(20, activation='relu', input_shape=(12288,))) #
    2 hidden layers
    model.add(layers.Dense(7, activation='relu'))
    model.add(layers.Dense(5, activation='relu'))
    model.add(layers.Dense(1, activation='sigmoid'))

    model.compile(optimizer='sgd',
                  loss='binary_crossentropy',
                  metrics=['accuracy'])

    return model
```

```
In [20]: keras_model = sklearn.KerasClassifier(build_model,
                                                epochs=15,
                                                batch_size=32,
                                                verbose=2)
```

Cross-Validation

```
In [21]: # Now that it is a keras model, you can cross-validate it  
cvs = cross_val_score(keras_model, train_img, train_y, cv=5)
```


Epoch 1/15
82/82 - 1s - loss: 0.6411 - accuracy: 0.8356
Epoch 2/15
82/82 - 0s - loss: 0.5714 - accuracy: 0.7255
Epoch 3/15
82/82 - 2s - loss: 0.5260 - accuracy: 0.7255
Epoch 4/15
82/82 - 2s - loss: 0.4847 - accuracy: 0.7255
Epoch 5/15
82/82 - 3s - loss: 0.4433 - accuracy: 0.7255
Epoch 6/15
82/82 - 1s - loss: 0.3987 - accuracy: 0.8402
Epoch 7/15
82/82 - 0s - loss: 0.3577 - accuracy: 0.9373
Epoch 8/15
82/82 - 0s - loss: 0.3210 - accuracy: 0.9362
Epoch 9/15
82/82 - 0s - loss: 0.2865 - accuracy: 0.9350
Epoch 10/15
82/82 - 0s - loss: 0.2727 - accuracy: 0.9205
Epoch 11/15
82/82 - 0s - loss: 0.2415 - accuracy: 0.9346
Epoch 12/15
82/82 - 1s - loss: 0.2358 - accuracy: 0.9297
Epoch 13/15
82/82 - 2s - loss: 0.2344 - accuracy: 0.9201
Epoch 14/15
82/82 - 1s - loss: 0.2304 - accuracy: 0.9220
Epoch 15/15
82/82 - 2s - loss: 0.2061 - accuracy: 0.9262
21/21 - 0s - loss: 0.1962 - accuracy: 0.9374
Epoch 1/15
82/82 - 1s - loss: 0.6662 - accuracy: 0.7298
Epoch 2/15
82/82 - 0s - loss: 0.6184 - accuracy: 0.7298
Epoch 3/15
82/82 - 0s - loss: 0.5971 - accuracy: 0.7298
Epoch 4/15
82/82 - 0s - loss: 0.5885 - accuracy: 0.7298
Epoch 5/15
82/82 - 0s - loss: 0.5854 - accuracy: 0.7298
Epoch 6/15
82/82 - 0s - loss: 0.5843 - accuracy: 0.7298
Epoch 7/15
82/82 - 0s - loss: 0.5838 - accuracy: 0.7298
Epoch 8/15
82/82 - 3s - loss: 0.5837 - accuracy: 0.7298
Epoch 9/15
82/82 - 3s - loss: 0.5836 - accuracy: 0.7298
Epoch 10/15
82/82 - 1s - loss: 0.5836 - accuracy: 0.7298
Epoch 11/15
82/82 - 1s - loss: 0.5836 - accuracy: 0.7298
Epoch 12/15
82/82 - 1s - loss: 0.5835 - accuracy: 0.7298
Epoch 13/15

82/82 - 2s - loss: 0.5835 - accuracy: 0.7298
Epoch 14/15
82/82 - 2s - loss: 0.5835 - accuracy: 0.7298
Epoch 15/15
82/82 - 2s - loss: 0.5835 - accuracy: 0.7298
21/21 - 1s - loss: 0.5809 - accuracy: 0.7324
Epoch 1/15
82/82 - 2s - loss: 0.6616 - accuracy: 0.7291
Epoch 2/15
82/82 - 2s - loss: 0.6244 - accuracy: 0.7291
Epoch 3/15
82/82 - 2s - loss: 0.6043 - accuracy: 0.7291
Epoch 4/15
82/82 - 2s - loss: 0.5938 - accuracy: 0.7291
Epoch 5/15
82/82 - 2s - loss: 0.5886 - accuracy: 0.7291
Epoch 6/15
82/82 - 1s - loss: 0.5862 - accuracy: 0.7291
Epoch 7/15
82/82 - 2s - loss: 0.5851 - accuracy: 0.7291
Epoch 8/15
82/82 - 1s - loss: 0.5846 - accuracy: 0.7291
Epoch 9/15
82/82 - 2s - loss: 0.5844 - accuracy: 0.7291
Epoch 10/15
82/82 - 1s - loss: 0.5843 - accuracy: 0.7291
Epoch 11/15
82/82 - 1s - loss: 0.5843 - accuracy: 0.7291
Epoch 12/15
82/82 - 1s - loss: 0.5843 - accuracy: 0.7291
Epoch 13/15
82/82 - 1s - loss: 0.5843 - accuracy: 0.7291
Epoch 14/15
82/82 - 2s - loss: 0.5842 - accuracy: 0.7291
Epoch 15/15
82/82 - 2s - loss: 0.5842 - accuracy: 0.7291
21/21 - 1s - loss: 0.5779 - accuracy: 0.7355
Epoch 1/15
82/82 - 1s - loss: 0.6736 - accuracy: 0.7218
Epoch 2/15
82/82 - 0s - loss: 0.6424 - accuracy: 0.7356
Epoch 3/15
82/82 - 0s - loss: 0.6219 - accuracy: 0.7356
Epoch 4/15
82/82 - 0s - loss: 0.6081 - accuracy: 0.7356
Epoch 5/15
82/82 - 0s - loss: 0.5987 - accuracy: 0.7356
Epoch 6/15
82/82 - 0s - loss: 0.5924 - accuracy: 0.7356
Epoch 7/15
82/82 - 2s - loss: 0.5880 - accuracy: 0.7356
Epoch 8/15
82/82 - 2s - loss: 0.5849 - accuracy: 0.7356
Epoch 9/15
82/82 - 2s - loss: 0.5828 - accuracy: 0.7356
Epoch 10/15

```

82/82 - 1s - loss: 0.5813 - accuracy: 0.7356
Epoch 11/15
82/82 - 1s - loss: 0.5803 - accuracy: 0.7356
Epoch 12/15
82/82 - 2s - loss: 0.5795 - accuracy: 0.7356
Epoch 13/15
82/82 - 2s - loss: 0.5790 - accuracy: 0.7356
Epoch 14/15
82/82 - 1s - loss: 0.5786 - accuracy: 0.7356
Epoch 15/15
82/82 - 2s - loss: 0.5784 - accuracy: 0.7356
21/21 - 0s - loss: 0.6029 - accuracy: 0.7095
Epoch 1/15
82/82 - 1s - loss: 0.6297 - accuracy: 0.7318
Epoch 2/15
82/82 - 0s - loss: 0.5899 - accuracy: 0.7318
Epoch 3/15
82/82 - 0s - loss: 0.5831 - accuracy: 0.7318
Epoch 4/15
82/82 - 0s - loss: 0.5818 - accuracy: 0.7318
Epoch 5/15
82/82 - 0s - loss: 0.5817 - accuracy: 0.7318
Epoch 6/15
82/82 - 0s - loss: 0.5817 - accuracy: 0.7318
Epoch 7/15
82/82 - 0s - loss: 0.5817 - accuracy: 0.7318
Epoch 8/15
82/82 - 0s - loss: 0.5817 - accuracy: 0.7318
Epoch 9/15
82/82 - 0s - loss: 0.5816 - accuracy: 0.7318
Epoch 10/15
82/82 - 0s - loss: 0.5816 - accuracy: 0.7318
Epoch 11/15
82/82 - 0s - loss: 0.5816 - accuracy: 0.7318
Epoch 12/15
82/82 - 2s - loss: 0.5817 - accuracy: 0.7318
Epoch 13/15
82/82 - 3s - loss: 0.5816 - accuracy: 0.7318
Epoch 14/15
82/82 - 3s - loss: 0.5817 - accuracy: 0.7318
Epoch 15/15
82/82 - 3s - loss: 0.5816 - accuracy: 0.7318
21/21 - 0s - loss: 0.5885 - accuracy: 0.7248

```

Results

```
In [22]: results_train = model.evaluate(train_img, train_y)
```

```

103/103 [=====] - 2s 19ms/step - loss: 0.5829 - ac
curacy: 0.7304

```

```
In [23]: results_test = model.evaluate(test_img, test_y)
```

```
55/55 [=====] - 1s 26ms/step - loss: 0.5834 - accuracy: 0.7299
```

```
In [24]: save_result('Initial Model', results_train[1], results_test[1], cvs[0], cvs[1], cvs[2], cvs[3], cvs[4])
```

Out[24]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	Initial Model	0.73	0.73	0.937	0.732	0.735	0.709	0.725	0.095	0.768

Convolutional Neural Network (CNN)

```
In [25]: model = models.Sequential()

model.add(layers.Conv2D(32, (3, 3), activation='relu',
                        input_shape=(64, 64, 3)))
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(32, (4, 4), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(1, activation='sigmoid'))
```

In [26]: `model.summary()`

Model: "sequential_1"

Layer (type)	Output Shape	Param #
=====		
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 28, 28, 32)	16416
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
conv2d_2 (Conv2D)	(None, 12, 12, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 64)	0
flatten (Flatten)	(None, 2304)	0
dense_24 (Dense)	(None, 64)	147520
dense_25 (Dense)	(None, 1)	65
=====		
Total params: 183,393		
Trainable params: 183,393		
Non-trainable params: 0		

In [27]: `model.compile(loss='binary_crossentropy',
optimizer="sgd",
metrics=['accuracy'])`

Train Initial Simple CNN

[illegible]

Epoch 1/25
103/103 [=====] - 5s 40ms/step - loss: 0.5833 - accuracy: 0.7307 - val_loss: 0.5686 - val_accuracy: 0.7304

Epoch 2/25
103/103 [=====] - 4s 41ms/step - loss: 0.5507 - accuracy: 0.7307 - val_loss: 0.5088 - val_accuracy: 0.7304

Epoch 3/25
103/103 [=====] - 4s 39ms/step - loss: 0.4919 - accuracy: 0.7615 - val_loss: 0.7125 - val_accuracy: 0.7304

Epoch 4/25
103/103 [=====] - 4s 39ms/step - loss: 0.4252 - accuracy: 0.8178 - val_loss: 0.4001 - val_accuracy: 0.7733

Epoch 5/25
103/103 [=====] - 4s 39ms/step - loss: 0.3494 - accuracy: 0.8563 - val_loss: 0.2769 - val_accuracy: 0.8787

Epoch 6/25
103/103 [=====] - 4s 39ms/step - loss: 0.2945 - accuracy: 0.8731 - val_loss: 0.2413 - val_accuracy: 0.9044

Epoch 7/25
103/103 [=====] - 4s 40ms/step - loss: 0.2560 - accuracy: 0.8912 - val_loss: 0.2123 - val_accuracy: 0.9093

Epoch 8/25
103/103 [=====] - 4s 38ms/step - loss: 0.2259 - accuracy: 0.9123 - val_loss: 0.2706 - val_accuracy: 0.8836

Epoch 9/25
103/103 [=====] - 4s 38ms/step - loss: 0.2263 - accuracy: 0.9074 - val_loss: 0.3273 - val_accuracy: 0.8517

Epoch 10/25
103/103 [=====] - 4s 38ms/step - loss: 0.2094 - accuracy: 0.9116 - val_loss: 0.1702 - val_accuracy: 0.9314

Epoch 11/25
103/103 [=====] - 4s 38ms/step - loss: 0.1955 - accuracy: 0.9214 - val_loss: 0.1664 - val_accuracy: 0.9363

Epoch 12/25
103/103 [=====] - 4s 40ms/step - loss: 0.1943 - accuracy: 0.9227 - val_loss: 0.2165 - val_accuracy: 0.9130

Epoch 13/25
103/103 [=====] - 4s 38ms/step - loss: 0.1833 - accuracy: 0.9275 - val_loss: 0.1596 - val_accuracy: 0.9375

Epoch 14/25
103/103 [=====] - 4s 38ms/step - loss: 0.1813 - accuracy: 0.9294 - val_loss: 0.1648 - val_accuracy: 0.9400

Epoch 15/25
103/103 [=====] - 4s 38ms/step - loss: 0.1727 - accuracy: 0.9327 - val_loss: 0.2302 - val_accuracy: 0.8995

Epoch 16/25
103/103 [=====] - 4s 38ms/step - loss: 0.1760 - accuracy: 0.9321 - val_loss: 0.1445 - val_accuracy: 0.9498

Epoch 17/25
103/103 [=====] - 4s 38ms/step - loss: 0.1683 - accuracy: 0.9355 - val_loss: 0.2540 - val_accuracy: 0.8934

Epoch 18/25
103/103 [=====] - 4s 40ms/step - loss: 0.1649 - accuracy: 0.9376 - val_loss: 0.1423 - val_accuracy: 0.9522

Epoch 19/25
103/103 [=====] - 4s 38ms/step - loss: 0.1594 - ac

```

curacy: 0.9422 - val_loss: 0.2119 - val_accuracy: 0.9179
Epoch 20/25
103/103 [=====] - 4s 38ms/step - loss: 0.1570 - ac
curacy: 0.9416 - val_loss: 0.1664 - val_accuracy: 0.9326
Epoch 21/25
103/103 [=====] - 4s 38ms/step - loss: 0.1518 - ac
curacy: 0.9431 - val_loss: 0.1826 - val_accuracy: 0.9326
Epoch 22/25
103/103 [=====] - 4s 38ms/step - loss: 0.1515 - ac
curacy: 0.9456 - val_loss: 0.1254 - val_accuracy: 0.9596
Epoch 23/25
103/103 [=====] - 4s 40ms/step - loss: 0.1546 - ac
curacy: 0.9428 - val_loss: 0.2358 - val_accuracy: 0.9007
Epoch 24/25
103/103 [=====] - 4s 38ms/step - loss: 0.1482 - ac
curacy: 0.9450 - val_loss: 0.1525 - val_accuracy: 0.9387
Epoch 25/25
103/103 [=====] - 4s 38ms/step - loss: 0.1429 - ac
curacy: 0.9468 - val_loss: 0.1265 - val_accuracy: 0.9583

```

```
In [29]: results_train = model.evaluate(train_images, train_y)
```

```

103/103 [=====] - 1s 11ms/step - loss: 0.1365 - ac
curacy: 0.9468 0s - loss: 0.1292 - accuracy

```

```
In [30]: results_test = model.evaluate(test_images, test_y)
```

```

55/55 [=====] - 1s 10ms/step - loss: 0.1720 - accu
racy: 0.9315

```

Prediction for Confusion Matrix

```
In [125]: predictions = model.predict(x = test_images, steps = 10, verbose=0)
```

```
In [126]: pred_check = np.round(predictions)
```

```
In [127]: pred_check = pred_check[:]
pred_check = pred_check.flatten()
pred_check
```

```
Out[127]: array([0., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [128]: test_check = test_labels[:,0]
test_check
```

```
Out[128]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [129]: cm = confusion_matrix(y_true=test_check, y_pred=pred_check)
```


In [130]: cm

Out[130]: array([[1234, 44],
[76, 397]], dtype=int64)

```
In [37]: def plot_confusion_matrix(cm, classes,  
                                   normalize=False,  
                                   title='Confusion matrix',  
                                   cmap=plt.cm.Blues):  
  
    """  
    This function prints and plots the confusion matrix.  
    Normalization can be applied by setting `normalize=True`.  
    """  
  
    plt.imshow(cm, interpolation='nearest', cmap=cmap)  
    plt.title(title)  
    plt.colorbar()  
    tick_marks = np.arange(len(classes))  
    plt.xticks(tick_marks, classes, rotation=45)  
    plt.yticks(tick_marks, classes)  
  
    if normalize:  
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]  
        print("Normalized confusion matrix")  
    else:  
        print('Confusion matrix, without normalization')  
  
    print(cm)  
  
    thresh = cm.max() / 2.  
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):  
        plt.text(j, i, cm[i, j],  
                 horizontalalignment="center",  
                 color="white" if cm[i, j] > thresh else "black")
```

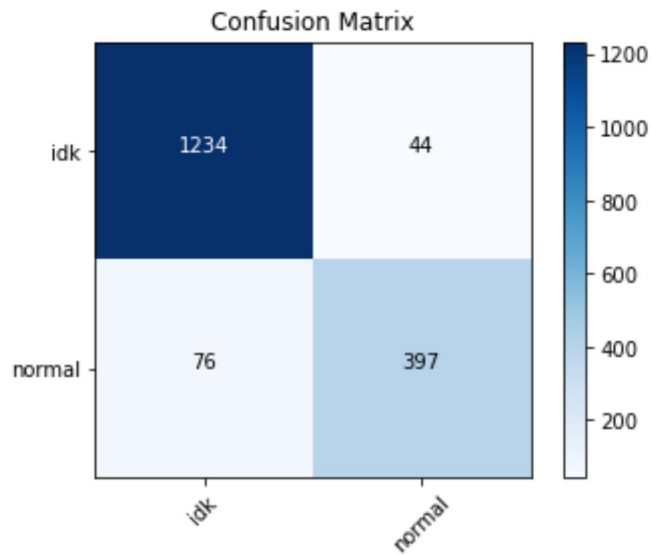
In [38]: {'normal': 0, 'pneumonia': 1}

Out[38]: {'normal': 0, 'pneumonia': 1}

```
In [39]: cm_plot_labels = ['idk', 'normal']  
plot_confusion_matrix(cm=cm, classes=cm_plot_labels, title='Confusion Matrix')
```

Confusion matrix, without normalization

```
[[1234  44]  
 [ 76 397]]
```

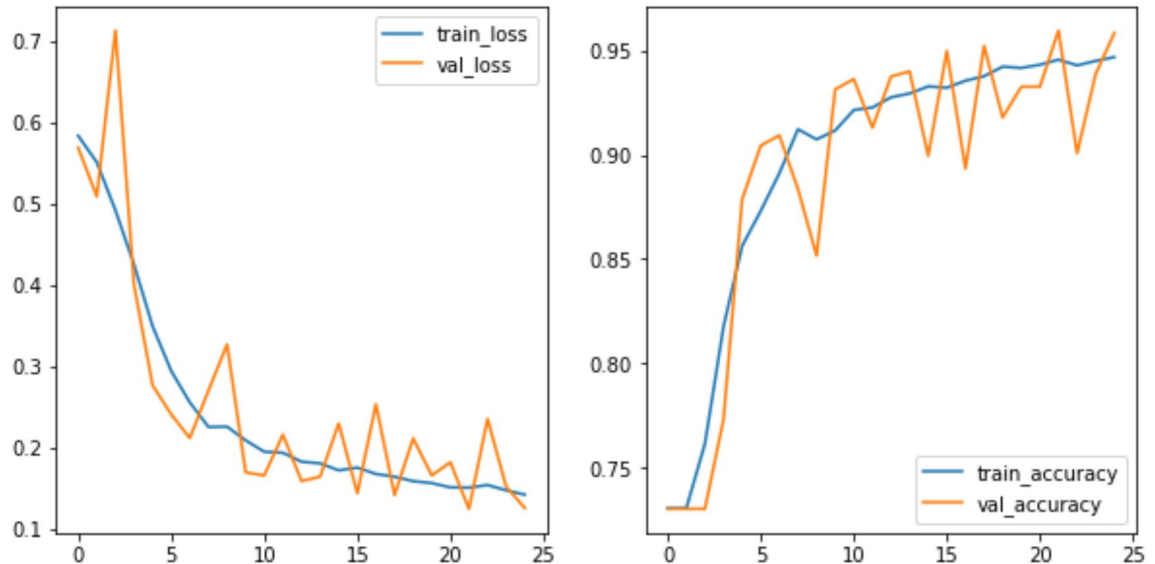


CNN #1 Results

```
In [40]: train_loss = cnn_1.history['loss']
train_acc = cnn_1.history['accuracy']
val_loss = cnn_1.history['val_loss']
val_acc = cnn_1.history['val_accuracy']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=cnn_1.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=cnn_1.epoch, y=train_acc, ax=ax2, label='train_accuracy')
sns.lineplot(x=cnn_1.epoch, y=val_loss, ax=ax1, label='val_loss')
sns.lineplot(x=cnn_1.epoch, y=val_acc, ax=ax2, label='val_accuracy')
```

Out[40]: <AxesSubplot: >



```
In [41]: def build_cnn():
    model = models.Sequential()
    model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
    model.add(layers.MaxPooling2D((2, 2)))

    model.add(layers.Conv2D(32, (4, 4), activation='relu'))
    model.add(layers.MaxPooling2D((2, 2)))

    model.add(layers.Conv2D(64, (3, 3), activation='relu'))
    model.add(layers.MaxPooling2D((2, 2)))

    model.add(layers.Flatten())
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dense(1, activation='sigmoid')) #Last layer must be 1 for binary classification

    model.compile(loss='binary_crossentropy',
                  optimizer='sgd',
                  metrics=['accuracy'])

    return model
```

```
In [42]: keras_model2 = sci ki t_ l earn. KerasCl assi fi er(bui l d_cnn,  
                                                             epochs=25,  
                                                             vali dati on_data=(val i d_i mages,  
val i d_y),  
                                                             vali dati on_steps = vali dati on_s  
i ze)
```

```
In [43]: # Now that it is a keras model, you can cross-validate it  
cvs = cross_val_score(keras_model2, train_images , train_y, cv=5)
```

Epoch 1/25
82/82 [=====] - 4s 46ms/step - loss: 0.5851 - accuracy: 0.7255 - val_loss: 0.5692 - val_accuracy: 0.7304
Epoch 2/25
82/82 [=====] - 3s 41ms/step - loss: 0.5613 - accuracy: 0.7255 - val_loss: 0.5381 - val_accuracy: 0.7304
Epoch 3/25
82/82 [=====] - 3s 40ms/step - loss: 0.5239 - accuracy: 0.7343 - val_loss: 0.4728 - val_accuracy: 0.7304
Epoch 4/25
82/82 [=====] - 3s 40ms/step - loss: 0.4602 - accuracy: 0.7878 - val_loss: 0.5395 - val_accuracy: 0.7402
Epoch 5/25
82/82 [=====] - 3s 40ms/step - loss: 0.4155 - accuracy: 0.8131 - val_loss: 0.3630 - val_accuracy: 0.8113
Epoch 6/25
82/82 [=====] - 3s 40ms/step - loss: 0.3353 - accuracy: 0.8547 - val_loss: 0.2638 - val_accuracy: 0.8922
Epoch 7/25
82/82 [=====] - 3s 40ms/step - loss: 0.3009 - accuracy: 0.8758 - val_loss: 0.2414 - val_accuracy: 0.8983
Epoch 8/25
82/82 [=====] - 3s 40ms/step - loss: 0.2719 - accuracy: 0.8819 - val_loss: 0.2105 - val_accuracy: 0.9118
Epoch 9/25
82/82 [=====] - 3s 40ms/step - loss: 0.2449 - accuracy: 0.9010 - val_loss: 0.1961 - val_accuracy: 0.9203
Epoch 10/25
82/82 [=====] - 3s 41ms/step - loss: 0.2301 - accuracy: 0.9094 - val_loss: 0.2306 - val_accuracy: 0.9044
Epoch 11/25
82/82 [=====] - 3s 40ms/step - loss: 0.2208 - accuracy: 0.9125 - val_loss: 0.2198 - val_accuracy: 0.9056
Epoch 12/25
82/82 [=====] - 3s 40ms/step - loss: 0.2026 - accuracy: 0.9193 - val_loss: 0.1716 - val_accuracy: 0.9301
Epoch 13/25
82/82 [=====] - 4s 45ms/step - loss: 0.2049 - accuracy: 0.9193 - val_loss: 0.2326 - val_accuracy: 0.9020
Epoch 14/25
82/82 [=====] - 3s 41ms/step - loss: 0.1845 - accuracy: 0.9304 - val_loss: 0.1617 - val_accuracy: 0.9387
Epoch 15/25
82/82 [=====] - 3s 41ms/step - loss: 0.1923 - accuracy: 0.9243 - val_loss: 0.1754 - val_accuracy: 0.9375
Epoch 16/25
82/82 [=====] - 3s 40ms/step - loss: 0.1794 - accuracy: 0.9312 - val_loss: 0.1755 - val_accuracy: 0.9314
Epoch 17/25
82/82 [=====] - 3s 41ms/step - loss: 0.1737 - accuracy: 0.9354 - val_loss: 0.1487 - val_accuracy: 0.9412
Epoch 18/25
82/82 [=====] - 3s 40ms/step - loss: 0.1791 - accuracy: 0.9327 - val_loss: 0.1621 - val_accuracy: 0.9424
Epoch 19/25
82/82 [=====] - 3s 43ms/step - loss: 0.1684 - accuracy:

racy: 0.9365 - val_loss: 0.1413 - val_accuracy: 0.9485
Epoch 20/25
82/82 [=====] - 3s 41ms/step - loss: 0.1702 - accu
racy: 0.9343 - val_loss: 0.1458 - val_accuracy: 0.9510
Epoch 21/25
82/82 [=====] - 3s 40ms/step - loss: 0.1632 - accu
racy: 0.9385 - val_loss: 0.1381 - val_accuracy: 0.9461
Epoch 22/25
82/82 [=====] - 3s 41ms/step - loss: 0.1643 - accu
racy: 0.9388 - val_loss: 0.1396 - val_accuracy: 0.9485
Epoch 23/25
82/82 [=====] - 3s 40ms/step - loss: 0.1558 - accu
racy: 0.9438 - val_loss: 0.1442 - val_accuracy: 0.9473
Epoch 24/25
82/82 [=====] - 3s 40ms/step - loss: 0.1523 - accu
racy: 0.9453 - val_loss: 0.1945 - val_accuracy: 0.9265
Epoch 25/25
82/82 [=====] - 3s 42ms/step - loss: 0.1533 - accu
racy: 0.9453 - val_loss: 0.1468 - val_accuracy: 0.9473
21/21 [=====] - 0s 10ms/step - loss: 0.1661 - accu
racy: 0.9328
Epoch 1/25
82/82 [=====] - 4s 42ms/step - loss: 0.5883 - accu
racy: 0.7298 - val_loss: 0.5768 - val_accuracy: 0.7304
Epoch 2/25
82/82 [=====] - 3s 40ms/step - loss: 0.5635 - accu
racy: 0.7298 - val_loss: 0.5458 - val_accuracy: 0.7304
Epoch 3/25
82/82 [=====] - 3s 40ms/step - loss: 0.5267 - accu
racy: 0.7352 - val_loss: 0.4869 - val_accuracy: 0.7855
Epoch 4/25
82/82 [=====] - 3s 40ms/step - loss: 0.4751 - accu
racy: 0.7864 - val_loss: 0.4060 - val_accuracy: 0.7708
Epoch 5/25
82/82 [=====] - 3s 41ms/step - loss: 0.4252 - accu
racy: 0.8082 - val_loss: 0.4533 - val_accuracy: 0.8235
Epoch 6/25
82/82 [=====] - 4s 43ms/step - loss: 0.3538 - accu
racy: 0.8567 - val_loss: 0.3078 - val_accuracy: 0.8529
Epoch 7/25
82/82 [=====] - 3s 41ms/step - loss: 0.3166 - accu
racy: 0.8670 - val_loss: 0.2686 - val_accuracy: 0.8824
Epoch 8/25
82/82 [=====] - 3s 41ms/step - loss: 0.2682 - accu
racy: 0.8850 - val_loss: 0.2127 - val_accuracy: 0.9167
Epoch 9/25
82/82 [=====] - 3s 41ms/step - loss: 0.2475 - accu
racy: 0.8964 - val_loss: 0.2887 - val_accuracy: 0.8787
Epoch 10/25
82/82 [=====] - 3s 41ms/step - loss: 0.2325 - accu
racy: 0.9037 - val_loss: 0.2469 - val_accuracy: 0.8934
Epoch 11/25
82/82 [=====] - 3s 41ms/step - loss: 0.2129 - accu
racy: 0.9163 - val_loss: 0.1780 - val_accuracy: 0.9314
Epoch 12/25
82/82 [=====] - 3s 41ms/step - loss: 0.2160 - accu

racy: 0.9136 - val_loss: 0.2329 - val_accuracy: 0.8971
Epoch 13/25
82/82 [=====] - 3s 42ms/step - loss: 0.1994 - accu
racy: 0.9220 - val_loss: 0.2262 - val_accuracy: 0.9044
Epoch 14/25
82/82 [=====] - 3s 41ms/step - loss: 0.1969 - accu
racy: 0.9243 - val_loss: 0.1633 - val_accuracy: 0.9338
Epoch 15/25
82/82 [=====] - 3s 40ms/step - loss: 0.1873 - accu
racy: 0.9305 - val_loss: 0.1612 - val_accuracy: 0.9363
Epoch 16/25
82/82 [=====] - 3s 40ms/step - loss: 0.1817 - accu
racy: 0.9285 - val_loss: 0.1553 - val_accuracy: 0.9412
Epoch 17/25
82/82 [=====] - 3s 41ms/step - loss: 0.1788 - accu
racy: 0.9312 - val_loss: 0.1510 - val_accuracy: 0.9436
Epoch 18/25
82/82 [=====] - 3s 40ms/step - loss: 0.1779 - accu
racy: 0.9289 - val_loss: 0.2068 - val_accuracy: 0.9228
Epoch 19/25
82/82 [=====] - 3s 42ms/step - loss: 0.1736 - accu
racy: 0.9320 - val_loss: 0.1493 - val_accuracy: 0.9461
Epoch 20/25
82/82 [=====] - 3s 41ms/step - loss: 0.1688 - accu
racy: 0.9312 - val_loss: 0.1451 - val_accuracy: 0.9449
Epoch 21/25
82/82 [=====] - 3s 40ms/step - loss: 0.1648 - accu
racy: 0.9366 - val_loss: 0.1377 - val_accuracy: 0.9510
Epoch 22/25
82/82 [=====] - 3s 41ms/step - loss: 0.1636 - accu
racy: 0.9377 - val_loss: 0.1356 - val_accuracy: 0.9547
Epoch 23/25
82/82 [=====] - 3s 40ms/step - loss: 0.1562 - accu
racy: 0.9408 - val_loss: 0.1361 - val_accuracy: 0.9449
Epoch 24/25
82/82 [=====] - 3s 41ms/step - loss: 0.1563 - accu
racy: 0.9400 - val_loss: 0.1299 - val_accuracy: 0.9547
Epoch 25/25
82/82 [=====] - 3s 42ms/step - loss: 0.1529 - accu
racy: 0.9434 - val_loss: 0.1285 - val_accuracy: 0.9571
21/21 [=====] - 0s 10ms/step - loss: 0.1395 - accu
racy: 0.9434
Epoch 1/25
82/82 [=====] - 4s 43ms/step - loss: 0.5873 - accu
racy: 0.7214 - val_loss: 0.5601 - val_accuracy: 0.7304
Epoch 2/25
82/82 [=====] - 3s 41ms/step - loss: 0.5522 - accu
racy: 0.7291 - val_loss: 0.5245 - val_accuracy: 0.7304
Epoch 3/25
82/82 [=====] - 3s 40ms/step - loss: 0.4964 - accu
racy: 0.7493 - val_loss: 0.4597 - val_accuracy: 0.8897
Epoch 4/25
82/82 [=====] - 3s 42ms/step - loss: 0.4400 - accu
racy: 0.8044 - val_loss: 0.3759 - val_accuracy: 0.7953
Epoch 5/25
82/82 [=====] - 3s 41ms/step - loss: 0.3677 - accu

racy: 0.8472 - val_loss: 1.1944 - val_accuracy: 0.3235
Epoch 6/25
82/82 [=====] - 4s 43ms/step - loss: 0.3071 - accu
racy: 0.8651 - val_loss: 0.2368 - val_accuracy: 0.9081
Epoch 7/25
82/82 [=====] - 3s 40ms/step - loss: 0.2608 - accu
racy: 0.8930 - val_loss: 0.2098 - val_accuracy: 0.9179
Epoch 8/25
82/82 [=====] - 3s 41ms/step - loss: 0.2456 - accu
racy: 0.8922 - val_loss: 0.1981 - val_accuracy: 0.9265
Epoch 9/25
82/82 [=====] - 3s 40ms/step - loss: 0.2205 - accu
racy: 0.9075 - val_loss: 0.2203 - val_accuracy: 0.9032
Epoch 10/25
82/82 [=====] - 3s 40ms/step - loss: 0.2074 - accu
racy: 0.9171 - val_loss: 0.1840 - val_accuracy: 0.9314
Epoch 11/25
82/82 [=====] - 3s 41ms/step - loss: 0.1984 - accu
racy: 0.9159 - val_loss: 0.1711 - val_accuracy: 0.9375
Epoch 12/25
82/82 [=====] - 3s 42ms/step - loss: 0.1995 - accu
racy: 0.9213 - val_loss: 0.1914 - val_accuracy: 0.9301
Epoch 13/25
82/82 [=====] - 3s 41ms/step - loss: 0.1868 - accu
racy: 0.9266 - val_loss: 0.1770 - val_accuracy: 0.9400
Epoch 14/25
82/82 [=====] - 3s 41ms/step - loss: 0.1808 - accu
racy: 0.9251 - val_loss: 0.1631 - val_accuracy: 0.9387
Epoch 15/25
82/82 [=====] - 3s 40ms/step - loss: 0.1791 - accu
racy: 0.9293 - val_loss: 0.1598 - val_accuracy: 0.9424
Epoch 16/25
82/82 [=====] - 3s 41ms/step - loss: 0.1690 - accu
racy: 0.9327 - val_loss: 0.1536 - val_accuracy: 0.9424
Epoch 17/25
82/82 [=====] - 3s 40ms/step - loss: 0.1730 - accu
racy: 0.9316 - val_loss: 0.1581 - val_accuracy: 0.9449
Epoch 18/25
82/82 [=====] - 3s 42ms/step - loss: 0.1673 - accu
racy: 0.9324 - val_loss: 0.1589 - val_accuracy: 0.9498
Epoch 19/25
82/82 [=====] - 3s 41ms/step - loss: 0.1654 - accu
racy: 0.9301 - val_loss: 0.1463 - val_accuracy: 0.9498
Epoch 20/25
82/82 [=====] - 3s 41ms/step - loss: 0.1595 - accu
racy: 0.9366 - val_loss: 0.1455 - val_accuracy: 0.9522
Epoch 21/25
82/82 [=====] - 3s 40ms/step - loss: 0.1566 - accu
racy: 0.9412 - val_loss: 0.1431 - val_accuracy: 0.9498
Epoch 22/25
82/82 [=====] - 3s 41ms/step - loss: 0.1578 - accu
racy: 0.9381 - val_loss: 0.1497 - val_accuracy: 0.9498
Epoch 23/25
82/82 [=====] - 3s 40ms/step - loss: 0.1538 - accu
racy: 0.9400 - val_loss: 0.1535 - val_accuracy: 0.9436
Epoch 24/25

82/82 [=====] - 3s 40ms/step - loss: 0.1539 - accuracy: 0.9412 - val_loss: 0.1372 - val_accuracy: 0.9571
Epoch 25/25
82/82 [=====] - 3s 42ms/step - loss: 0.1471 - accuracy: 0.9415 - val_loss: 0.1320 - val_accuracy: 0.9559
21/21 [=====] - 0s 10ms/step - loss: 0.1634 - accuracy: 0.9343
Epoch 1/25
82/82 [=====] - 4s 45ms/step - loss: 0.6022 - accuracy: 0.7356 - val_loss: 0.5850 - val_accuracy: 0.7304
Epoch 2/25
82/82 [=====] - 3s 42ms/step - loss: 0.5746 - accuracy: 0.7356 - val_loss: 0.5752 - val_accuracy: 0.7304
Epoch 3/25
82/82 [=====] - 3s 40ms/step - loss: 0.5621 - accuracy: 0.7356 - val_loss: 0.5540 - val_accuracy: 0.7304
Epoch 4/25
82/82 [=====] - 3s 41ms/step - loss: 0.5352 - accuracy: 0.7367 - val_loss: 0.5097 - val_accuracy: 0.7304
Epoch 5/25
82/82 [=====] - 3s 41ms/step - loss: 0.4902 - accuracy: 0.7707 - val_loss: 0.4276 - val_accuracy: 0.7488
Epoch 6/25
82/82 [=====] - 3s 40ms/step - loss: 0.4672 - accuracy: 0.7753 - val_loss: 0.3998 - val_accuracy: 0.8885
Epoch 7/25
82/82 [=====] - 3s 40ms/step - loss: 0.4011 - accuracy: 0.8292 - val_loss: 0.4613 - val_accuracy: 0.7978
Epoch 8/25
82/82 [=====] - 3s 42ms/step - loss: 0.3409 - accuracy: 0.8559 - val_loss: 0.2591 - val_accuracy: 0.9069
Epoch 9/25
82/82 [=====] - 3s 40ms/step - loss: 0.3042 - accuracy: 0.8674 - val_loss: 0.2539 - val_accuracy: 0.8909
Epoch 10/25
82/82 [=====] - 3s 40ms/step - loss: 0.2562 - accuracy: 0.8938 - val_loss: 0.2275 - val_accuracy: 0.9056
Epoch 11/25
82/82 [=====] - 3s 40ms/step - loss: 0.2521 - accuracy: 0.8938 - val_loss: 0.2122 - val_accuracy: 0.9118
Epoch 12/25
82/82 [=====] - 3s 40ms/step - loss: 0.2219 - accuracy: 0.9113 - val_loss: 0.1906 - val_accuracy: 0.9289
Epoch 13/25
82/82 [=====] - 3s 40ms/step - loss: 0.2258 - accuracy: 0.9121 - val_loss: 0.1786 - val_accuracy: 0.9350
Epoch 14/25
82/82 [=====] - 3s 43ms/step - loss: 0.1940 - accuracy: 0.9251 - val_loss: 0.1754 - val_accuracy: 0.9338
Epoch 15/25
82/82 [=====] - 3s 40ms/step - loss: 0.2000 - accuracy: 0.9198 - val_loss: 0.1642 - val_accuracy: 0.9363
Epoch 16/25
82/82 [=====] - 3s 40ms/step - loss: 0.1928 - accuracy: 0.9228 - val_loss: 0.1719 - val_accuracy: 0.9363
Epoch 17/25

82/82 [=====] - 3s 40ms/step - loss: 0.1890 - accuracy: 0.9266 - val_loss: 0.2023 - val_accuracy: 0.9142
Epoch 18/25
82/82 [=====] - 3s 40ms/step - loss: 0.1877 - accuracy: 0.9205 - val_loss: 0.1538 - val_accuracy: 0.9436
Epoch 19/25
82/82 [=====] - 3s 40ms/step - loss: 0.1814 - accuracy: 0.9301 - val_loss: 0.1633 - val_accuracy: 0.9375
Epoch 20/25
82/82 [=====] - 3s 42ms/step - loss: 0.1759 - accuracy: 0.9327 - val_loss: 0.1614 - val_accuracy: 0.9412
Epoch 21/25
82/82 [=====] - 3s 40ms/step - loss: 0.1748 - accuracy: 0.9350 - val_loss: 0.1688 - val_accuracy: 0.9424
Epoch 22/25
82/82 [=====] - 3s 41ms/step - loss: 0.1710 - accuracy: 0.9358 - val_loss: 0.1428 - val_accuracy: 0.9510
Epoch 23/25
82/82 [=====] - 3s 40ms/step - loss: 0.1735 - accuracy: 0.9339 - val_loss: 0.1430 - val_accuracy: 0.9534
Epoch 24/25
82/82 [=====] - 3s 40ms/step - loss: 0.1707 - accuracy: 0.9362 - val_loss: 0.1366 - val_accuracy: 0.9534
Epoch 25/25
82/82 [=====] - 3s 41ms/step - loss: 0.1641 - accuracy: 0.9362 - val_loss: 0.1576 - val_accuracy: 0.9461
21/21 [=====] - 0s 11ms/step - loss: 0.1589 - accuracy: 0.9434
Epoch 1/25
82/82 [=====] - 4s 48ms/step - loss: 0.5942 - accuracy: 0.7233 - val_loss: 0.5818 - val_accuracy: 0.7304
Epoch 2/25
82/82 [=====] - 3s 41ms/step - loss: 0.5741 - accuracy: 0.7318 - val_loss: 0.5673 - val_accuracy: 0.7304
Epoch 3/25
82/82 [=====] - 3s 41ms/step - loss: 0.5550 - accuracy: 0.7318 - val_loss: 0.5340 - val_accuracy: 0.7304
Epoch 4/25
82/82 [=====] - 3s 41ms/step - loss: 0.5204 - accuracy: 0.7432 - val_loss: 0.4870 - val_accuracy: 0.8260
Epoch 5/25
82/82 [=====] - 3s 41ms/step - loss: 0.4805 - accuracy: 0.7791 - val_loss: 0.3957 - val_accuracy: 0.8088
Epoch 6/25
82/82 [=====] - 3s 41ms/step - loss: 0.4400 - accuracy: 0.8105 - val_loss: 0.3809 - val_accuracy: 0.7880
Epoch 7/25
82/82 [=====] - 3s 43ms/step - loss: 0.3783 - accuracy: 0.8292 - val_loss: 0.3273 - val_accuracy: 0.8358
Epoch 8/25
82/82 [=====] - 3s 41ms/step - loss: 0.3157 - accuracy: 0.8686 - val_loss: 0.3696 - val_accuracy: 0.8235
Epoch 9/25
82/82 [=====] - 3s 41ms/step - loss: 0.2827 - accuracy: 0.8815 - val_loss: 0.3093 - val_accuracy: 0.8676
Epoch 10/25

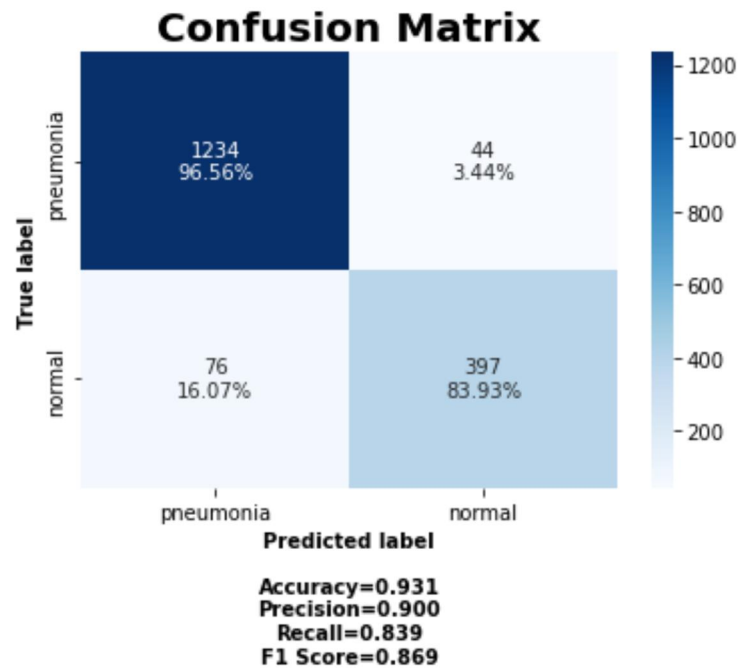
82/82 [=====] - 3s 41ms/step - loss: 0.2627 - accuracy: 0.8903 - val_loss: 0.2651 - val_accuracy: 0.8824
Epoch 11/25
82/82 [=====] - 3s 41ms/step - loss: 0.2245 - accuracy: 0.9106 - val_loss: 0.1885 - val_accuracy: 0.9216
Epoch 12/25
82/82 [=====] - 3s 41ms/step - loss: 0.2101 - accuracy: 0.9110 - val_loss: 0.1887 - val_accuracy: 0.9203
Epoch 13/25
82/82 [=====] - 3s 41ms/step - loss: 0.2036 - accuracy: 0.9148 - val_loss: 0.2155 - val_accuracy: 0.9044
Epoch 14/25
82/82 [=====] - 3s 40ms/step - loss: 0.1920 - accuracy: 0.9205 - val_loss: 0.1655 - val_accuracy: 0.9326
Epoch 15/25
82/82 [=====] - 3s 40ms/step - loss: 0.1902 - accuracy: 0.9224 - val_loss: 0.1665 - val_accuracy: 0.9338
Epoch 16/25
82/82 [=====] - 3s 41ms/step - loss: 0.1773 - accuracy: 0.9289 - val_loss: 0.1554 - val_accuracy: 0.9400
Epoch 17/25
82/82 [=====] - 3s 41ms/step - loss: 0.1719 - accuracy: 0.9347 - val_loss: 0.2370 - val_accuracy: 0.8873
Epoch 18/25
82/82 [=====] - 3s 41ms/step - loss: 0.1699 - accuracy: 0.9350 - val_loss: 0.1491 - val_accuracy: 0.9424
Epoch 19/25
82/82 [=====] - 3s 41ms/step - loss: 0.1708 - accuracy: 0.9335 - val_loss: 0.1476 - val_accuracy: 0.9412
Epoch 20/25
82/82 [=====] - 3s 41ms/step - loss: 0.1600 - accuracy: 0.9392 - val_loss: 0.1896 - val_accuracy: 0.9203
Epoch 21/25
82/82 [=====] - 3s 42ms/step - loss: 0.1602 - accuracy: 0.9373 - val_loss: 0.1747 - val_accuracy: 0.9277
Epoch 22/25
82/82 [=====] - 3s 41ms/step - loss: 0.1551 - accuracy: 0.9389 - val_loss: 0.1462 - val_accuracy: 0.9485
Epoch 23/25
82/82 [=====] - 3s 41ms/step - loss: 0.1516 - accuracy: 0.9419 - val_loss: 0.1541 - val_accuracy: 0.9412
Epoch 24/25
82/82 [=====] - 3s 41ms/step - loss: 0.1537 - accuracy: 0.9412 - val_loss: 0.1358 - val_accuracy: 0.9510
Epoch 25/25
82/82 [=====] - 3s 42ms/step - loss: 0.1551 - accuracy: 0.9392 - val_loss: 0.1699 - val_accuracy: 0.9412

```
In [44]: save_result('CNN #1', results_train[1], results_test[1], cvs[0], cvs[1], cvs[2], cvs[3], cvs[4])
```

Out[44]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	CNN #1	0.947	0.931	0.933	0.943	0.934	0.943	0.917	0.011	0.934
0	Initial Model	0.730	0.730	0.937	0.732	0.735	0.709	0.725	0.095	0.768

```
In [133]: cm_plot_labels = ['pneumonia', 'normal']  
make_confusion_matrix(cm, categories = cm_plot_labels, title='Confusion Matrix')
```



CNN Model 2

For this model, I added another round of Conv2D and MaxPooling layers, and changed the optimizer to "adam"

```
In [45]: model 2 = models.Sequential()

model 2.add(layers.Conv2D(32, (3, 3), activation='relu',
                           input_shape=(64, 64, 3)))
model 2.add(layers.MaxPooling2D((2, 2)))

model 2.add(layers.Conv2D(32, (4, 4), activation='relu'))
model 2.add(layers.MaxPooling2D((2, 2)))

model 2.add(layers.Conv2D(64, (3, 3), activation='relu'))
model 2.add(layers.MaxPooling2D((2, 2)))

model 2.add(layers.Conv2D(96, (3, 3), activation='relu'))
model 2.add(layers.MaxPooling2D((2, 2)))

model 2.add(layers.Flatten())
model 2.add(layers.Dense(64, activation='relu'))
model 2.add(layers.Dense(1, activation='sigmoid'))

model 2.compile(loss='binary_crossentropy',
                 optimizer="adam",
                 metrics=['accuracy'])
```

[illegible]

Epoch 1/50
103/103 [=====] - 5s 43ms/step - loss: 0.4916 - accuracy: 0.7710 - val_loss: 0.2972 - val_accuracy: 0.8738
Epoch 2/50
103/103 [=====] - 4s 41ms/step - loss: 0.2453 - accuracy: 0.8948 - val_loss: 0.1695 - val_accuracy: 0.9301
Epoch 3/50
103/103 [=====] - 4s 42ms/step - loss: 0.1757 - accuracy: 0.9309 - val_loss: 0.1237 - val_accuracy: 0.9559
Epoch 4/50
103/103 [=====] - 5s 48ms/step - loss: 0.1339 - accuracy: 0.9468 - val_loss: 0.1083 - val_accuracy: 0.9632
Epoch 5/50
103/103 [=====] - 4s 43ms/step - loss: 0.1274 - accuracy: 0.9514 - val_loss: 0.1092 - val_accuracy: 0.9510
Epoch 6/50
103/103 [=====] - 4s 42ms/step - loss: 0.1194 - accuracy: 0.9544 - val_loss: 0.1070 - val_accuracy: 0.9608
Epoch 7/50
103/103 [=====] - 4s 42ms/step - loss: 0.1136 - accuracy: 0.9590 - val_loss: 0.1119 - val_accuracy: 0.9522
Epoch 8/50
103/103 [=====] - 4s 42ms/step - loss: 0.0903 - accuracy: 0.9688 - val_loss: 0.0871 - val_accuracy: 0.9706
Epoch 9/50
103/103 [=====] - 4s 42ms/step - loss: 0.0780 - accuracy: 0.9722 - val_loss: 0.0970 - val_accuracy: 0.9694
Epoch 10/50
103/103 [=====] - 4s 44ms/step - loss: 0.0691 - accuracy: 0.9743 - val_loss: 0.1086 - val_accuracy: 0.9645
Epoch 11/50
103/103 [=====] - 4s 42ms/step - loss: 0.0675 - accuracy: 0.9771 - val_loss: 0.1007 - val_accuracy: 0.9694
Epoch 12/50
103/103 [=====] - 4s 42ms/step - loss: 0.0512 - accuracy: 0.9823 - val_loss: 0.0991 - val_accuracy: 0.9706
Epoch 13/50
103/103 [=====] - 4s 43ms/step - loss: 0.0510 - accuracy: 0.9841 - val_loss: 0.0933 - val_accuracy: 0.9706
Epoch 14/50
103/103 [=====] - 4s 42ms/step - loss: 0.0444 - accuracy: 0.9847 - val_loss: 0.0944 - val_accuracy: 0.9730
Epoch 15/50
103/103 [=====] - 4s 43ms/step - loss: 0.0416 - accuracy: 0.9853 - val_loss: 0.1057 - val_accuracy: 0.9694
Epoch 16/50
103/103 [=====] - 4s 42ms/step - loss: 0.0282 - accuracy: 0.9911 - val_loss: 0.1062 - val_accuracy: 0.9706
Epoch 17/50
103/103 [=====] - 4s 42ms/step - loss: 0.0198 - accuracy: 0.9945 - val_loss: 0.1398 - val_accuracy: 0.9669
Epoch 18/50
103/103 [=====] - 4s 42ms/step - loss: 0.0319 - accuracy: 0.9890 - val_loss: 0.1328 - val_accuracy: 0.9718
Epoch 19/50
103/103 [=====] - 4s 42ms/step - loss: 0.0218 - ac

curacy: 0.9917 - val_loss: 0.1347 - val_accuracy: 0.9681
Epoch 20/50
103/103 [=====] - 4s 42ms/step - loss: 0.0181 - accuracy: 0.9942 - val_loss: 0.1640 - val_accuracy: 0.9694
Epoch 21/50
103/103 [=====] - 4s 43ms/step - loss: 0.0143 - accuracy: 0.9942 - val_loss: 0.1298 - val_accuracy: 0.9669
Epoch 22/50
103/103 [=====] - 4s 42ms/step - loss: 0.0062 - accuracy: 0.9985 - val_loss: 0.1838 - val_accuracy: 0.9657
Epoch 23/50
103/103 [=====] - 4s 42ms/step - loss: 0.0213 - accuracy: 0.9908 - val_loss: 0.1629 - val_accuracy: 0.9706
Epoch 24/50
103/103 [=====] - 4s 42ms/step - loss: 0.0146 - accuracy: 0.9954 - val_loss: 0.1662 - val_accuracy: 0.9657
Epoch 25/50
103/103 [=====] - 4s 42ms/step - loss: 0.0099 - accuracy: 0.9969 - val_loss: 0.1664 - val_accuracy: 0.9694
Epoch 26/50
103/103 [=====] - 4s 43ms/step - loss: 0.0101 - accuracy: 0.9969 - val_loss: 0.1913 - val_accuracy: 0.9645
Epoch 27/50
103/103 [=====] - 4s 42ms/step - loss: 0.0108 - accuracy: 0.9954 - val_loss: 0.1408 - val_accuracy: 0.9620
Epoch 28/50
103/103 [=====] - 4s 42ms/step - loss: 0.0035 - accuracy: 0.9994 - val_loss: 0.1844 - val_accuracy: 0.9620
Epoch 29/50
103/103 [=====] - 4s 42ms/step - loss: 5.1243e-04 - accuracy: 1.0000 - val_loss: 0.1884 - val_accuracy: 0.9706
Epoch 30/50
103/103 [=====] - 4s 42ms/step - loss: 1.6940e-04 - accuracy: 1.0000 - val_loss: 0.1926 - val_accuracy: 0.9694
Epoch 31/50
103/103 [=====] - 4s 42ms/step - loss: 1.1738e-04 - accuracy: 1.0000 - val_loss: 0.1965 - val_accuracy: 0.9694
Epoch 32/50
103/103 [=====] - 4s 43ms/step - loss: 9.4963e-05 - accuracy: 1.0000 - val_loss: 0.1993 - val_accuracy: 0.9694
Epoch 33/50
103/103 [=====] - 4s 42ms/step - loss: 8.1633e-05 - accuracy: 1.0000 - val_loss: 0.2022 - val_accuracy: 0.9681
Epoch 34/50
103/103 [=====] - 4s 42ms/step - loss: 7.1860e-05 - accuracy: 1.0000 - val_loss: 0.2045 - val_accuracy: 0.9694
Epoch 35/50
103/103 [=====] - 4s 42ms/step - loss: 6.3143e-05 - accuracy: 1.0000 - val_loss: 0.2066 - val_accuracy: 0.9669
Epoch 36/50
103/103 [=====] - 4s 42ms/step - loss: 5.6993e-05 - accuracy: 1.0000 - val_loss: 0.2086 - val_accuracy: 0.9669
Epoch 37/50
103/103 [=====] - 5s 44ms/step - loss: 5.0583e-05 - accuracy: 1.0000 - val_loss: 0.2104 - val_accuracy: 0.9694
Epoch 38/50

```

103/103 [=====] - 4s 42ms/step - loss: 4.7869e-05
- accuracy: 1.0000 - val_loss: 0.2119 - val_accuracy: 0.9669
Epoch 39/50
103/103 [=====] - 4s 42ms/step - loss: 4.2553e-05
- accuracy: 1.0000 - val_loss: 0.2138 - val_accuracy: 0.9669
Epoch 40/50
103/103 [=====] - 4s 42ms/step - loss: 3.8941e-05
- accuracy: 1.0000 - val_loss: 0.2157 - val_accuracy: 0.9669
Epoch 41/50
103/103 [=====] - 4s 44ms/step - loss: 3.5869e-05
- accuracy: 1.0000 - val_loss: 0.2172 - val_accuracy: 0.9669
Epoch 42/50
103/103 [=====] - 4s 42ms/step - loss: 3.3604e-05
- accuracy: 1.0000 - val_loss: 0.2189 - val_accuracy: 0.9669
Epoch 43/50
103/103 [=====] - 4s 42ms/step - loss: 3.0837e-05
- accuracy: 1.0000 - val_loss: 0.2205 - val_accuracy: 0.9669
Epoch 44/50
103/103 [=====] - 4s 42ms/step - loss: 2.8667e-05
- accuracy: 1.0000 - val_loss: 0.2217 - val_accuracy: 0.9669
Epoch 45/50
103/103 [=====] - 4s 42ms/step - loss: 2.6556e-05
- accuracy: 1.0000 - val_loss: 0.2232 - val_accuracy: 0.9669
Epoch 46/50
103/103 [=====] - 4s 42ms/step - loss: 2.4866e-05
- accuracy: 1.0000 - val_loss: 0.2246 - val_accuracy: 0.9681
Epoch 47/50
103/103 [=====] - 4s 42ms/step - loss: 2.3281e-05
- accuracy: 1.0000 - val_loss: 0.2259 - val_accuracy: 0.9681
Epoch 48/50
103/103 [=====] - 4s 44ms/step - loss: 2.1520e-05
- accuracy: 1.0000 - val_loss: 0.2270 - val_accuracy: 0.9681
Epoch 49/50
103/103 [=====] - 4s 42ms/step - loss: 2.0070e-05
- accuracy: 1.0000 - val_loss: 0.2281 - val_accuracy: 0.9681
Epoch 50/50
103/103 [=====] - 4s 42ms/step - loss: 1.8915e-05
- accuracy: 1.0000 - val_loss: 0.2296 - val_accuracy: 0.9694

```

```
In [47]: results_train = model2.evaluate(train_images, train_y)
```

```

103/103 [=====] - 1s 11ms/step - loss: 1.7580e-05
- accuracy: 1.0000

```

```
In [48]: results_test = model2.evaluate(test_images, test_y)
```

```

55/55 [=====] - 1s 11ms/step - loss: 0.3307 - accuracy: 0.9486

```

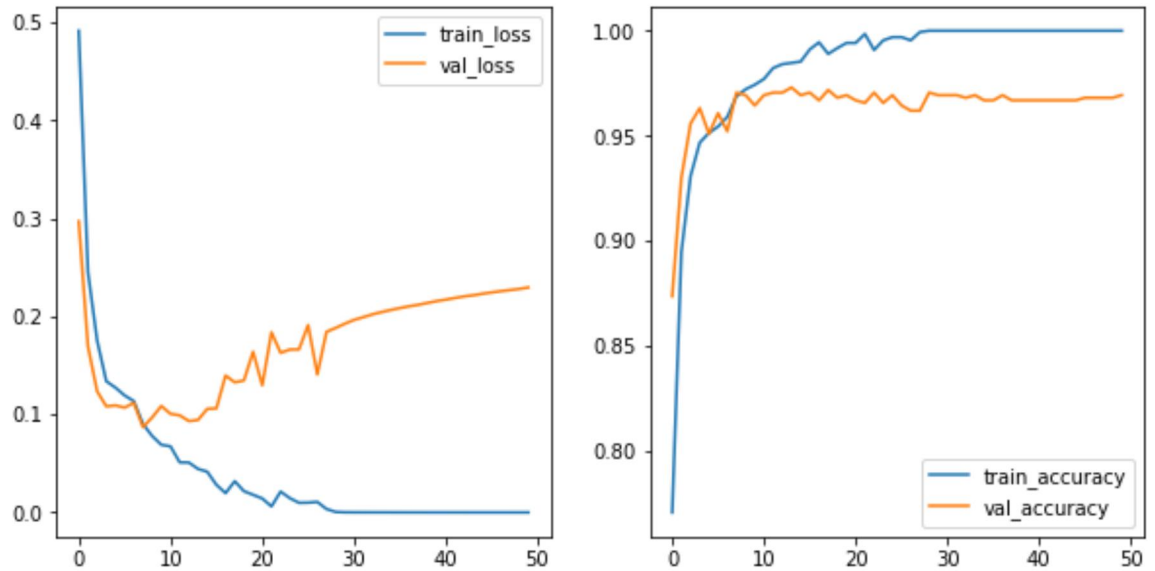
```

In [49]: train_loss = history.history['loss']
train_acc = history.history['accuracy']
val_loss = history.history['val_loss']
val_acc = history.history['val_accuracy']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=history.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=history.epoch, y=train_acc, ax=ax2, label='train_accuracy')
sns.lineplot(x=history.epoch, y=val_loss, ax=ax1, label='val_loss')
sns.lineplot(x=history.epoch, y=val_acc, ax=ax2, label='val_accuracy')

```

Out[49]: <AxesSubplot: >




```
In [52]: cvs = cross_val_score(keras_model3, train_images , train_y, cv=5)
```

Epoch 1/50
82/82 [=====] - 4s 45ms/step - loss: 0.5400 - accuracy: 0.7416 - val_loss: 0.2919 - val_accuracy: 0.8738
Epoch 2/50
82/82 [=====] - 4s 46ms/step - loss: 0.2888 - accuracy: 0.8758 - val_loss: 0.2811 - val_accuracy: 0.8664
Epoch 3/50
82/82 [=====] - 4s 44ms/step - loss: 0.2299 - accuracy: 0.9063 - val_loss: 0.2169 - val_accuracy: 0.9142
Epoch 4/50
82/82 [=====] - 3s 42ms/step - loss: 0.1956 - accuracy: 0.9228 - val_loss: 0.1761 - val_accuracy: 0.9326
Epoch 5/50
82/82 [=====] - 3s 42ms/step - loss: 0.1626 - accuracy: 0.9362 - val_loss: 0.1402 - val_accuracy: 0.9547
Epoch 6/50
82/82 [=====] - 3s 42ms/step - loss: 0.1563 - accuracy: 0.9400 - val_loss: 0.1242 - val_accuracy: 0.9657
Epoch 7/50
82/82 [=====] - 3s 42ms/step - loss: 0.1649 - accuracy: 0.9373 - val_loss: 0.1197 - val_accuracy: 0.9583
Epoch 8/50
82/82 [=====] - 3s 42ms/step - loss: 0.1277 - accuracy: 0.9507 - val_loss: 0.1178 - val_accuracy: 0.9632
Epoch 9/50
82/82 [=====] - 3s 42ms/step - loss: 0.1135 - accuracy: 0.9568 - val_loss: 0.1165 - val_accuracy: 0.9632
Epoch 10/50
82/82 [=====] - 3s 42ms/step - loss: 0.1051 - accuracy: 0.9618 - val_loss: 0.1098 - val_accuracy: 0.9608
Epoch 11/50
82/82 [=====] - 4s 43ms/step - loss: 0.0996 - accuracy: 0.9652 - val_loss: 0.1237 - val_accuracy: 0.9571
Epoch 12/50
82/82 [=====] - 3s 42ms/step - loss: 0.0948 - accuracy: 0.9625 - val_loss: 0.1120 - val_accuracy: 0.9608
Epoch 13/50
82/82 [=====] - 3s 42ms/step - loss: 0.0930 - accuracy: 0.9652 - val_loss: 0.1336 - val_accuracy: 0.9559
Epoch 14/50
82/82 [=====] - 3s 43ms/step - loss: 0.0808 - accuracy: 0.9694 - val_loss: 0.1069 - val_accuracy: 0.9620
Epoch 15/50
82/82 [=====] - 3s 42ms/step - loss: 0.0882 - accuracy: 0.9671 - val_loss: 0.1012 - val_accuracy: 0.9632
Epoch 16/50
82/82 [=====] - 3s 42ms/step - loss: 0.0725 - accuracy: 0.9713 - val_loss: 0.1094 - val_accuracy: 0.9596
Epoch 17/50
82/82 [=====] - 3s 42ms/step - loss: 0.0726 - accuracy: 0.9744 - val_loss: 0.1182 - val_accuracy: 0.9583
Epoch 18/50
82/82 [=====] - 4s 44ms/step - loss: 0.0640 - accuracy: 0.9794 - val_loss: 0.1218 - val_accuracy: 0.9596
Epoch 19/50
82/82 [=====] - 3s 42ms/step - loss: 0.0573 - accuracy:

racy: 0.9786 - val_loss: 0.1126 - val_accuracy: 0.9571
Epoch 20/50
82/82 [=====] - 3s 42ms/step - loss: 0.0561 - accu
racy: 0.9786 - val_loss: 0.1195 - val_accuracy: 0.9583
Epoch 21/50
82/82 [=====] - 3s 42ms/step - loss: 0.0447 - accu
racy: 0.9836 - val_loss: 0.1117 - val_accuracy: 0.9596
Epoch 22/50
82/82 [=====] - 3s 43ms/step - loss: 0.0320 - accu
racy: 0.9904 - val_loss: 0.1097 - val_accuracy: 0.9645
Epoch 23/50
82/82 [=====] - 3s 42ms/step - loss: 0.0323 - accu
racy: 0.9893 - val_loss: 0.1240 - val_accuracy: 0.9620
Epoch 24/50
82/82 [=====] - 4s 44ms/step - loss: 0.0261 - accu
racy: 0.9889 - val_loss: 0.1391 - val_accuracy: 0.9559
Epoch 25/50
82/82 [=====] - 3s 42ms/step - loss: 0.0244 - accu
racy: 0.9920 - val_loss: 0.1467 - val_accuracy: 0.9571
Epoch 26/50
82/82 [=====] - 3s 42ms/step - loss: 0.0354 - accu
racy: 0.9893 - val_loss: 0.1641 - val_accuracy: 0.9498
Epoch 27/50
82/82 [=====] - 3s 43ms/step - loss: 0.0646 - accu
racy: 0.9759 - val_loss: 0.1237 - val_accuracy: 0.9608
Epoch 28/50
82/82 [=====] - 3s 42ms/step - loss: 0.0219 - accu
racy: 0.9924 - val_loss: 0.1607 - val_accuracy: 0.9620
Epoch 29/50
82/82 [=====] - 3s 42ms/step - loss: 0.0170 - accu
racy: 0.9927 - val_loss: 0.2135 - val_accuracy: 0.9449
Epoch 30/50
82/82 [=====] - 4s 43ms/step - loss: 0.0122 - accu
racy: 0.9969 - val_loss: 0.1521 - val_accuracy: 0.9596
Epoch 31/50
82/82 [=====] - 4s 44ms/step - loss: 0.0042 - accu
racy: 1.0000 - val_loss: 0.1761 - val_accuracy: 0.9620
Epoch 32/50
82/82 [=====] - 3s 42ms/step - loss: 0.0028 - accu
racy: 1.0000 - val_loss: 0.1802 - val_accuracy: 0.9583
Epoch 33/50
82/82 [=====] - 3s 42ms/step - loss: 0.0041 - accu
racy: 0.9996 - val_loss: 0.1766 - val_accuracy: 0.9596
Epoch 34/50
82/82 [=====] - 3s 42ms/step - loss: 0.0015 - accu
racy: 1.0000 - val_loss: 0.1860 - val_accuracy: 0.9608
Epoch 35/50
82/82 [=====] - 3s 42ms/step - loss: 9.5021e-04 -
accuracy: 1.0000 - val_loss: 0.1925 - val_accuracy: 0.9596
Epoch 36/50
82/82 [=====] - 3s 42ms/step - loss: 8.2422e-04 -
accuracy: 1.0000 - val_loss: 0.1924 - val_accuracy: 0.9608
Epoch 37/50
82/82 [=====] - 4s 44ms/step - loss: 6.6059e-04 -
accuracy: 1.0000 - val_loss: 0.1999 - val_accuracy: 0.9608
Epoch 38/50

82/82 [=====] - 3s 42ms/step - loss: 6.0012e-04 - accuracy: 1.0000 - val_loss: 0.2036 - val_accuracy: 0.9608
Epoch 39/50
82/82 [=====] - 3s 43ms/step - loss: 4.7597e-04 - accuracy: 1.0000 - val_loss: 0.2078 - val_accuracy: 0.9608
Epoch 40/50
82/82 [=====] - 3s 42ms/step - loss: 3.9908e-04 - accuracy: 1.0000 - val_loss: 0.2113 - val_accuracy: 0.9596
Epoch 41/50
82/82 [=====] - 3s 42ms/step - loss: 3.8389e-04 - accuracy: 1.0000 - val_loss: 0.2150 - val_accuracy: 0.9620
Epoch 42/50
82/82 [=====] - 3s 43ms/step - loss: 3.3017e-04 - accuracy: 1.0000 - val_loss: 0.2169 - val_accuracy: 0.9596
Epoch 43/50
82/82 [=====] - 4s 44ms/step - loss: 3.1774e-04 - accuracy: 1.0000 - val_loss: 0.2209 - val_accuracy: 0.9608
Epoch 44/50
82/82 [=====] - 3s 42ms/step - loss: 2.5272e-04 - accuracy: 1.0000 - val_loss: 0.2231 - val_accuracy: 0.9608
Epoch 45/50
82/82 [=====] - 3s 43ms/step - loss: 2.4373e-04 - accuracy: 1.0000 - val_loss: 0.2280 - val_accuracy: 0.9608
Epoch 46/50
82/82 [=====] - 3s 42ms/step - loss: 2.2642e-04 - accuracy: 1.0000 - val_loss: 0.2272 - val_accuracy: 0.9583
Epoch 47/50
82/82 [=====] - 3s 42ms/step - loss: 2.1068e-04 - accuracy: 1.0000 - val_loss: 0.2308 - val_accuracy: 0.9596
Epoch 48/50
82/82 [=====] - 3s 43ms/step - loss: 1.8560e-04 - accuracy: 1.0000 - val_loss: 0.2338 - val_accuracy: 0.9596
Epoch 49/50
82/82 [=====] - 4s 44ms/step - loss: 1.6445e-04 - accuracy: 1.0000 - val_loss: 0.2356 - val_accuracy: 0.9583
Epoch 50/50
82/82 [=====] - 3s 42ms/step - loss: 1.4738e-04 - accuracy: 1.0000 - val_loss: 0.2380 - val_accuracy: 0.9583
21/21 [=====] - 0s 10ms/step - loss: 0.2972 - accuracy: 0.9573
Epoch 1/50
82/82 [=====] - 4s 44ms/step - loss: 0.5716 - accuracy: 0.7298 - val_loss: 0.4294 - val_accuracy: 0.9032
Epoch 2/50
82/82 [=====] - 3s 42ms/step - loss: 0.2816 - accuracy: 0.8873 - val_loss: 0.2092 - val_accuracy: 0.9240
Epoch 3/50
82/82 [=====] - 3s 42ms/step - loss: 0.1988 - accuracy: 0.9205 - val_loss: 0.1910 - val_accuracy: 0.9228
Epoch 4/50
82/82 [=====] - 3s 42ms/step - loss: 0.1610 - accuracy: 0.9412 - val_loss: 0.1281 - val_accuracy: 0.9608
Epoch 5/50
82/82 [=====] - 4s 44ms/step - loss: 0.1497 - accuracy: 0.9461 - val_loss: 0.1183 - val_accuracy: 0.9571
Epoch 6/50

82/82 [=====] - 3s 42ms/step - loss: 0.1271 - accuracy: 0.9553 - val_loss: 0.1131 - val_accuracy: 0.9522
Epoch 7/50
82/82 [=====] - 3s 42ms/step - loss: 0.1277 - accuracy: 0.9522 - val_loss: 0.1097 - val_accuracy: 0.9596
Epoch 8/50
82/82 [=====] - 3s 42ms/step - loss: 0.1151 - accuracy: 0.9541 - val_loss: 0.0989 - val_accuracy: 0.9669
Epoch 9/50
82/82 [=====] - 3s 42ms/step - loss: 0.1058 - accuracy: 0.9656 - val_loss: 0.1065 - val_accuracy: 0.9596
Epoch 10/50
82/82 [=====] - 3s 42ms/step - loss: 0.0934 - accuracy: 0.9668 - val_loss: 0.1153 - val_accuracy: 0.9632
Epoch 11/50
82/82 [=====] - 4s 44ms/step - loss: 0.0857 - accuracy: 0.9664 - val_loss: 0.1145 - val_accuracy: 0.9669
Epoch 12/50
82/82 [=====] - 3s 42ms/step - loss: 0.0809 - accuracy: 0.9702 - val_loss: 0.1171 - val_accuracy: 0.9620
Epoch 13/50
82/82 [=====] - 3s 42ms/step - loss: 0.0761 - accuracy: 0.9721 - val_loss: 0.0956 - val_accuracy: 0.9718
Epoch 14/50
82/82 [=====] - 3s 42ms/step - loss: 0.0617 - accuracy: 0.9782 - val_loss: 0.1050 - val_accuracy: 0.9730
Epoch 15/50
82/82 [=====] - 3s 42ms/step - loss: 0.0545 - accuracy: 0.9805 - val_loss: 0.1835 - val_accuracy: 0.9363
Epoch 16/50
82/82 [=====] - 3s 42ms/step - loss: 0.0611 - accuracy: 0.9778 - val_loss: 0.1091 - val_accuracy: 0.9718
Epoch 17/50
82/82 [=====] - 4s 44ms/step - loss: 0.0461 - accuracy: 0.9836 - val_loss: 0.1089 - val_accuracy: 0.9681
Epoch 18/50
82/82 [=====] - 3s 42ms/step - loss: 0.0718 - accuracy: 0.9740 - val_loss: 0.1045 - val_accuracy: 0.9743
Epoch 19/50
82/82 [=====] - 3s 42ms/step - loss: 0.0461 - accuracy: 0.9820 - val_loss: 0.1225 - val_accuracy: 0.9706
Epoch 20/50
82/82 [=====] - 4s 44ms/step - loss: 0.0522 - accuracy: 0.9801 - val_loss: 0.1158 - val_accuracy: 0.9669
Epoch 21/50
82/82 [=====] - 3s 42ms/step - loss: 0.0276 - accuracy: 0.9908 - val_loss: 0.1462 - val_accuracy: 0.9681
Epoch 22/50
82/82 [=====] - 3s 42ms/step - loss: 0.0188 - accuracy: 0.9954 - val_loss: 0.1468 - val_accuracy: 0.9657
Epoch 23/50
82/82 [=====] - 4s 44ms/step - loss: 0.0401 - accuracy: 0.9847 - val_loss: 0.1190 - val_accuracy: 0.9669
Epoch 24/50
82/82 [=====] - 4s 43ms/step - loss: 0.0203 - accuracy: 0.9927 - val_loss: 0.1450 - val_accuracy: 0.9730

Epoch 25/50
82/82 [=====] - 3s 43ms/step - loss: 0.0159 - accuracy: 0.9958 - val_loss: 0.1296 - val_accuracy: 0.9706
Epoch 26/50
82/82 [=====] - 3s 43ms/step - loss: 0.0109 - accuracy: 0.9977 - val_loss: 0.1677 - val_accuracy: 0.9620
Epoch 27/50
82/82 [=====] - 3s 42ms/step - loss: 0.0059 - accuracy: 0.9996 - val_loss: 0.1772 - val_accuracy: 0.9645
Epoch 28/50
82/82 [=====] - 4s 44ms/step - loss: 0.0035 - accuracy: 0.9996 - val_loss: 0.1911 - val_accuracy: 0.9657
Epoch 29/50
82/82 [=====] - 3s 42ms/step - loss: 0.0017 - accuracy: 1.0000 - val_loss: 0.2008 - val_accuracy: 0.9632
Epoch 30/50
82/82 [=====] - 4s 44ms/step - loss: 9.4031e-04 - accuracy: 1.0000 - val_loss: 0.2080 - val_accuracy: 0.9657
Epoch 31/50
82/82 [=====] - 3s 42ms/step - loss: 7.4947e-04 - accuracy: 1.0000 - val_loss: 0.2109 - val_accuracy: 0.9645
Epoch 32/50
82/82 [=====] - 3s 42ms/step - loss: 5.7984e-04 - accuracy: 1.0000 - val_loss: 0.2189 - val_accuracy: 0.9657
Epoch 33/50
82/82 [=====] - 3s 42ms/step - loss: 5.4017e-04 - accuracy: 1.0000 - val_loss: 0.2246 - val_accuracy: 0.9645
Epoch 34/50
82/82 [=====] - 3s 42ms/step - loss: 4.2592e-04 - accuracy: 1.0000 - val_loss: 0.2285 - val_accuracy: 0.9632
Epoch 35/50
82/82 [=====] - 3s 43ms/step - loss: 3.4844e-04 - accuracy: 1.0000 - val_loss: 0.2332 - val_accuracy: 0.9620
Epoch 36/50
82/82 [=====] - 4s 44ms/step - loss: 3.0983e-04 - accuracy: 1.0000 - val_loss: 0.2357 - val_accuracy: 0.9632
Epoch 37/50
82/82 [=====] - 3s 42ms/step - loss: 2.6340e-04 - accuracy: 1.0000 - val_loss: 0.2401 - val_accuracy: 0.9620
Epoch 38/50
82/82 [=====] - 3s 42ms/step - loss: 2.3312e-04 - accuracy: 1.0000 - val_loss: 0.2431 - val_accuracy: 0.9620
Epoch 39/50
82/82 [=====] - 3s 42ms/step - loss: 2.2427e-04 - accuracy: 1.0000 - val_loss: 0.2461 - val_accuracy: 0.9620
Epoch 40/50
82/82 [=====] - 3s 42ms/step - loss: 1.9053e-04 - accuracy: 1.0000 - val_loss: 0.2503 - val_accuracy: 0.9632
Epoch 41/50
82/82 [=====] - 3s 43ms/step - loss: 1.7934e-04 - accuracy: 1.0000 - val_loss: 0.2516 - val_accuracy: 0.9645
Epoch 42/50
82/82 [=====] - 4s 44ms/step - loss: 1.5544e-04 - accuracy: 1.0000 - val_loss: 0.2559 - val_accuracy: 0.9632
Epoch 43/50
82/82 [=====] - 3s 43ms/step - loss: 1.4493e-04 -

accuracy: 1.0000 - val_loss: 0.2591 - val_accuracy: 0.9632
Epoch 44/50
82/82 [=====] - 3s 42ms/step - loss: 1.3533e-04 -
accuracy: 1.0000 - val_loss: 0.2620 - val_accuracy: 0.9645
Epoch 45/50
82/82 [=====] - 3s 42ms/step - loss: 1.2275e-04 -
accuracy: 1.0000 - val_loss: 0.2626 - val_accuracy: 0.9632
Epoch 46/50
82/82 [=====] - 3s 42ms/step - loss: 1.0982e-04 -
accuracy: 1.0000 - val_loss: 0.2672 - val_accuracy: 0.9632
Epoch 47/50
82/82 [=====] - 3s 42ms/step - loss: 1.0919e-04 -
accuracy: 1.0000 - val_loss: 0.2694 - val_accuracy: 0.9632
Epoch 48/50
82/82 [=====] - 4s 44ms/step - loss: 9.6391e-05 -
accuracy: 1.0000 - val_loss: 0.2725 - val_accuracy: 0.9645
Epoch 49/50
82/82 [=====] - 3s 43ms/step - loss: 9.1741e-05 -
accuracy: 1.0000 - val_loss: 0.2743 - val_accuracy: 0.9632
Epoch 50/50
82/82 [=====] - 3s 42ms/step - loss: 8.0975e-05 -
accuracy: 1.0000 - val_loss: 0.2741 - val_accuracy: 0.9632
21/21 [=====] - 0s 11ms/step - loss: 0.1569 - accu
racy: 0.9694
Epoch 1/50
82/82 [=====] - 4s 45ms/step - loss: 0.5519 - accu
racy: 0.7516 - val_loss: 0.2965 - val_accuracy: 0.8824
Epoch 2/50
82/82 [=====] - 4s 44ms/step - loss: 0.2903 - accu
racy: 0.8785 - val_loss: 0.2407 - val_accuracy: 0.8946
Epoch 3/50
82/82 [=====] - 3s 42ms/step - loss: 0.2139 - accu
racy: 0.9079 - val_loss: 0.1594 - val_accuracy: 0.9424
Epoch 4/50
82/82 [=====] - 4s 44ms/step - loss: 0.1788 - accu
racy: 0.9305 - val_loss: 0.1423 - val_accuracy: 0.9522
Epoch 5/50
82/82 [=====] - 3s 42ms/step - loss: 0.1516 - accu
racy: 0.9454 - val_loss: 0.1710 - val_accuracy: 0.9338
Epoch 6/50
82/82 [=====] - 3s 42ms/step - loss: 0.1387 - accu
racy: 0.9446 - val_loss: 0.2065 - val_accuracy: 0.9130
Epoch 7/50
82/82 [=====] - 4s 43ms/step - loss: 0.1491 - accu
racy: 0.9434 - val_loss: 0.1696 - val_accuracy: 0.9314
Epoch 8/50
82/82 [=====] - 3s 42ms/step - loss: 0.1402 - accu
racy: 0.9446 - val_loss: 0.1193 - val_accuracy: 0.9718
Epoch 9/50
82/82 [=====] - 3s 42ms/step - loss: 0.1258 - accu
racy: 0.9492 - val_loss: 0.1076 - val_accuracy: 0.9730
Epoch 10/50
82/82 [=====] - 3s 42ms/step - loss: 0.1166 - accu
racy: 0.9557 - val_loss: 0.1126 - val_accuracy: 0.9706
Epoch 11/50
82/82 [=====] - 4s 44ms/step - loss: 0.1034 - accu

racy: 0.9614 - val_loss: 0.1050 - val_accuracy: 0.9681
Epoch 12/50
82/82 [=====] - 3s 42ms/step - loss: 0.0952 - accu
racy: 0.9656 - val_loss: 0.0951 - val_accuracy: 0.9694
Epoch 13/50
82/82 [=====] - 3s 43ms/step - loss: 0.0839 - accu
racy: 0.9687 - val_loss: 0.0938 - val_accuracy: 0.9681
Epoch 14/50
82/82 [=====] - 3s 42ms/step - loss: 0.0799 - accu
racy: 0.9687 - val_loss: 0.1144 - val_accuracy: 0.9706
Epoch 15/50
82/82 [=====] - 3s 42ms/step - loss: 0.0783 - accu
racy: 0.9710 - val_loss: 0.1110 - val_accuracy: 0.9547
Epoch 16/50
82/82 [=====] - 3s 42ms/step - loss: 0.0656 - accu
racy: 0.9740 - val_loss: 0.0981 - val_accuracy: 0.9657
Epoch 17/50
82/82 [=====] - 4s 44ms/step - loss: 0.0691 - accu
racy: 0.9755 - val_loss: 0.0969 - val_accuracy: 0.9718
Epoch 18/50
82/82 [=====] - 3s 42ms/step - loss: 0.0488 - accu
racy: 0.9828 - val_loss: 0.0905 - val_accuracy: 0.9706
Epoch 19/50
82/82 [=====] - 3s 43ms/step - loss: 0.0515 - accu
racy: 0.9805 - val_loss: 0.0967 - val_accuracy: 0.9669
Epoch 20/50
82/82 [=====] - 3s 42ms/step - loss: 0.0434 - accu
racy: 0.9847 - val_loss: 0.1135 - val_accuracy: 0.9632
Epoch 21/50
82/82 [=====] - 3s 43ms/step - loss: 0.0448 - accu
racy: 0.9828 - val_loss: 0.1342 - val_accuracy: 0.9632
Epoch 22/50
82/82 [=====] - 3s 42ms/step - loss: 0.0467 - accu
racy: 0.9847 - val_loss: 0.1106 - val_accuracy: 0.9706
Epoch 23/50
82/82 [=====] - 4s 44ms/step - loss: 0.0388 - accu
racy: 0.9851 - val_loss: 0.1099 - val_accuracy: 0.9645
Epoch 24/50
82/82 [=====] - 4s 43ms/step - loss: 0.0361 - accu
racy: 0.9859 - val_loss: 0.1137 - val_accuracy: 0.9657
Epoch 25/50
82/82 [=====] - 3s 43ms/step - loss: 0.0172 - accu
racy: 0.9947 - val_loss: 0.1340 - val_accuracy: 0.9694
Epoch 26/50
82/82 [=====] - 3s 42ms/step - loss: 0.0116 - accu
racy: 0.9958 - val_loss: 0.1672 - val_accuracy: 0.9571
Epoch 27/50
82/82 [=====] - 3s 43ms/step - loss: 0.0465 - accu
racy: 0.9794 - val_loss: 0.1064 - val_accuracy: 0.9730
Epoch 28/50
82/82 [=====] - 3s 42ms/step - loss: 0.0162 - accu
racy: 0.9947 - val_loss: 0.1508 - val_accuracy: 0.9669
Epoch 29/50
82/82 [=====] - 4s 44ms/step - loss: 0.0353 - accu
racy: 0.9878 - val_loss: 0.1391 - val_accuracy: 0.9681
Epoch 30/50

82/82 [=====] - 3s 42ms/step - loss: 0.0403 - accuracy: 0.9820 - val_loss: 0.1470 - val_accuracy: 0.9694
Epoch 31/50
82/82 [=====] - 3s 43ms/step - loss: 0.0188 - accuracy: 0.9924 - val_loss: 0.1335 - val_accuracy: 0.9706
Epoch 32/50
82/82 [=====] - 3s 42ms/step - loss: 0.0064 - accuracy: 0.9985 - val_loss: 0.1523 - val_accuracy: 0.9718
Epoch 33/50
82/82 [=====] - 3s 43ms/step - loss: 0.0032 - accuracy: 0.9996 - val_loss: 0.1677 - val_accuracy: 0.9718
Epoch 34/50
82/82 [=====] - 3s 42ms/step - loss: 0.0015 - accuracy: 1.0000 - val_loss: 0.1737 - val_accuracy: 0.9743
Epoch 35/50
82/82 [=====] - 3s 43ms/step - loss: 8.6725e-04 - accuracy: 1.0000 - val_loss: 0.1817 - val_accuracy: 0.9706
Epoch 36/50
82/82 [=====] - 4s 44ms/step - loss: 6.4678e-04 - accuracy: 1.0000 - val_loss: 0.1882 - val_accuracy: 0.9706
Epoch 37/50
82/82 [=====] - 3s 43ms/step - loss: 5.5152e-04 - accuracy: 1.0000 - val_loss: 0.1941 - val_accuracy: 0.9706
Epoch 38/50
82/82 [=====] - 3s 42ms/step - loss: 4.6476e-04 - accuracy: 1.0000 - val_loss: 0.1986 - val_accuracy: 0.9706
Epoch 39/50
82/82 [=====] - 4s 43ms/step - loss: 4.1378e-04 - accuracy: 1.0000 - val_loss: 0.2038 - val_accuracy: 0.9706
Epoch 40/50
82/82 [=====] - 3s 42ms/step - loss: 3.4679e-04 - accuracy: 1.0000 - val_loss: 0.2086 - val_accuracy: 0.9706
Epoch 41/50
82/82 [=====] - 3s 42ms/step - loss: 3.2036e-04 - accuracy: 1.0000 - val_loss: 0.2112 - val_accuracy: 0.9706
Epoch 42/50
82/82 [=====] - 3s 43ms/step - loss: 2.8017e-04 - accuracy: 1.0000 - val_loss: 0.2145 - val_accuracy: 0.9706
Epoch 43/50
82/82 [=====] - 4s 44ms/step - loss: 2.4849e-04 - accuracy: 1.0000 - val_loss: 0.2177 - val_accuracy: 0.9706
Epoch 44/50
82/82 [=====] - 3s 42ms/step - loss: 2.1901e-04 - accuracy: 1.0000 - val_loss: 0.2210 - val_accuracy: 0.9706
Epoch 45/50
82/82 [=====] - 3s 43ms/step - loss: 2.1609e-04 - accuracy: 1.0000 - val_loss: 0.2242 - val_accuracy: 0.9706
Epoch 46/50
82/82 [=====] - 3s 43ms/step - loss: 1.8470e-04 - accuracy: 1.0000 - val_loss: 0.2269 - val_accuracy: 0.9706
Epoch 47/50
82/82 [=====] - 4s 44ms/step - loss: 1.5904e-04 - accuracy: 1.0000 - val_loss: 0.2294 - val_accuracy: 0.9706
Epoch 48/50
82/82 [=====] - 3s 43ms/step - loss: 1.5150e-04 - accuracy: 1.0000 - val_loss: 0.2311 - val_accuracy: 0.9706

Epoch 49/50
82/82 [=====] - 3s 43ms/step - loss: 1.3177e-04 - accuracy: 1.0000 - val_loss: 0.2342 - val_accuracy: 0.9706
Epoch 50/50
82/82 [=====] - 4s 44ms/step - loss: 1.2321e-04 - accuracy: 1.0000 - val_loss: 0.2371 - val_accuracy: 0.9706
21/21 [=====] - 0s 10ms/step - loss: 0.3247 - accuracy: 0.9557
Epoch 1/50
82/82 [=====] - 4s 45ms/step - loss: 0.5754 - accuracy: 0.7283 - val_loss: 0.4660 - val_accuracy: 0.7316
Epoch 2/50
82/82 [=====] - 4s 44ms/step - loss: 0.3150 - accuracy: 0.8575 - val_loss: 0.2091 - val_accuracy: 0.9105
Epoch 3/50
82/82 [=====] - 3s 43ms/step - loss: 0.2160 - accuracy: 0.9110 - val_loss: 0.1728 - val_accuracy: 0.9387
Epoch 4/50
82/82 [=====] - 4s 45ms/step - loss: 0.1911 - accuracy: 0.9232 - val_loss: 0.1389 - val_accuracy: 0.9571
Epoch 5/50
82/82 [=====] - 3s 43ms/step - loss: 0.1345 - accuracy: 0.9519 - val_loss: 0.1551 - val_accuracy: 0.9436
Epoch 6/50
82/82 [=====] - 4s 48ms/step - loss: 0.1245 - accuracy: 0.9549 - val_loss: 0.1225 - val_accuracy: 0.9559
Epoch 7/50
82/82 [=====] - 4s 47ms/step - loss: 0.1196 - accuracy: 0.9530 - val_loss: 0.1149 - val_accuracy: 0.9669
Epoch 8/50
82/82 [=====] - 4s 48ms/step - loss: 0.1022 - accuracy: 0.9622 - val_loss: 0.1119 - val_accuracy: 0.9608
Epoch 9/50
82/82 [=====] - 4s 48ms/step - loss: 0.0910 - accuracy: 0.9698 - val_loss: 0.1198 - val_accuracy: 0.9620
Epoch 10/50
82/82 [=====] - 4s 45ms/step - loss: 0.0857 - accuracy: 0.9690 - val_loss: 0.1091 - val_accuracy: 0.9669
Epoch 11/50
82/82 [=====] - 4s 45ms/step - loss: 0.0818 - accuracy: 0.9675 - val_loss: 0.1129 - val_accuracy: 0.9620
Epoch 12/50
82/82 [=====] - 4s 45ms/step - loss: 0.0941 - accuracy: 0.9660 - val_loss: 0.1900 - val_accuracy: 0.9338
Epoch 13/50
82/82 [=====] - 4s 45ms/step - loss: 0.0707 - accuracy: 0.9755 - val_loss: 0.1120 - val_accuracy: 0.9657
Epoch 14/50
82/82 [=====] - 4s 45ms/step - loss: 0.0657 - accuracy: 0.9763 - val_loss: 0.1130 - val_accuracy: 0.9657
Epoch 15/50
82/82 [=====] - 4s 45ms/step - loss: 0.0568 - accuracy: 0.9824 - val_loss: 0.1246 - val_accuracy: 0.9620
Epoch 16/50
82/82 [=====] - 4s 48ms/step - loss: 0.0389 - accuracy: 0.9885 - val_loss: 0.1195 - val_accuracy: 0.9718

Epoch 17/50
82/82 [=====] - 4s 46ms/step - loss: 0.0475 - accuracy: 0.9817 - val_loss: 0.1052 - val_accuracy: 0.9743
Epoch 18/50
82/82 [=====] - 4s 46ms/step - loss: 0.0376 - accuracy: 0.9882 - val_loss: 0.1176 - val_accuracy: 0.9669
Epoch 19/50
82/82 [=====] - 4s 45ms/step - loss: 0.0235 - accuracy: 0.9943 - val_loss: 0.1300 - val_accuracy: 0.9694
Epoch 20/50
82/82 [=====] - 4s 46ms/step - loss: 0.0206 - accuracy: 0.9943 - val_loss: 0.1246 - val_accuracy: 0.9706
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0173 - accuracy: 0.9977 - val_loss: 0.1238 - val_accuracy: 0.9743
Epoch 22/50
82/82 [=====] - 4s 46ms/step - loss: 0.0126 - accuracy: 0.9981 - val_loss: 0.1235 - val_accuracy: 0.9743
Epoch 23/50
82/82 [=====] - 4s 46ms/step - loss: 0.0329 - accuracy: 0.9882 - val_loss: 0.1447 - val_accuracy: 0.9645
Epoch 24/50
82/82 [=====] - 4s 46ms/step - loss: 0.0156 - accuracy: 0.9947 - val_loss: 0.1688 - val_accuracy: 0.9596
Epoch 25/50
82/82 [=====] - 4s 46ms/step - loss: 0.0095 - accuracy: 0.9981 - val_loss: 0.1479 - val_accuracy: 0.9681
Epoch 26/50
82/82 [=====] - 4s 45ms/step - loss: 0.0047 - accuracy: 1.0000 - val_loss: 0.1622 - val_accuracy: 0.9694
Epoch 27/50
82/82 [=====] - 4s 47ms/step - loss: 0.0034 - accuracy: 0.9996 - val_loss: 0.1858 - val_accuracy: 0.9694
Epoch 28/50
82/82 [=====] - 4s 45ms/step - loss: 0.0020 - accuracy: 1.0000 - val_loss: 0.1750 - val_accuracy: 0.9694
Epoch 29/50
82/82 [=====] - 4s 45ms/step - loss: 8.7292e-04 - accuracy: 1.0000 - val_loss: 0.1829 - val_accuracy: 0.9706
Epoch 30/50
82/82 [=====] - 4s 46ms/step - loss: 5.6547e-04 - accuracy: 1.0000 - val_loss: 0.1824 - val_accuracy: 0.9706
Epoch 31/50
82/82 [=====] - 4s 46ms/step - loss: 5.0831e-04 - accuracy: 1.0000 - val_loss: 0.1872 - val_accuracy: 0.9730
Epoch 32/50
82/82 [=====] - 4s 47ms/step - loss: 3.7873e-04 - accuracy: 1.0000 - val_loss: 0.1896 - val_accuracy: 0.9730
Epoch 33/50
82/82 [=====] - 4s 45ms/step - loss: 3.1857e-04 - accuracy: 1.0000 - val_loss: 0.1923 - val_accuracy: 0.9706
Epoch 34/50
82/82 [=====] - 4s 45ms/step - loss: 2.7865e-04 - accuracy: 1.0000 - val_loss: 0.1958 - val_accuracy: 0.9706
Epoch 35/50
82/82 [=====] - 4s 46ms/step - loss: 2.3756e-04 -

accuracy: 1.0000 - val_loss: 0.1976 - val_accuracy: 0.9730
Epoch 36/50
82/82 [=====] - 4s 46ms/step - loss: 2.2161e-04 -
accuracy: 1.0000 - val_loss: 0.2010 - val_accuracy: 0.9706
Epoch 37/50
82/82 [=====] - 4s 47ms/step - loss: 1.9559e-04 -
accuracy: 1.0000 - val_loss: 0.2057 - val_accuracy: 0.9706
Epoch 38/50
82/82 [=====] - 4s 46ms/step - loss: 1.8379e-04 -
accuracy: 1.0000 - val_loss: 0.2068 - val_accuracy: 0.9706
Epoch 39/50
82/82 [=====] - 4s 46ms/step - loss: 1.6044e-04 -
accuracy: 1.0000 - val_loss: 0.2084 - val_accuracy: 0.9706
Epoch 40/50
82/82 [=====] - 4s 46ms/step - loss: 1.4755e-04 -
accuracy: 1.0000 - val_loss: 0.2093 - val_accuracy: 0.9718
Epoch 41/50
82/82 [=====] - 4s 45ms/step - loss: 1.3602e-04 -
accuracy: 1.0000 - val_loss: 0.2128 - val_accuracy: 0.9706
Epoch 42/50
82/82 [=====] - 4s 47ms/step - loss: 1.3048e-04 -
accuracy: 1.0000 - val_loss: 0.2160 - val_accuracy: 0.9706
Epoch 43/50
82/82 [=====] - 4s 46ms/step - loss: 1.1365e-04 -
accuracy: 1.0000 - val_loss: 0.2160 - val_accuracy: 0.9718
Epoch 44/50
82/82 [=====] - 4s 45ms/step - loss: 1.0510e-04 -
accuracy: 1.0000 - val_loss: 0.2206 - val_accuracy: 0.9706
Epoch 45/50
82/82 [=====] - 4s 45ms/step - loss: 9.9180e-05 -
accuracy: 1.0000 - val_loss: 0.2198 - val_accuracy: 0.9718
Epoch 46/50
82/82 [=====] - 4s 46ms/step - loss: 9.3764e-05 -
accuracy: 1.0000 - val_loss: 0.2218 - val_accuracy: 0.9718
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 8.6116e-05 -
accuracy: 1.0000 - val_loss: 0.2242 - val_accuracy: 0.9718
Epoch 48/50
82/82 [=====] - 4s 45ms/step - loss: 7.9427e-05 -
accuracy: 1.0000 - val_loss: 0.2259 - val_accuracy: 0.9718
Epoch 49/50
82/82 [=====] - 4s 45ms/step - loss: 7.3735e-05 -
accuracy: 1.0000 - val_loss: 0.2266 - val_accuracy: 0.9718
Epoch 50/50
82/82 [=====] - 4s 45ms/step - loss: 6.9748e-05 -
accuracy: 1.0000 - val_loss: 0.2313 - val_accuracy: 0.9706
21/21 [=====] - 0s 11ms/step - loss: 0.1753 - accu
racy: 0.9648
Epoch 1/50
82/82 [=====] - 5s 50ms/step - loss: 0.5817 - accu
racy: 0.7226 - val_loss: 0.4686 - val_accuracy: 0.7316
Epoch 2/50
82/82 [=====] - 4s 46ms/step - loss: 0.3343 - accu
racy: 0.8563 - val_loss: 0.2208 - val_accuracy: 0.9093
Epoch 3/50
82/82 [=====] - 4s 46ms/step - loss: 0.2288 - accu

racy: 0.9041 - val_loss: 0.3098 - val_accuracy: 0.8566
Epoch 4/50
82/82 [=====] - 4s 45ms/step - loss: 0.1868 - accu
racy: 0.9243 - val_loss: 0.1422 - val_accuracy: 0.9547
Epoch 5/50
82/82 [=====] - 4s 46ms/step - loss: 0.1420 - accu
racy: 0.9473 - val_loss: 0.1598 - val_accuracy: 0.9412
Epoch 6/50
82/82 [=====] - 4s 49ms/step - loss: 0.1344 - accu
racy: 0.9499 - val_loss: 0.1212 - val_accuracy: 0.9596
Epoch 7/50
82/82 [=====] - 4s 45ms/step - loss: 0.1388 - accu
racy: 0.9484 - val_loss: 0.1099 - val_accuracy: 0.9657
Epoch 8/50
82/82 [=====] - 4s 46ms/step - loss: 0.1132 - accu
racy: 0.9583 - val_loss: 0.1171 - val_accuracy: 0.9596
Epoch 9/50
82/82 [=====] - 4s 45ms/step - loss: 0.1042 - accu
racy: 0.9610 - val_loss: 0.1026 - val_accuracy: 0.9657
Epoch 10/50
82/82 [=====] - 4s 46ms/step - loss: 0.1024 - accu
racy: 0.9652 - val_loss: 0.1525 - val_accuracy: 0.9485
Epoch 11/50
82/82 [=====] - 4s 47ms/step - loss: 0.1137 - accu
racy: 0.9553 - val_loss: 0.1103 - val_accuracy: 0.9620
Epoch 12/50
82/82 [=====] - 4s 46ms/step - loss: 0.0793 - accu
racy: 0.9713 - val_loss: 0.1024 - val_accuracy: 0.9632
Epoch 13/50
82/82 [=====] - 4s 45ms/step - loss: 0.0747 - accu
racy: 0.9748 - val_loss: 0.1192 - val_accuracy: 0.9620
Epoch 14/50
82/82 [=====] - 4s 46ms/step - loss: 0.0670 - accu
racy: 0.9778 - val_loss: 0.1073 - val_accuracy: 0.9681
Epoch 15/50
82/82 [=====] - 4s 45ms/step - loss: 0.0729 - accu
racy: 0.9744 - val_loss: 0.1037 - val_accuracy: 0.9694
Epoch 16/50
82/82 [=====] - 4s 48ms/step - loss: 0.0588 - accu
racy: 0.9767 - val_loss: 0.1141 - val_accuracy: 0.9657
Epoch 17/50
82/82 [=====] - 4s 45ms/step - loss: 0.0592 - accu
racy: 0.9805 - val_loss: 0.1092 - val_accuracy: 0.9669
Epoch 18/50
82/82 [=====] - 4s 46ms/step - loss: 0.0570 - accu
racy: 0.9782 - val_loss: 0.1168 - val_accuracy: 0.9657
Epoch 19/50
82/82 [=====] - 4s 46ms/step - loss: 0.0513 - accu
racy: 0.9801 - val_loss: 0.1599 - val_accuracy: 0.9473
Epoch 20/50
82/82 [=====] - 4s 45ms/step - loss: 0.0357 - accu
racy: 0.9897 - val_loss: 0.1125 - val_accuracy: 0.9743
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0282 - accu
racy: 0.9920 - val_loss: 0.1245 - val_accuracy: 0.9681
Epoch 22/50

82/82 [=====] - 4s 46ms/step - loss: 0.0262 - accuracy: 0.9924 - val_loss: 0.1427 - val_accuracy: 0.9706
Epoch 23/50
82/82 [=====] - 4s 46ms/step - loss: 0.0358 - accuracy: 0.9866 - val_loss: 0.1332 - val_accuracy: 0.9694
Epoch 24/50
82/82 [=====] - 4s 45ms/step - loss: 0.0362 - accuracy: 0.9836 - val_loss: 0.1640 - val_accuracy: 0.9596
Epoch 25/50
82/82 [=====] - 4s 47ms/step - loss: 0.0226 - accuracy: 0.9916 - val_loss: 0.1242 - val_accuracy: 0.9681
Epoch 26/50
82/82 [=====] - 4s 46ms/step - loss: 0.0150 - accuracy: 0.9958 - val_loss: 0.1487 - val_accuracy: 0.9694
Epoch 27/50
82/82 [=====] - 4s 46ms/step - loss: 0.0102 - accuracy: 0.9981 - val_loss: 0.1545 - val_accuracy: 0.9694
Epoch 28/50
82/82 [=====] - 4s 45ms/step - loss: 0.0134 - accuracy: 0.9943 - val_loss: 0.1509 - val_accuracy: 0.9718
Epoch 29/50
82/82 [=====] - 4s 46ms/step - loss: 0.0097 - accuracy: 0.9966 - val_loss: 0.1439 - val_accuracy: 0.9706
Epoch 30/50
82/82 [=====] - 4s 47ms/step - loss: 0.0049 - accuracy: 0.9989 - val_loss: 0.1609 - val_accuracy: 0.9694
Epoch 31/50
82/82 [=====] - 4s 45ms/step - loss: 0.0028 - accuracy: 0.9996 - val_loss: 0.1722 - val_accuracy: 0.9694
Epoch 32/50
82/82 [=====] - 4s 46ms/step - loss: 0.0026 - accuracy: 0.9992 - val_loss: 0.1787 - val_accuracy: 0.9694
Epoch 33/50
82/82 [=====] - 4s 46ms/step - loss: 6.0842e-04 - accuracy: 1.0000 - val_loss: 0.1875 - val_accuracy: 0.9706
Epoch 34/50
82/82 [=====] - 4s 46ms/step - loss: 4.0372e-04 - accuracy: 1.0000 - val_loss: 0.1971 - val_accuracy: 0.9681
Epoch 35/50
82/82 [=====] - 4s 46ms/step - loss: 3.1348e-04 - accuracy: 1.0000 - val_loss: 0.1989 - val_accuracy: 0.9694
Epoch 36/50
82/82 [=====] - 4s 46ms/step - loss: 2.8285e-04 - accuracy: 1.0000 - val_loss: 0.2031 - val_accuracy: 0.9694
Epoch 37/50
82/82 [=====] - 4s 47ms/step - loss: 2.4130e-04 - accuracy: 1.0000 - val_loss: 0.2112 - val_accuracy: 0.9681
Epoch 38/50
82/82 [=====] - 4s 46ms/step - loss: 2.0904e-04 - accuracy: 1.0000 - val_loss: 0.2114 - val_accuracy: 0.9694
Epoch 39/50
82/82 [=====] - 4s 45ms/step - loss: 1.8363e-04 - accuracy: 1.0000 - val_loss: 0.2167 - val_accuracy: 0.9694
Epoch 40/50
82/82 [=====] - 4s 45ms/step - loss: 1.6431e-04 - accuracy: 1.0000 - val_loss: 0.2191 - val_accuracy: 0.9694

```

Epoch 41/50
82/82 [=====] - 4s 48ms/step - loss: 1.4709e-04 -
accuracy: 1.0000 - val_loss: 0.2235 - val_accuracy: 0.9681
Epoch 42/50
82/82 [=====] - 4s 46ms/step - loss: 1.3450e-04 -
accuracy: 1.0000 - val_loss: 0.2247 - val_accuracy: 0.9694
Epoch 43/50
82/82 [=====] - 4s 46ms/step - loss: 1.2518e-04 -
accuracy: 1.0000 - val_loss: 0.2264 - val_accuracy: 0.9694
Epoch 44/50
82/82 [=====] - 4s 46ms/step - loss: 1.1320e-04 -
accuracy: 1.0000 - val_loss: 0.2300 - val_accuracy: 0.9694
Epoch 45/50
82/82 [=====] - 4s 45ms/step - loss: 1.0577e-04 -
accuracy: 1.0000 - val_loss: 0.2302 - val_accuracy: 0.9694
Epoch 46/50
82/82 [=====] - 4s 47ms/step - loss: 9.5166e-05 -
accuracy: 1.0000 - val_loss: 0.2327 - val_accuracy: 0.9694
Epoch 47/50
82/82 [=====] - 4s 46ms/step - loss: 8.9782e-05 -
accuracy: 1.0000 - val_loss: 0.2364 - val_accuracy: 0.9694
Epoch 48/50
82/82 [=====] - 4s 45ms/step - loss: 8.2793e-05 -
accuracy: 1.0000 - val_loss: 0.2399 - val_accuracy: 0.9694
Epoch 49/50
82/82 [=====] - 4s 46ms/step - loss: 7.6265e-05 -
accuracy: 1.0000 - val_loss: 0.2412 - val_accuracy: 0.9694

```

```
In [53]: save_result('CNN #2', results_train[1], results_test[1], cvs[0], cvs[1], cvs[2], cvs[3], cvs[4])
```

Out[53]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	CNN #2	1.000	0.949	0.957	0.969	0.956	0.965	0.945	0.009	0.958
0	CNN #1	0.947	0.931	0.933	0.943	0.934	0.943	0.917	0.011	0.934
0	Initial Model	0.730	0.730	0.937	0.732	0.735	0.709	0.725	0.095	0.768

Prediction for Confusion Matrix

```
In [134]: predictions = model2.predict(x = test_images, steps = 10, verbose=0)
```

```
In [135]: pred_check = np.round(predictions)
```

```
In [136]: pred_check = pred_check[:]
pred_check = pred_check.flatten()
pred_check
```

Out[136]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)

```
In [137]: test_check = test_labels[:,0]
test_check
```

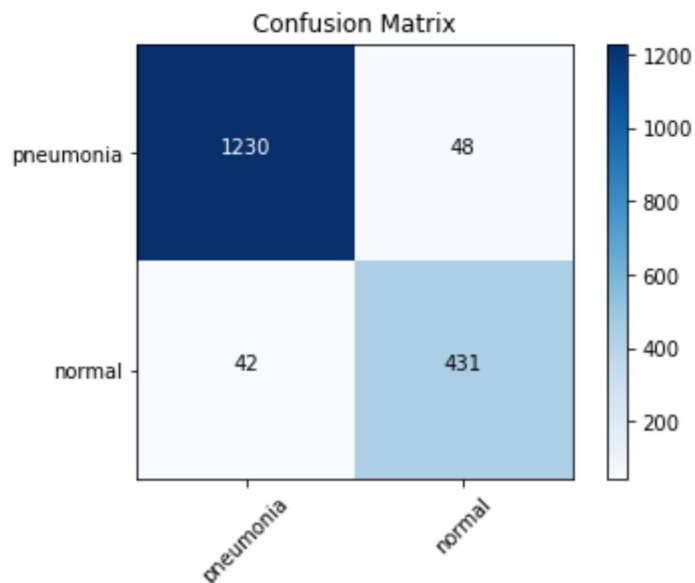
```
Out[137]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [138]: cm = confusion_matrix(y_true=test_check, y_pred=pred_check)
```

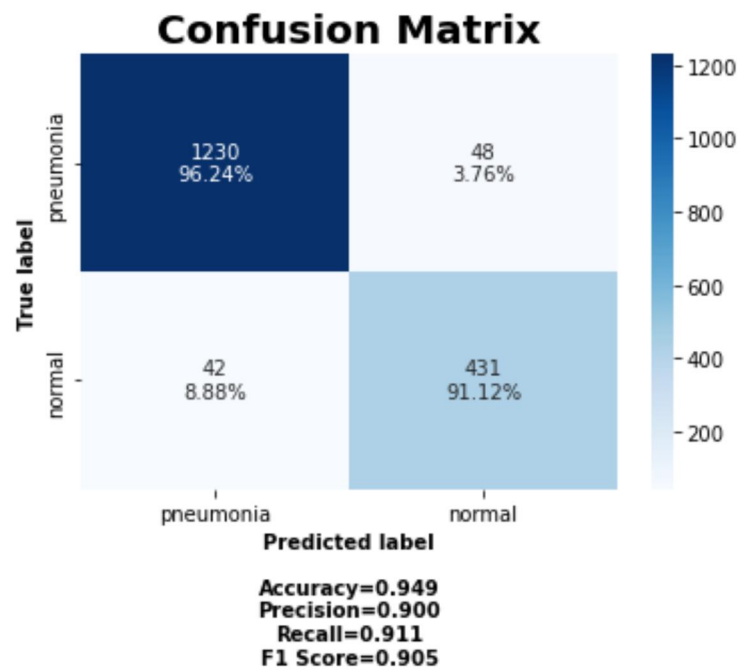
```
In [139]: cm_plot_labels = ['pneumonia', 'normal']
plot_confusion_matrix(cm=cm, classes=cm_plot_labels, title='Confusion Matrix')
```

Confusion matrix, without normalization

```
[[1230  48]
 [ 42 431]]
```



```
In [140]: cm_plot_labels = ['pneumonia', 'normal']  
make_confusion_matrix(cm, categories = cm_plot_labels, title='Confusion Matrix')
```



CNN Model 3

For this model, I add two layers of batch normalization and a 10% dropout. Also added one more layer of Conv2D and MaxPooling.

```
In [60]: model 3 = models.Sequential()

model 3.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
model 3.add(layers.MaxPooling2D((2, 2)))

model 3.add(layers.Conv2D(32, (4, 4), activation='relu'))
model 3.add(BatchNormalization())
model 3.add(layers.MaxPooling2D((2, 2)))

model 3.add(layers.Conv2D(64, (3, 3), activation='relu'))
model 3.add(layers.MaxPooling2D((2, 2)))

model 3.add(layers.Conv2D(128, (3, 3), activation='relu'))
model 3.add(BatchNormalization())
model 3.add(layers.MaxPooling2D((2, 2)))

model 3.add(layers.Flatten())
model 3.add(layers.Dense(64, activation='relu'))
model 3.add(Dropout(0.1))
model 3.add(layers.Dense(1, activation='sigmoid'))

model 3.compile(loss='binary_crossentropy',
                 optimizer="adam",
                 metrics=['accuracy'])
```

In [61]: model 3. summary()

Model: "sequential_13"

Layer (type)	Output Shape	Param #
=====		
conv2d_42 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_42 (MaxPooling)	(None, 31, 31, 32)	0
conv2d_43 (Conv2D)	(None, 28, 28, 32)	16416
batch_normalization (BatchNo	(None, 28, 28, 32)	128
max_pooling2d_43 (MaxPooling)	(None, 14, 14, 32)	0
conv2d_44 (Conv2D)	(None, 12, 12, 64)	18496
max_pooling2d_44 (MaxPooling)	(None, 6, 6, 64)	0
conv2d_45 (Conv2D)	(None, 4, 4, 128)	73856
batch_normalization_1 (Batch	(None, 4, 4, 128)	512
max_pooling2d_45 (MaxPooling)	(None, 2, 2, 128)	0
flatten_12 (Flatten)	(None, 512)	0
dense_48 (Dense)	(None, 64)	32832
dropout (Dropout)	(None, 64)	0
dense_49 (Dense)	(None, 1)	65
=====		
Total params: 143,201		
Trainable params: 142,881		
Non-trainable params: 320		

In [62]: print(f'train size {train_size}, validation size {validation_size}')

train size 3271, validation size 816

[illegible]

Epoch 1/50
103/103 [=====] - 5s 47ms/step - loss: 0.2068 - accuracy: 0.9220 - val_loss: 0.5385 - val_accuracy: 0.7304
Epoch 2/50
103/103 [=====] - 5s 46ms/step - loss: 0.1576 - accuracy: 0.9416 - val_loss: 0.5346 - val_accuracy: 0.7304
Epoch 3/50
103/103 [=====] - 5s 45ms/step - loss: 0.1373 - accuracy: 0.9508 - val_loss: 0.2866 - val_accuracy: 0.9485
Epoch 4/50
103/103 [=====] - 5s 45ms/step - loss: 0.1156 - accuracy: 0.9569 - val_loss: 0.2079 - val_accuracy: 0.9314
Epoch 5/50
103/103 [=====] - 5s 45ms/step - loss: 0.1116 - accuracy: 0.9609 - val_loss: 0.5114 - val_accuracy: 0.7757
Epoch 6/50
103/103 [=====] - 5s 46ms/step - loss: 0.0881 - accuracy: 0.9667 - val_loss: 0.1449 - val_accuracy: 0.9473
Epoch 7/50
103/103 [=====] - 5s 44ms/step - loss: 0.0742 - accuracy: 0.9765 - val_loss: 0.2381 - val_accuracy: 0.9154
Epoch 8/50
103/103 [=====] - 5s 45ms/step - loss: 0.0916 - accuracy: 0.9670 - val_loss: 2.1743 - val_accuracy: 0.7316
Epoch 9/50
103/103 [=====] - 5s 44ms/step - loss: 0.0794 - accuracy: 0.9722 - val_loss: 0.2695 - val_accuracy: 0.9081
Epoch 10/50
103/103 [=====] - 5s 44ms/step - loss: 0.0687 - accuracy: 0.9752 - val_loss: 0.7832 - val_accuracy: 0.8039
Epoch 11/50
103/103 [=====] - 5s 46ms/step - loss: 0.0614 - accuracy: 0.9765 - val_loss: 0.1615 - val_accuracy: 0.9461
Epoch 12/50
103/103 [=====] - 4s 44ms/step - loss: 0.0778 - accuracy: 0.9713 - val_loss: 0.1487 - val_accuracy: 0.9510
Epoch 13/50
103/103 [=====] - 5s 44ms/step - loss: 0.0392 - accuracy: 0.9856 - val_loss: 0.4601 - val_accuracy: 0.8958
Epoch 14/50
103/103 [=====] - 5s 44ms/step - loss: 0.0669 - accuracy: 0.9783 - val_loss: 1.2939 - val_accuracy: 0.5637
Epoch 15/50
103/103 [=====] - 5s 44ms/step - loss: 0.0544 - accuracy: 0.9792 - val_loss: 0.1586 - val_accuracy: 0.9449
Epoch 16/50
103/103 [=====] - 5s 46ms/step - loss: 0.0327 - accuracy: 0.9865 - val_loss: 0.1481 - val_accuracy: 0.9583
Epoch 17/50
103/103 [=====] - 4s 44ms/step - loss: 0.0263 - accuracy: 0.9911 - val_loss: 0.2839 - val_accuracy: 0.8995
Epoch 18/50
103/103 [=====] - 5s 44ms/step - loss: 0.0234 - accuracy: 0.9924 - val_loss: 0.2125 - val_accuracy: 0.9350
Epoch 19/50
103/103 [=====] - 5s 44ms/step - loss: 0.0128 - ac

curacy: 0.9957 - val_loss: 1.1072 - val_accuracy: 0.8174
Epoch 20/50
103/103 [=====] - 5s 46ms/step - loss: 0.0186 - ac
curacy: 0.9939 - val_loss: 0.3067 - val_accuracy: 0.9326
Epoch 21/50
103/103 [=====] - 4s 44ms/step - loss: 0.0061 - ac
curacy: 0.9982 - val_loss: 0.2895 - val_accuracy: 0.9498
Epoch 22/50
103/103 [=====] - 5s 44ms/step - loss: 0.0342 - ac
curacy: 0.9884 - val_loss: 0.2224 - val_accuracy: 0.9326
Epoch 23/50
103/103 [=====] - 5s 44ms/step - loss: 0.0278 - ac
curacy: 0.9896 - val_loss: 0.2745 - val_accuracy: 0.9252
Epoch 24/50
103/103 [=====] - 4s 44ms/step - loss: 0.0104 - ac
curacy: 0.9972 - val_loss: 0.3317 - val_accuracy: 0.9032
Epoch 25/50
103/103 [=====] - 5s 46ms/step - loss: 0.0070 - ac
curacy: 0.9979 - val_loss: 0.2710 - val_accuracy: 0.9240
Epoch 26/50
103/103 [=====] - 4s 44ms/step - loss: 0.0164 - ac
curacy: 0.9951 - val_loss: 0.6644 - val_accuracy: 0.8922
Epoch 27/50
103/103 [=====] - 4s 44ms/step - loss: 0.0093 - ac
curacy: 0.9972 - val_loss: 0.6888 - val_accuracy: 0.8456
Epoch 28/50
103/103 [=====] - 5s 44ms/step - loss: 0.0501 - ac
curacy: 0.9814 - val_loss: 0.3646 - val_accuracy: 0.8836
Epoch 29/50
103/103 [=====] - 5s 44ms/step - loss: 0.0188 - ac
curacy: 0.9933 - val_loss: 0.1814 - val_accuracy: 0.9498
Epoch 30/50
103/103 [=====] - 5s 46ms/step - loss: 0.0070 - ac
curacy: 0.9982 - val_loss: 0.1787 - val_accuracy: 0.9571
Epoch 31/50
103/103 [=====] - 4s 44ms/step - loss: 0.0030 - ac
curacy: 0.9991 - val_loss: 0.7890 - val_accuracy: 0.7855
Epoch 32/50
103/103 [=====] - 5s 44ms/step - loss: 0.0016 - ac
curacy: 0.9997 - val_loss: 0.2327 - val_accuracy: 0.9485
Epoch 33/50
103/103 [=====] - 5s 44ms/step - loss: 6.2217e-04
- accuracy: 1.0000 - val_loss: 0.1895 - val_accuracy: 0.9583
Epoch 34/50
103/103 [=====] - 5s 46ms/step - loss: 4.0277e-04
- accuracy: 1.0000 - val_loss: 0.1950 - val_accuracy: 0.9632
Epoch 35/50
103/103 [=====] - 5s 44ms/step - loss: 2.1403e-04
- accuracy: 1.0000 - val_loss: 0.2232 - val_accuracy: 0.9632
Epoch 36/50
103/103 [=====] - 4s 43ms/step - loss: 1.3092e-04
- accuracy: 1.0000 - val_loss: 0.2193 - val_accuracy: 0.9608
Epoch 37/50
103/103 [=====] - 4s 44ms/step - loss: 1.3321e-04
- accuracy: 1.0000 - val_loss: 0.2187 - val_accuracy: 0.9620
Epoch 38/50

```

103/103 [=====] - 5s 44ms/step - loss: 8.9893e-05
- accuracy: 1.0000 - val_loss: 0.2210 - val_accuracy: 0.9620
Epoch 39/50
103/103 [=====] - 5s 45ms/step - loss: 3.4146e-04
- accuracy: 0.9997 - val_loss: 0.2190 - val_accuracy: 0.9632
Epoch 40/50
103/103 [=====] - 5s 44ms/step - loss: 2.2384e-04
- accuracy: 1.0000 - val_loss: 0.2222 - val_accuracy: 0.9632
Epoch 41/50
103/103 [=====] - 4s 44ms/step - loss: 9.5846e-05
- accuracy: 1.0000 - val_loss: 0.2353 - val_accuracy: 0.9608
Epoch 42/50
103/103 [=====] - 5s 44ms/step - loss: 5.6376e-05
- accuracy: 1.0000 - val_loss: 0.2322 - val_accuracy: 0.9620
Epoch 43/50
103/103 [=====] - 4s 44ms/step - loss: 0.0014 - ac
curacy: 0.9994 - val_loss: 0.4423 - val_accuracy: 0.9436
Epoch 44/50
103/103 [=====] - 5s 46ms/step - loss: 0.0872 - ac
curacy: 0.9673 - val_loss: 3.4656 - val_accuracy: 0.7316
Epoch 45/50
103/103 [=====] - 4s 44ms/step - loss: 0.0591 - ac
curacy: 0.9771 - val_loss: 0.2383 - val_accuracy: 0.9350
Epoch 46/50
103/103 [=====] - 5s 44ms/step - loss: 0.0253 - ac
curacy: 0.9908 - val_loss: 0.1470 - val_accuracy: 0.9547
Epoch 47/50
103/103 [=====] - 5s 44ms/step - loss: 0.0106 - ac
curacy: 0.9960 - val_loss: 0.2057 - val_accuracy: 0.9596
Epoch 48/50
103/103 [=====] - 5s 44ms/step - loss: 0.0076 - ac
curacy: 0.9985 - val_loss: 0.1377 - val_accuracy: 0.9583
Epoch 49/50
103/103 [=====] - 5s 45ms/step - loss: 0.0019 - ac
curacy: 1.0000 - val_loss: 0.1687 - val_accuracy: 0.9620
Epoch 50/50
103/103 [=====] - 5s 44ms/step - loss: 6.8034e-04
- accuracy: 1.0000 - val_loss: 0.1920 - val_accuracy: 0.9620

```

```
In [64]: results_train = model3.evaluate(train_images, train_y)
```

```

103/103 [=====] - 1s 12ms/step - loss: 6.3973e-04
- accuracy: 1.0000

```

```
In [65]: results_test = model3.evaluate(test_images, test_y)
```

```

55/55 [=====] - 1s 11ms/step - loss: 0.2682 - accu
racy: 0.9475

```

```
In [66]: results_train
```

```
Out[66]: [0.0006397273973561823, 1.0]
```

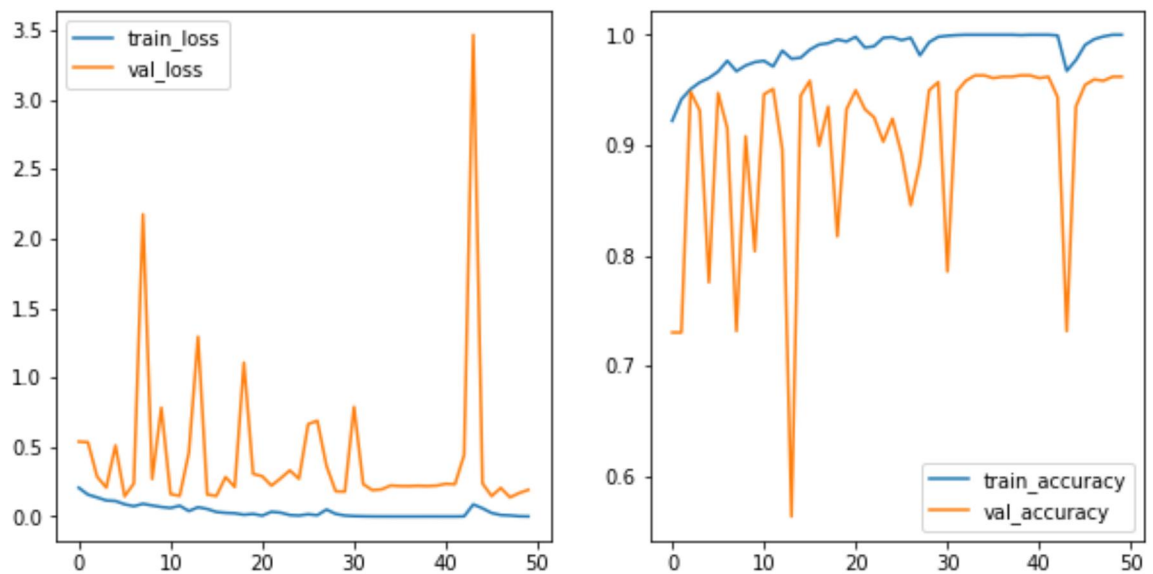
```
In [67]: results_test
```

```
Out[67]: [0.26823556423187256, 0.9474586248397827]
```

```
In [68]: train_loss = history.history['loss']
train_acc = history.history['accuracy']
val_loss = history.history['val_loss']
val_acc = history.history['val_accuracy']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=history.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=history.epoch, y=train_acc, ax=ax2, label='train_accuracy')
sns.lineplot(x=history.epoch, y=val_loss, ax=ax1, label='val_loss')
sns.lineplot(x=history.epoch, y=val_acc, ax=ax2, label='val_accuracy')
```

```
Out[68]: <AxesSubplot: >
```




```
In [71]: cvs = cross_val_score(keras_model4, train_images , train_y, cv=5)
```

Epoch 1/50
82/82 [=====] - 5s 50ms/step - loss: 0.2359 - accuracy: 0.9136 - val_loss: 0.5173 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 48ms/step - loss: 0.1394 - accuracy: 0.9476 - val_loss: 0.6349 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 49ms/step - loss: 0.1302 - accuracy: 0.9549 - val_loss: 0.5756 - val_accuracy: 0.7414
Epoch 4/50
82/82 [=====] - 4s 47ms/step - loss: 0.1256 - accuracy: 0.9576 - val_loss: 0.3394 - val_accuracy: 0.7990
Epoch 5/50
82/82 [=====] - 4s 48ms/step - loss: 0.1178 - accuracy: 0.9587 - val_loss: 0.2045 - val_accuracy: 0.9179
Epoch 6/50
82/82 [=====] - 4s 48ms/step - loss: 0.0919 - accuracy: 0.9633 - val_loss: 0.2068 - val_accuracy: 0.9179
Epoch 7/50
82/82 [=====] - 4s 48ms/step - loss: 0.0761 - accuracy: 0.9721 - val_loss: 0.1113 - val_accuracy: 0.9608
Epoch 8/50
82/82 [=====] - 4s 47ms/step - loss: 0.0671 - accuracy: 0.9744 - val_loss: 0.3842 - val_accuracy: 0.8088
Epoch 9/50
82/82 [=====] - 4s 47ms/step - loss: 0.0663 - accuracy: 0.9752 - val_loss: 0.1197 - val_accuracy: 0.9608
Epoch 10/50
82/82 [=====] - 4s 53ms/step - loss: 0.0496 - accuracy: 0.9832 - val_loss: 1.1718 - val_accuracy: 0.7635
Epoch 11/50
82/82 [=====] - 4s 53ms/step - loss: 0.0474 - accuracy: 0.9809 - val_loss: 0.6967 - val_accuracy: 0.8309
Epoch 12/50
82/82 [=====] - 4s 50ms/step - loss: 0.0399 - accuracy: 0.9847 - val_loss: 0.2641 - val_accuracy: 0.9338
Epoch 13/50
82/82 [=====] - 4s 50ms/step - loss: 0.0212 - accuracy: 0.9927 - val_loss: 0.1860 - val_accuracy: 0.9412
Epoch 14/50
82/82 [=====] - 4s 50ms/step - loss: 0.0134 - accuracy: 0.9954 - val_loss: 0.1389 - val_accuracy: 0.9534
Epoch 15/50
82/82 [=====] - 4s 51ms/step - loss: 0.0071 - accuracy: 0.9977 - val_loss: 0.2691 - val_accuracy: 0.9363
Epoch 16/50
82/82 [=====] - 4s 50ms/step - loss: 0.0039 - accuracy: 0.9996 - val_loss: 0.3312 - val_accuracy: 0.8958
Epoch 17/50
82/82 [=====] - 4s 50ms/step - loss: 0.0035 - accuracy: 0.9996 - val_loss: 0.1873 - val_accuracy: 0.9596
Epoch 18/50
82/82 [=====] - 4s 52ms/step - loss: 0.0288 - accuracy: 0.9912 - val_loss: 0.8084 - val_accuracy: 0.8309
Epoch 19/50
82/82 [=====] - 4s 49ms/step - loss: 0.0907 - accuracy:

racy: 0.9648 - val_loss: 0.2378 - val_accuracy: 0.9277
Epoch 20/50
82/82 [=====] - 4s 49ms/step - loss: 0.0392 - accuracy: 0.9851 - val_loss: 0.7862 - val_accuracy: 0.8260
Epoch 21/50
82/82 [=====] - 4s 50ms/step - loss: 0.0186 - accuracy: 0.9935 - val_loss: 0.3159 - val_accuracy: 0.9314
Epoch 22/50
82/82 [=====] - 4s 49ms/step - loss: 0.0232 - accuracy: 0.9916 - val_loss: 0.2008 - val_accuracy: 0.9449
Epoch 23/50
82/82 [=====] - 4s 49ms/step - loss: 0.0272 - accuracy: 0.9889 - val_loss: 0.6963 - val_accuracy: 0.8027
Epoch 24/50
82/82 [=====] - 4s 49ms/step - loss: 0.0132 - accuracy: 0.9966 - val_loss: 0.1695 - val_accuracy: 0.9510
Epoch 25/50
82/82 [=====] - 4s 49ms/step - loss: 0.0091 - accuracy: 0.9966 - val_loss: 0.2204 - val_accuracy: 0.9485
Epoch 26/50
82/82 [=====] - 4s 50ms/step - loss: 0.0231 - accuracy: 0.9916 - val_loss: 0.1665 - val_accuracy: 0.9583
Epoch 27/50
82/82 [=====] - 4s 52ms/step - loss: 0.0130 - accuracy: 0.9958 - val_loss: 0.2168 - val_accuracy: 0.9583
Epoch 28/50
82/82 [=====] - 4s 49ms/step - loss: 0.0251 - accuracy: 0.9920 - val_loss: 1.3862 - val_accuracy: 0.6728
Epoch 29/50
82/82 [=====] - 4s 50ms/step - loss: 0.0182 - accuracy: 0.9935 - val_loss: 0.1824 - val_accuracy: 0.9449
Epoch 30/50
82/82 [=====] - 4s 49ms/step - loss: 0.0025 - accuracy: 0.9996 - val_loss: 0.1820 - val_accuracy: 0.9645
Epoch 31/50
82/82 [=====] - 4s 51ms/step - loss: 4.0266e-04 - accuracy: 1.0000 - val_loss: 0.2028 - val_accuracy: 0.9632
Epoch 32/50
82/82 [=====] - 4s 50ms/step - loss: 2.7150e-04 - accuracy: 1.0000 - val_loss: 0.2039 - val_accuracy: 0.9669
Epoch 33/50
82/82 [=====] - 4s 49ms/step - loss: 2.1268e-04 - accuracy: 1.0000 - val_loss: 0.2081 - val_accuracy: 0.9694
Epoch 34/50
82/82 [=====] - 4s 50ms/step - loss: 1.3574e-04 - accuracy: 1.0000 - val_loss: 0.2104 - val_accuracy: 0.9681
Epoch 35/50
82/82 [=====] - 4s 49ms/step - loss: 1.3742e-04 - accuracy: 1.0000 - val_loss: 0.2158 - val_accuracy: 0.9657
Epoch 36/50
82/82 [=====] - 4s 51ms/step - loss: 1.4442e-04 - accuracy: 1.0000 - val_loss: 0.2191 - val_accuracy: 0.9657
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 9.7308e-05 - accuracy: 1.0000 - val_loss: 0.2238 - val_accuracy: 0.9657
Epoch 38/50

82/82 [=====] - 4s 49ms/step - loss: 8.0381e-05 - accuracy: 1.0000 - val_loss: 0.2230 - val_accuracy: 0.9669
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 6.5652e-05 - accuracy: 1.0000 - val_loss: 0.2235 - val_accuracy: 0.9669
Epoch 40/50
82/82 [=====] - 4s 49ms/step - loss: 8.4616e-05 - accuracy: 1.0000 - val_loss: 0.2306 - val_accuracy: 0.9669
Epoch 41/50
82/82 [=====] - 4s 49ms/step - loss: 7.0354e-05 - accuracy: 1.0000 - val_loss: 0.2269 - val_accuracy: 0.9681
Epoch 42/50
82/82 [=====] - 4s 51ms/step - loss: 7.9278e-05 - accuracy: 1.0000 - val_loss: 0.2311 - val_accuracy: 0.9657
Epoch 43/50
82/82 [=====] - 4s 50ms/step - loss: 5.6180e-05 - accuracy: 1.0000 - val_loss: 0.2353 - val_accuracy: 0.9657
Epoch 44/50
82/82 [=====] - 4s 50ms/step - loss: 4.4311e-05 - accuracy: 1.0000 - val_loss: 0.2373 - val_accuracy: 0.9669
Epoch 45/50
82/82 [=====] - 4s 48ms/step - loss: 3.9376e-05 - accuracy: 1.0000 - val_loss: 0.2406 - val_accuracy: 0.9657
Epoch 46/50
82/82 [=====] - 4s 51ms/step - loss: 3.9005e-05 - accuracy: 1.0000 - val_loss: 0.2407 - val_accuracy: 0.9657
Epoch 47/50
82/82 [=====] - 4s 50ms/step - loss: 3.1072e-05 - accuracy: 1.0000 - val_loss: 0.2420 - val_accuracy: 0.9694
Epoch 48/50
82/82 [=====] - 4s 49ms/step - loss: 3.1112e-05 - accuracy: 1.0000 - val_loss: 0.2444 - val_accuracy: 0.9681
Epoch 49/50
82/82 [=====] - 4s 50ms/step - loss: 3.1058e-05 - accuracy: 1.0000 - val_loss: 0.2448 - val_accuracy: 0.9694
Epoch 50/50
82/82 [=====] - 4s 49ms/step - loss: 3.1710e-05 - accuracy: 1.0000 - val_loss: 0.2474 - val_accuracy: 0.9657
21/21 [=====] - 0s 12ms/step - loss: 0.2283 - accuracy: 0.9573
Epoch 1/50
82/82 [=====] - 5s 53ms/step - loss: 0.2248 - accuracy: 0.9190 - val_loss: 0.5443 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 49ms/step - loss: 0.1540 - accuracy: 0.9404 - val_loss: 0.4759 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 49ms/step - loss: 0.1340 - accuracy: 0.9431 - val_loss: 0.5053 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 49ms/step - loss: 0.1114 - accuracy: 0.9633 - val_loss: 0.3407 - val_accuracy: 0.7929
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.0928 - accuracy: 0.9671 - val_loss: 0.3714 - val_accuracy: 0.7953
Epoch 6/50

82/82 [=====] - 4s 51ms/step - loss: 0.0808 - accuracy: 0.9729 - val_loss: 0.2191 - val_accuracy: 0.9449
Epoch 7/50
82/82 [=====] - 4s 49ms/step - loss: 0.0727 - accuracy: 0.9733 - val_loss: 0.1860 - val_accuracy: 0.9326
Epoch 8/50
82/82 [=====] - 4s 50ms/step - loss: 0.0622 - accuracy: 0.9771 - val_loss: 0.1298 - val_accuracy: 0.9547
Epoch 9/50
82/82 [=====] - 4s 49ms/step - loss: 0.0447 - accuracy: 0.9855 - val_loss: 0.3065 - val_accuracy: 0.8542
Epoch 10/50
82/82 [=====] - 4s 49ms/step - loss: 0.0462 - accuracy: 0.9836 - val_loss: 0.2494 - val_accuracy: 0.9105
Epoch 11/50
82/82 [=====] - 4s 51ms/step - loss: 0.0583 - accuracy: 0.9786 - val_loss: 0.4336 - val_accuracy: 0.8027
Epoch 12/50
82/82 [=====] - 4s 49ms/step - loss: 0.0335 - accuracy: 0.9897 - val_loss: 0.1666 - val_accuracy: 0.9400
Epoch 13/50
82/82 [=====] - 4s 49ms/step - loss: 0.0295 - accuracy: 0.9889 - val_loss: 0.1798 - val_accuracy: 0.9387
Epoch 14/50
82/82 [=====] - 4s 49ms/step - loss: 0.0313 - accuracy: 0.9901 - val_loss: 4.1096 - val_accuracy: 0.7365
Epoch 15/50
82/82 [=====] - 4s 49ms/step - loss: 0.0339 - accuracy: 0.9870 - val_loss: 0.1466 - val_accuracy: 0.9534
Epoch 16/50
82/82 [=====] - 4s 51ms/step - loss: 0.0252 - accuracy: 0.9893 - val_loss: 0.3183 - val_accuracy: 0.9240
Epoch 17/50
82/82 [=====] - 4s 49ms/step - loss: 0.0241 - accuracy: 0.9901 - val_loss: 0.8215 - val_accuracy: 0.8615
Epoch 18/50
82/82 [=====] - 4s 49ms/step - loss: 0.0321 - accuracy: 0.9874 - val_loss: 7.7294 - val_accuracy: 0.7304
Epoch 19/50
82/82 [=====] - 4s 49ms/step - loss: 0.0381 - accuracy: 0.9859 - val_loss: 0.1387 - val_accuracy: 0.9571
Epoch 20/50
82/82 [=====] - 4s 49ms/step - loss: 0.0198 - accuracy: 0.9931 - val_loss: 0.1242 - val_accuracy: 0.9620
Epoch 21/50
82/82 [=====] - 4s 50ms/step - loss: 0.0086 - accuracy: 0.9969 - val_loss: 0.1658 - val_accuracy: 0.9559
Epoch 22/50
82/82 [=====] - 4s 51ms/step - loss: 0.0117 - accuracy: 0.9950 - val_loss: 3.7782 - val_accuracy: 0.7537
Epoch 23/50
82/82 [=====] - 4s 49ms/step - loss: 0.0189 - accuracy: 0.9920 - val_loss: 0.2544 - val_accuracy: 0.9412
Epoch 24/50
82/82 [=====] - 4s 49ms/step - loss: 0.0432 - accuracy: 0.9840 - val_loss: 2.4078 - val_accuracy: 0.7426

Epoch 25/50
82/82 [=====] - 4s 49ms/step - loss: 0.0242 - accuracy: 0.9908 - val_loss: 0.2592 - val_accuracy: 0.9240
Epoch 26/50
82/82 [=====] - 4s 49ms/step - loss: 0.0135 - accuracy: 0.9950 - val_loss: 0.3456 - val_accuracy: 0.9044
Epoch 27/50
82/82 [=====] - 4s 49ms/step - loss: 0.0042 - accuracy: 0.9992 - val_loss: 0.1672 - val_accuracy: 0.9645
Epoch 28/50
82/82 [=====] - 4s 51ms/step - loss: 0.0021 - accuracy: 0.9992 - val_loss: 0.2009 - val_accuracy: 0.9547
Epoch 29/50
82/82 [=====] - 4s 50ms/step - loss: 0.0012 - accuracy: 1.0000 - val_loss: 0.1910 - val_accuracy: 0.9608
Epoch 30/50
82/82 [=====] - 4s 49ms/step - loss: 3.3376e-04 - accuracy: 1.0000 - val_loss: 0.1973 - val_accuracy: 0.9645
Epoch 31/50
82/82 [=====] - 4s 49ms/step - loss: 2.0395e-04 - accuracy: 1.0000 - val_loss: 0.2048 - val_accuracy: 0.9632
Epoch 32/50
82/82 [=====] - 4s 51ms/step - loss: 2.1752e-04 - accuracy: 1.0000 - val_loss: 0.2189 - val_accuracy: 0.9620
Epoch 33/50
82/82 [=====] - 4s 49ms/step - loss: 1.5807e-04 - accuracy: 1.0000 - val_loss: 0.2132 - val_accuracy: 0.9632
Epoch 34/50
82/82 [=====] - 4s 49ms/step - loss: 1.4281e-04 - accuracy: 1.0000 - val_loss: 0.2112 - val_accuracy: 0.9632
Epoch 35/50
82/82 [=====] - 4s 49ms/step - loss: 9.4155e-05 - accuracy: 1.0000 - val_loss: 0.2175 - val_accuracy: 0.9645
Epoch 36/50
82/82 [=====] - 4s 50ms/step - loss: 7.5664e-05 - accuracy: 1.0000 - val_loss: 0.2236 - val_accuracy: 0.9645
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 6.9416e-05 - accuracy: 1.0000 - val_loss: 0.2262 - val_accuracy: 0.9657
Epoch 38/50
82/82 [=====] - 4s 50ms/step - loss: 7.5851e-05 - accuracy: 1.0000 - val_loss: 0.2159 - val_accuracy: 0.9645
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 7.8009e-05 - accuracy: 1.0000 - val_loss: 0.2226 - val_accuracy: 0.9632
Epoch 40/50
82/82 [=====] - 4s 49ms/step - loss: 4.1865e-05 - accuracy: 1.0000 - val_loss: 0.2259 - val_accuracy: 0.9632
Epoch 41/50
82/82 [=====] - 4s 51ms/step - loss: 4.6224e-05 - accuracy: 1.0000 - val_loss: 0.2274 - val_accuracy: 0.9645
Epoch 42/50
82/82 [=====] - 4s 49ms/step - loss: 4.3539e-05 - accuracy: 1.0000 - val_loss: 0.2310 - val_accuracy: 0.9632
Epoch 43/50
82/82 [=====] - 4s 49ms/step - loss: 4.1878e-05 -

accuracy: 1.0000 - val_loss: 0.2313 - val_accuracy: 0.9632
Epoch 44/50
82/82 [=====] - 4s 49ms/step - loss: 2.3801e-05 -
accuracy: 1.0000 - val_loss: 0.2320 - val_accuracy: 0.9632
Epoch 45/50
82/82 [=====] - 4s 49ms/step - loss: 2.2926e-05 -
accuracy: 1.0000 - val_loss: 0.2336 - val_accuracy: 0.9645
Epoch 46/50
82/82 [=====] - 4s 49ms/step - loss: 2.9875e-05 -
accuracy: 1.0000 - val_loss: 0.2314 - val_accuracy: 0.9657
Epoch 47/50
82/82 [=====] - 4s 51ms/step - loss: 3.8097e-05 -
accuracy: 1.0000 - val_loss: 0.2420 - val_accuracy: 0.9657
Epoch 48/50
82/82 [=====] - 4s 49ms/step - loss: 2.2402e-05 -
accuracy: 1.0000 - val_loss: 0.2379 - val_accuracy: 0.9645
Epoch 49/50
82/82 [=====] - 4s 48ms/step - loss: 2.5539e-05 -
accuracy: 1.0000 - val_loss: 0.2372 - val_accuracy: 0.9645
Epoch 50/50
82/82 [=====] - 4s 50ms/step - loss: 2.2974e-05 -
accuracy: 1.0000 - val_loss: 0.2391 - val_accuracy: 0.9645
21/21 [=====] - 1s 13ms/step - loss: 0.1438 - accu
racy: 0.9694
Epoch 1/50
82/82 [=====] - 5s 53ms/step - loss: 0.2380 - accu
racy: 0.9056 - val_loss: 0.5480 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 51ms/step - loss: 0.1308 - accu
racy: 0.9503 - val_loss: 0.5223 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 49ms/step - loss: 0.1287 - accu
racy: 0.9496 - val_loss: 0.9058 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 49ms/step - loss: 0.1069 - accu
racy: 0.9614 - val_loss: 0.9747 - val_accuracy: 0.7304
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.0853 - accu
racy: 0.9690 - val_loss: 1.2974 - val_accuracy: 0.7304
Epoch 6/50
82/82 [=====] - 4s 49ms/step - loss: 0.0829 - accu
racy: 0.9694 - val_loss: 0.6479 - val_accuracy: 0.7512
Epoch 7/50
82/82 [=====] - 4s 51ms/step - loss: 0.0699 - accu
racy: 0.9717 - val_loss: 2.7622 - val_accuracy: 0.7304
Epoch 8/50
82/82 [=====] - 4s 49ms/step - loss: 0.0567 - accu
racy: 0.9775 - val_loss: 1.8267 - val_accuracy: 0.3199
Epoch 9/50
82/82 [=====] - 4s 49ms/step - loss: 0.0755 - accu
racy: 0.9713 - val_loss: 0.2041 - val_accuracy: 0.9179
Epoch 10/50
82/82 [=====] - 4s 49ms/step - loss: 0.0667 - accu
racy: 0.9748 - val_loss: 0.1190 - val_accuracy: 0.9608
Epoch 11/50
82/82 [=====] - 4s 51ms/step - loss: 0.0305 - accu

racy: 0.9882 - val_loss: 0.1672 - val_accuracy: 0.9412
Epoch 12/50
82/82 [=====] - 4s 49ms/step - loss: 0.0251 - accu
racy: 0.9904 - val_loss: 0.1107 - val_accuracy: 0.9657
Epoch 13/50
82/82 [=====] - 4s 49ms/step - loss: 0.0177 - accu
racy: 0.9950 - val_loss: 0.4529 - val_accuracy: 0.8934
Epoch 14/50
82/82 [=====] - 4s 49ms/step - loss: 0.0121 - accu
racy: 0.9969 - val_loss: 0.3245 - val_accuracy: 0.8860
Epoch 15/50
82/82 [=====] - 4s 50ms/step - loss: 0.0133 - accu
racy: 0.9962 - val_loss: 0.3054 - val_accuracy: 0.9436
Epoch 16/50
82/82 [=====] - 4s 49ms/step - loss: 0.0271 - accu
racy: 0.9908 - val_loss: 0.3146 - val_accuracy: 0.9412
Epoch 17/50
82/82 [=====] - 4s 51ms/step - loss: 0.0223 - accu
racy: 0.9920 - val_loss: 1.2965 - val_accuracy: 0.7806
Epoch 18/50
82/82 [=====] - 4s 49ms/step - loss: 0.0158 - accu
racy: 0.9943 - val_loss: 0.2481 - val_accuracy: 0.9314
Epoch 19/50
82/82 [=====] - 4s 49ms/step - loss: 0.0154 - accu
racy: 0.9935 - val_loss: 0.1495 - val_accuracy: 0.9473
Epoch 20/50
82/82 [=====] - 4s 50ms/step - loss: 0.0174 - accu
racy: 0.9943 - val_loss: 0.4614 - val_accuracy: 0.8958
Epoch 21/50
82/82 [=====] - 4s 49ms/step - loss: 0.0161 - accu
racy: 0.9935 - val_loss: 0.2412 - val_accuracy: 0.9498
Epoch 22/50
82/82 [=====] - 4s 50ms/step - loss: 0.0107 - accu
racy: 0.9966 - val_loss: 0.1564 - val_accuracy: 0.9547
Epoch 23/50
82/82 [=====] - 4s 51ms/step - loss: 0.0058 - accu
racy: 0.9981 - val_loss: 0.1221 - val_accuracy: 0.9632
Epoch 24/50
82/82 [=====] - 4s 49ms/step - loss: 0.0037 - accu
racy: 0.9985 - val_loss: 0.1610 - val_accuracy: 0.9681
Epoch 25/50
82/82 [=====] - 4s 49ms/step - loss: 0.0018 - accu
racy: 0.9996 - val_loss: 0.1790 - val_accuracy: 0.9657
Epoch 26/50
82/82 [=====] - 4s 49ms/step - loss: 0.0018 - accu
racy: 0.9992 - val_loss: 1.1571 - val_accuracy: 0.8480
Epoch 27/50
82/82 [=====] - 4s 49ms/step - loss: 5.1606e-04 -
accuracy: 1.0000 - val_loss: 0.1598 - val_accuracy: 0.9669
Epoch 28/50
82/82 [=====] - 4s 49ms/step - loss: 2.3609e-04 -
accuracy: 1.0000 - val_loss: 0.1645 - val_accuracy: 0.9645
Epoch 29/50
82/82 [=====] - 4s 49ms/step - loss: 1.2894e-04 -
accuracy: 1.0000 - val_loss: 0.1669 - val_accuracy: 0.9694
Epoch 30/50

82/82 [=====] - 4s 51ms/step - loss: 8.5582e-05 -
accuracy: 1.0000 - val_loss: 0.1696 - val_accuracy: 0.9694
Epoch 31/50
82/82 [=====] - 4s 49ms/step - loss: 1.4383e-04 -
accuracy: 1.0000 - val_loss: 0.1709 - val_accuracy: 0.9681
Epoch 32/50
82/82 [=====] - 4s 49ms/step - loss: 3.8188e-05 -
accuracy: 1.0000 - val_loss: 0.1720 - val_accuracy: 0.9669
Epoch 33/50
82/82 [=====] - 4s 49ms/step - loss: 4.4959e-05 -
accuracy: 1.0000 - val_loss: 0.1748 - val_accuracy: 0.9706
Epoch 34/50
82/82 [=====] - 4s 49ms/step - loss: 9.3493e-05 -
accuracy: 1.0000 - val_loss: 0.1808 - val_accuracy: 0.9706
Epoch 35/50
82/82 [=====] - 4s 51ms/step - loss: 4.8533e-05 -
accuracy: 1.0000 - val_loss: 0.1849 - val_accuracy: 0.9706
Epoch 36/50
82/82 [=====] - 4s 49ms/step - loss: 4.1866e-05 -
accuracy: 1.0000 - val_loss: 0.1845 - val_accuracy: 0.9706
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 3.0501e-05 -
accuracy: 1.0000 - val_loss: 0.1857 - val_accuracy: 0.9706
Epoch 38/50
82/82 [=====] - 4s 49ms/step - loss: 2.2793e-05 -
accuracy: 1.0000 - val_loss: 0.1859 - val_accuracy: 0.9694
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 2.9546e-05 -
accuracy: 1.0000 - val_loss: 0.1882 - val_accuracy: 0.9694
Epoch 40/50
82/82 [=====] - 4s 51ms/step - loss: 2.2750e-05 -
accuracy: 1.0000 - val_loss: 0.1881 - val_accuracy: 0.9694
Epoch 41/50
82/82 [=====] - 4s 50ms/step - loss: 2.6390e-05 -
accuracy: 1.0000 - val_loss: 0.1904 - val_accuracy: 0.9681
Epoch 42/50
82/82 [=====] - 4s 49ms/step - loss: 2.4831e-05 -
accuracy: 1.0000 - val_loss: 0.1902 - val_accuracy: 0.9694
Epoch 43/50
82/82 [=====] - 4s 49ms/step - loss: 2.6252e-05 -
accuracy: 1.0000 - val_loss: 0.1919 - val_accuracy: 0.9706
Epoch 44/50
82/82 [=====] - 4s 49ms/step - loss: 1.6124e-05 -
accuracy: 1.0000 - val_loss: 0.1921 - val_accuracy: 0.9706
Epoch 45/50
82/82 [=====] - 4s 51ms/step - loss: 2.1360e-05 -
accuracy: 1.0000 - val_loss: 0.1929 - val_accuracy: 0.9694
Epoch 46/50
82/82 [=====] - 4s 49ms/step - loss: 2.6623e-05 -
accuracy: 1.0000 - val_loss: 0.1983 - val_accuracy: 0.9657
Epoch 47/50
82/82 [=====] - 4s 49ms/step - loss: 1.3578e-05 -
accuracy: 1.0000 - val_loss: 0.1986 - val_accuracy: 0.9694
Epoch 48/50
82/82 [=====] - 4s 49ms/step - loss: 1.6040e-05 -
accuracy: 1.0000 - val_loss: 0.2007 - val_accuracy: 0.9694

Epoch 49/50
82/82 [=====] - 4s 50ms/step - loss: 1.1972e-05 - accuracy: 1.0000 - val_loss: 0.2017 - val_accuracy: 0.9694
Epoch 50/50
82/82 [=====] - 4s 51ms/step - loss: 1.2384e-05 - accuracy: 1.0000 - val_loss: 0.2017 - val_accuracy: 0.9694
21/21 [=====] - 0s 11ms/step - loss: 0.3201 - accuracy: 0.9572
Epoch 1/50
82/82 [=====] - 5s 51ms/step - loss: 0.2403 - accuracy: 0.9041 - val_loss: 0.5562 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 50ms/step - loss: 0.1486 - accuracy: 0.9476 - val_loss: 0.3950 - val_accuracy: 0.7537
Epoch 3/50
82/82 [=====] - 4s 49ms/step - loss: 0.1180 - accuracy: 0.9580 - val_loss: 0.2208 - val_accuracy: 0.9436
Epoch 4/50
82/82 [=====] - 4s 51ms/step - loss: 0.1294 - accuracy: 0.9522 - val_loss: 0.2203 - val_accuracy: 0.9191
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.1002 - accuracy: 0.9618 - val_loss: 0.2396 - val_accuracy: 0.9056
Epoch 6/50
82/82 [=====] - 4s 50ms/step - loss: 0.1006 - accuracy: 0.9652 - val_loss: 0.3636 - val_accuracy: 0.8444
Epoch 7/50
82/82 [=====] - 4s 49ms/step - loss: 0.0856 - accuracy: 0.9733 - val_loss: 0.3309 - val_accuracy: 0.8480
Epoch 8/50
82/82 [=====] - 4s 49ms/step - loss: 0.0772 - accuracy: 0.9706 - val_loss: 1.5337 - val_accuracy: 0.7316
Epoch 9/50
82/82 [=====] - 4s 51ms/step - loss: 0.0793 - accuracy: 0.9694 - val_loss: 0.5650 - val_accuracy: 0.7757
Epoch 10/50
82/82 [=====] - 4s 49ms/step - loss: 0.0695 - accuracy: 0.9721 - val_loss: 0.5363 - val_accuracy: 0.8002
Epoch 11/50
82/82 [=====] - 4s 49ms/step - loss: 0.0590 - accuracy: 0.9790 - val_loss: 0.1406 - val_accuracy: 0.9583
Epoch 12/50
82/82 [=====] - 4s 49ms/step - loss: 0.0455 - accuracy: 0.9851 - val_loss: 0.3158 - val_accuracy: 0.8456
Epoch 13/50
82/82 [=====] - 4s 49ms/step - loss: 0.0546 - accuracy: 0.9817 - val_loss: 0.2532 - val_accuracy: 0.9167
Epoch 14/50
82/82 [=====] - 4s 51ms/step - loss: 0.0281 - accuracy: 0.9912 - val_loss: 0.1456 - val_accuracy: 0.9534
Epoch 15/50
82/82 [=====] - 4s 49ms/step - loss: 0.0406 - accuracy: 0.9851 - val_loss: 0.3709 - val_accuracy: 0.8750
Epoch 16/50
82/82 [=====] - 4s 50ms/step - loss: 0.0208 - accuracy: 0.9939 - val_loss: 0.1305 - val_accuracy: 0.9669

Epoch 17/50
82/82 [=====] - 4s 49ms/step - loss: 0.0264 - accuracy: 0.9904 - val_loss: 0.3984 - val_accuracy: 0.8529
Epoch 18/50
82/82 [=====] - 4s 49ms/step - loss: 0.0232 - accuracy: 0.9927 - val_loss: 0.1759 - val_accuracy: 0.9400
Epoch 19/50
82/82 [=====] - 4s 49ms/step - loss: 0.0250 - accuracy: 0.9920 - val_loss: 0.1530 - val_accuracy: 0.9669
Epoch 20/50
82/82 [=====] - 4s 51ms/step - loss: 0.0136 - accuracy: 0.9954 - val_loss: 1.0277 - val_accuracy: 0.8554
Epoch 21/50
82/82 [=====] - 4s 50ms/step - loss: 0.0069 - accuracy: 0.9981 - val_loss: 0.1646 - val_accuracy: 0.9608
Epoch 22/50
82/82 [=====] - 4s 52ms/step - loss: 0.0024 - accuracy: 0.9996 - val_loss: 0.1631 - val_accuracy: 0.9657
Epoch 23/50
82/82 [=====] - 4s 49ms/step - loss: 9.2684e-04 - accuracy: 1.0000 - val_loss: 0.1751 - val_accuracy: 0.9657
Epoch 24/50
82/82 [=====] - 4s 49ms/step - loss: 3.8780e-04 - accuracy: 1.0000 - val_loss: 0.1746 - val_accuracy: 0.9681
Epoch 25/50
82/82 [=====] - 4s 49ms/step - loss: 1.9712e-04 - accuracy: 1.0000 - val_loss: 0.1842 - val_accuracy: 0.9681
Epoch 26/50
82/82 [=====] - 4s 49ms/step - loss: 1.5223e-04 - accuracy: 1.0000 - val_loss: 0.1904 - val_accuracy: 0.9681
Epoch 27/50
82/82 [=====] - 4s 51ms/step - loss: 1.5231e-04 - accuracy: 1.0000 - val_loss: 0.1946 - val_accuracy: 0.9681
Epoch 28/50
82/82 [=====] - 4s 48ms/step - loss: 1.2464e-04 - accuracy: 1.0000 - val_loss: 0.2003 - val_accuracy: 0.9694
Epoch 29/50
82/82 [=====] - 4s 49ms/step - loss: 1.7286e-04 - accuracy: 1.0000 - val_loss: 0.2065 - val_accuracy: 0.9681
Epoch 30/50
82/82 [=====] - 4s 50ms/step - loss: 1.0183e-04 - accuracy: 1.0000 - val_loss: 0.2040 - val_accuracy: 0.9681
Epoch 31/50
82/82 [=====] - 4s 49ms/step - loss: 6.1352e-05 - accuracy: 1.0000 - val_loss: 0.2065 - val_accuracy: 0.9694
Epoch 32/50
82/82 [=====] - 4s 51ms/step - loss: 9.4271e-05 - accuracy: 1.0000 - val_loss: 0.2073 - val_accuracy: 0.9694
Epoch 33/50
82/82 [=====] - 4s 48ms/step - loss: 7.9686e-05 - accuracy: 1.0000 - val_loss: 0.2101 - val_accuracy: 0.9694
Epoch 34/50
82/82 [=====] - 4s 50ms/step - loss: 4.5793e-05 - accuracy: 1.0000 - val_loss: 0.2155 - val_accuracy: 0.9681
Epoch 35/50
82/82 [=====] - 4s 49ms/step - loss: 4.3010e-05 -

accuracy: 1.0000 - val_loss: 0.2163 - val_accuracy: 0.9706
Epoch 36/50
82/82 [=====] - 4s 49ms/step - loss: 7.4336e-05 -
accuracy: 1.0000 - val_loss: 0.2243 - val_accuracy: 0.9694
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 4.3973e-05 -
accuracy: 1.0000 - val_loss: 0.2190 - val_accuracy: 0.9681
Epoch 38/50
82/82 [=====] - 4s 51ms/step - loss: 3.7015e-05 -
accuracy: 1.0000 - val_loss: 0.2170 - val_accuracy: 0.9706
Epoch 39/50
82/82 [=====] - 4s 48ms/step - loss: 3.3385e-05 -
accuracy: 1.0000 - val_loss: 0.2200 - val_accuracy: 0.9694
Epoch 40/50
82/82 [=====] - 4s 49ms/step - loss: 2.6523e-05 -
accuracy: 1.0000 - val_loss: 0.2189 - val_accuracy: 0.9694
Epoch 41/50
82/82 [=====] - 4s 49ms/step - loss: 2.9780e-05 -
accuracy: 1.0000 - val_loss: 0.2192 - val_accuracy: 0.9694
Epoch 42/50
82/82 [=====] - 4s 49ms/step - loss: 2.9950e-05 -
accuracy: 1.0000 - val_loss: 0.2227 - val_accuracy: 0.9694
Epoch 43/50
82/82 [=====] - 4s 50ms/step - loss: 2.7687e-05 -
accuracy: 1.0000 - val_loss: 0.2249 - val_accuracy: 0.9681
Epoch 44/50
82/82 [=====] - 4s 50ms/step - loss: 2.4257e-05 -
accuracy: 1.0000 - val_loss: 0.2297 - val_accuracy: 0.9694
Epoch 45/50
82/82 [=====] - 4s 49ms/step - loss: 1.6731e-05 -
accuracy: 1.0000 - val_loss: 0.2272 - val_accuracy: 0.9681
Epoch 46/50
82/82 [=====] - 4s 50ms/step - loss: 1.6394e-05 -
accuracy: 1.0000 - val_loss: 0.2254 - val_accuracy: 0.9694
Epoch 47/50
82/82 [=====] - 4s 49ms/step - loss: 1.8855e-05 -
accuracy: 1.0000 - val_loss: 0.2283 - val_accuracy: 0.9706
Epoch 48/50
82/82 [=====] - 4s 49ms/step - loss: 5.5925e-05 -
accuracy: 1.0000 - val_loss: 0.2405 - val_accuracy: 0.9657
Epoch 49/50
82/82 [=====] - 4s 49ms/step - loss: 3.1006e-05 -
accuracy: 1.0000 - val_loss: 0.2426 - val_accuracy: 0.9645
Epoch 50/50
82/82 [=====] - 4s 51ms/step - loss: 2.8263e-05 -
accuracy: 1.0000 - val_loss: 0.2327 - val_accuracy: 0.9694
21/21 [=====] - 0s 11ms/step - loss: 0.1858 - accu
racy: 0.9648
Epoch 1/50
82/82 [=====] - 5s 56ms/step - loss: 0.2331 - accu
racy: 0.9091 - val_loss: 0.5093 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 49ms/step - loss: 0.1490 - accu
racy: 0.9423 - val_loss: 0.5508 - val_accuracy: 0.8909
Epoch 3/50
82/82 [=====] - 4s 49ms/step - loss: 0.1112 - accu

racy: 0.9648 - val_loss: 0.4934 - val_accuracy: 0.8640
Epoch 4/50
82/82 [=====] - 4s 51ms/step - loss: 0.1099 - accuracy: 0.9572 - val_loss: 0.1748 - val_accuracy: 0.9473
Epoch 5/50
82/82 [=====] - 4s 50ms/step - loss: 0.0761 - accuracy: 0.9702 - val_loss: 0.1228 - val_accuracy: 0.9596
Epoch 6/50
82/82 [=====] - 4s 49ms/step - loss: 0.0802 - accuracy: 0.9733 - val_loss: 2.6466 - val_accuracy: 0.7304
Epoch 7/50
82/82 [=====] - 4s 50ms/step - loss: 0.0690 - accuracy: 0.9767 - val_loss: 0.7224 - val_accuracy: 0.6140
Epoch 8/50
82/82 [=====] - 4s 49ms/step - loss: 0.0444 - accuracy: 0.9847 - val_loss: 0.2302 - val_accuracy: 0.8995
Epoch 9/50
82/82 [=====] - 4s 49ms/step - loss: 0.0514 - accuracy: 0.9813 - val_loss: 0.3920 - val_accuracy: 0.8811
Epoch 10/50
82/82 [=====] - 4s 50ms/step - loss: 0.0341 - accuracy: 0.9878 - val_loss: 0.1893 - val_accuracy: 0.9375
Epoch 11/50
82/82 [=====] - 4s 49ms/step - loss: 0.0283 - accuracy: 0.9908 - val_loss: 0.4499 - val_accuracy: 0.9020
Epoch 12/50
82/82 [=====] - 4s 50ms/step - loss: 0.0448 - accuracy: 0.9813 - val_loss: 0.1919 - val_accuracy: 0.9473
Epoch 13/50
82/82 [=====] - 4s 49ms/step - loss: 0.0174 - accuracy: 0.9958 - val_loss: 0.1246 - val_accuracy: 0.9632
Epoch 14/50
82/82 [=====] - 4s 50ms/step - loss: 0.0148 - accuracy: 0.9954 - val_loss: 1.1264 - val_accuracy: 0.8015
Epoch 15/50
82/82 [=====] - 4s 51ms/step - loss: 0.0040 - accuracy: 0.9996 - val_loss: 0.3856 - val_accuracy: 0.9179
Epoch 16/50
82/82 [=====] - 4s 50ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 0.1653 - val_accuracy: 0.9681
Epoch 17/50
82/82 [=====] - 4s 49ms/step - loss: 5.8004e-04 - accuracy: 1.0000 - val_loss: 0.1907 - val_accuracy: 0.9596
Epoch 18/50
82/82 [=====] - 4s 50ms/step - loss: 3.0797e-04 - accuracy: 1.0000 - val_loss: 0.1664 - val_accuracy: 0.9706
Epoch 19/50
82/82 [=====] - 4s 50ms/step - loss: 4.4267e-04 - accuracy: 1.0000 - val_loss: 0.1834 - val_accuracy: 0.9632
Epoch 20/50
82/82 [=====] - 4s 49ms/step - loss: 7.5300e-04 - accuracy: 1.0000 - val_loss: 0.1757 - val_accuracy: 0.9498
Epoch 21/50
82/82 [=====] - 4s 50ms/step - loss: 0.0856 - accuracy: 0.9690 - val_loss: 1.1302 - val_accuracy: 0.7549
Epoch 22/50

82/82 [=====] - 4s 51ms/step - loss: 0.0662 - accuracy: 0.9778 - val_loss: 1.9139 - val_accuracy: 0.4926
Epoch 23/50
82/82 [=====] - 4s 49ms/step - loss: 0.0551 - accuracy: 0.9801 - val_loss: 0.5768 - val_accuracy: 0.7292
Epoch 24/50
82/82 [=====] - 4s 49ms/step - loss: 0.0562 - accuracy: 0.9797 - val_loss: 0.1943 - val_accuracy: 0.9461
Epoch 25/50
82/82 [=====] - 4s 49ms/step - loss: 0.0278 - accuracy: 0.9901 - val_loss: 0.4397 - val_accuracy: 0.9130
Epoch 26/50
82/82 [=====] - 4s 49ms/step - loss: 0.0226 - accuracy: 0.9916 - val_loss: 1.2419 - val_accuracy: 0.8125
Epoch 27/50
82/82 [=====] - 4s 51ms/step - loss: 0.0128 - accuracy: 0.9943 - val_loss: 0.2820 - val_accuracy: 0.9461
Epoch 28/50
82/82 [=====] - 4s 49ms/step - loss: 0.0174 - accuracy: 0.9935 - val_loss: 0.5382 - val_accuracy: 0.8750
Epoch 29/50
82/82 [=====] - 4s 49ms/step - loss: 0.0211 - accuracy: 0.9912 - val_loss: 0.2273 - val_accuracy: 0.9375
Epoch 30/50
82/82 [=====] - 4s 50ms/step - loss: 0.0178 - accuracy: 0.9943 - val_loss: 0.2763 - val_accuracy: 0.9301
Epoch 31/50
82/82 [=====] - 4s 51ms/step - loss: 0.0107 - accuracy: 0.9966 - val_loss: 0.1768 - val_accuracy: 0.9559
Epoch 32/50
82/82 [=====] - 4s 50ms/step - loss: 0.0088 - accuracy: 0.9969 - val_loss: 0.4638 - val_accuracy: 0.9216
Epoch 33/50
82/82 [=====] - 4s 49ms/step - loss: 0.0055 - accuracy: 0.9973 - val_loss: 0.1808 - val_accuracy: 0.9559
Epoch 34/50
82/82 [=====] - 4s 49ms/step - loss: 0.0022 - accuracy: 0.9996 - val_loss: 0.2254 - val_accuracy: 0.9461
Epoch 35/50
82/82 [=====] - 4s 49ms/step - loss: 7.8102e-04 - accuracy: 1.0000 - val_loss: 0.2415 - val_accuracy: 0.9436
Epoch 36/50
82/82 [=====] - 4s 51ms/step - loss: 6.4559e-04 - accuracy: 0.9996 - val_loss: 0.1893 - val_accuracy: 0.9657
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 2.7182e-04 - accuracy: 1.0000 - val_loss: 0.1959 - val_accuracy: 0.9681
Epoch 38/50
82/82 [=====] - 4s 49ms/step - loss: 1.1282e-04 - accuracy: 1.0000 - val_loss: 0.1922 - val_accuracy: 0.9657
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 8.5764e-05 - accuracy: 1.0000 - val_loss: 0.1973 - val_accuracy: 0.9681
Epoch 40/50
82/82 [=====] - 4s 49ms/step - loss: 5.5017e-05 - accuracy: 1.0000 - val_loss: 0.1978 - val_accuracy: 0.9681

```

Epoch 41/50
82/82 [=====] - 4s 49ms/step - loss: 5.6858e-05 -
accuracy: 1.0000 - val_loss: 0.1997 - val_accuracy: 0.9669
Epoch 42/50
82/82 [=====] - 4s 51ms/step - loss: 5.7721e-05 -
accuracy: 1.0000 - val_loss: 0.1999 - val_accuracy: 0.9669
Epoch 43/50
82/82 [=====] - 4s 50ms/step - loss: 5.1162e-05 -
accuracy: 1.0000 - val_loss: 0.2035 - val_accuracy: 0.9669
Epoch 44/50
82/82 [=====] - 4s 49ms/step - loss: 6.4626e-05 -
accuracy: 1.0000 - val_loss: 0.2075 - val_accuracy: 0.9681
Epoch 45/50
82/82 [=====] - 4s 49ms/step - loss: 4.7109e-05 -
accuracy: 1.0000 - val_loss: 0.2098 - val_accuracy: 0.9669
Epoch 46/50
82/82 [=====] - 4s 52ms/step - loss: 2.7930e-05 -
accuracy: 1.0000 - val_loss: 0.2123 - val_accuracy: 0.9669
Epoch 47/50
82/82 [=====] - 4s 50ms/step - loss: 3.4467e-05 -
accuracy: 1.0000 - val_loss: 0.2125 - val_accuracy: 0.9681
Epoch 48/50
82/82 [=====] - 4s 49ms/step - loss: 2.1732e-05 -
accuracy: 1.0000 - val_loss: 0.2140 - val_accuracy: 0.9681
Epoch 49/50
82/82 [=====] - 4s 49ms/step - loss: 2.9864e-05 -
accuracy: 1.0000 - val_loss: 0.2169 - val_accuracy: 0.9681

```

```
In [72]: save_result('CNN #3', results_train[1], results_test[1], cvs[0], cvs[1], cvs[2], cvs[3], cvs[4])
```

Out[72]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	CNN #3	1.000	0.947	0.957	0.969	0.957	0.965	0.948	0.008	0.959
0	CNN #2	1.000	0.949	0.957	0.969	0.956	0.965	0.945	0.009	0.958
0	CNN #1	0.947	0.931	0.933	0.943	0.934	0.943	0.917	0.011	0.934
0	Initial Model	0.730	0.730	0.937	0.732	0.735	0.709	0.725	0.095	0.768

Prediction for Confusion Matrix

```
In [141]: predictions = model3.predict(x = test_images, steps = 10, verbose=0)
```

```
In [142]: pred_check = np.round(predictions)
```

```
In [143]: pred_check = pred_check[:]
pred_check = pred_check.flatten()
pred_check
```

Out[143]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)

```
In [144]: test_check = test_labels[:,0]
test_check
```

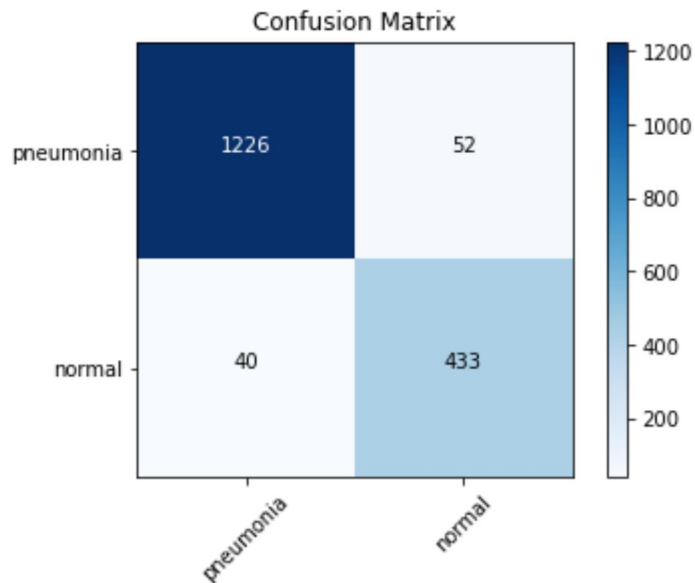
```
Out[144]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [145]: cm = confusion_matrix(y_true=test_check, y_pred=pred_check)
```

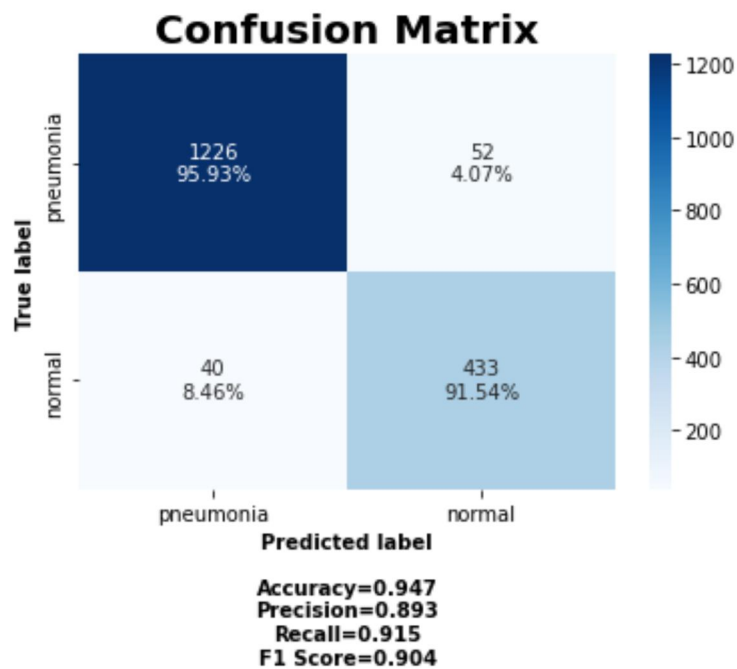
```
In [146]: cm_plot_labels = ['pneumonia', 'normal']
plot_confusion_matrix(cm=cm, classes=cm_plot_labels, title='Confusion Matrix')
```

Confusion matrix, without normalization

```
[[1226  52]
 [  40 433]]
```



```
In [147]: cm_plot_labels = ['pneumonia', 'normal']
make_confusion_matrix(cm, categories = cm_plot_labels, title='Confusion Matrix')
```



CNN Model 4

For this model, I will add weights

```
In [79]: norm_tot = len(os.listdir('re-split_data/train/normal'))
pneu_tot = len(os.listdir('re-split_data/train/pneumonia'))
norm_weight = norm_tot / (norm_tot + pneu_tot)
pneu_weight = pneu_tot / (norm_tot + pneu_tot)

print(f' normal weight is {round(norm_weight,2)}, pneumonia weight is {round(pneu_weight,2)}')
```

normal weight is 0.27, pneumonia weight is 0.73

```
In [80]: classWeight = {0 : norm_weight, 1 : pneu_weight}
```

```
In [81]: model 4 = models.Sequential()

model 4.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
model 4.add(layers.MaxPooling2D((2, 2)))

model 4.add(layers.Conv2D(32, (4, 4), activation='relu'))
model 4.add(BatchNormalization())
model 4.add(layers.MaxPooling2D((2, 2)))

model 4.add(layers.Conv2D(64, (3, 3), activation='relu'))
model 4.add(layers.MaxPooling2D((2, 2)))

model 4.add(layers.Conv2D(128, (3, 3), activation='relu'))
model 4.add(BatchNormalization())
model 4.add(layers.MaxPooling2D((2, 2)))

model 4.add(layers.Flatten())
model 4.add(layers.Dense(64, activation='relu'))
model 4.add(Dropout(0.1))
model 4.add(layers.Dense(1, activation='sigmoid'))

model 4.compile(loss='binary_crossentropy',
                 optimizer="adam",
                 metrics=['accuracy'])
```


Epoch 1/50
103/103 [=====] - 5s 47ms/step - loss: 0.0964 - accuracy: 0.9040 - val_loss: 0.5668 - val_accuracy: 0.7304
Epoch 2/50
103/103 [=====] - 5s 44ms/step - loss: 0.0759 - accuracy: 0.9236 - val_loss: 0.5885 - val_accuracy: 0.6789
Epoch 3/50
103/103 [=====] - 5s 45ms/step - loss: 0.0592 - accuracy: 0.9431 - val_loss: 0.2302 - val_accuracy: 0.9216
Epoch 4/50
103/103 [=====] - 5s 44ms/step - loss: 0.0599 - accuracy: 0.9419 - val_loss: 0.6683 - val_accuracy: 0.5699
Epoch 5/50
103/103 [=====] - 5s 44ms/step - loss: 0.0502 - accuracy: 0.9477 - val_loss: 0.5978 - val_accuracy: 0.7647
Epoch 6/50
103/103 [=====] - 5s 45ms/step - loss: 0.0486 - accuracy: 0.9566 - val_loss: 0.1500 - val_accuracy: 0.9436
Epoch 7/50
103/103 [=====] - 5s 44ms/step - loss: 0.0392 - accuracy: 0.9596 - val_loss: 0.1570 - val_accuracy: 0.9350
Epoch 8/50
103/103 [=====] - 5s 44ms/step - loss: 0.0283 - accuracy: 0.9713 - val_loss: 0.6939 - val_accuracy: 0.7672
Epoch 9/50
103/103 [=====] - 5s 44ms/step - loss: 0.0306 - accuracy: 0.9743 - val_loss: 0.4995 - val_accuracy: 0.7843
Epoch 10/50
103/103 [=====] - 5s 46ms/step - loss: 0.0224 - accuracy: 0.9798 - val_loss: 0.6731 - val_accuracy: 0.7010
Epoch 11/50
103/103 [=====] - 5s 44ms/step - loss: 0.0483 - accuracy: 0.9526 - val_loss: 2.8794 - val_accuracy: 0.7304
Epoch 12/50
103/103 [=====] - 5s 44ms/step - loss: 0.0289 - accuracy: 0.9716 - val_loss: 0.1260 - val_accuracy: 0.9620
Epoch 13/50
103/103 [=====] - 5s 44ms/step - loss: 0.0260 - accuracy: 0.9762 - val_loss: 0.1833 - val_accuracy: 0.9338
Epoch 14/50
103/103 [=====] - 5s 44ms/step - loss: 0.0204 - accuracy: 0.9817 - val_loss: 1.5700 - val_accuracy: 0.7843
Epoch 15/50
103/103 [=====] - 5s 46ms/step - loss: 0.0144 - accuracy: 0.9875 - val_loss: 0.2575 - val_accuracy: 0.9032
Epoch 16/50
103/103 [=====] - 5s 45ms/step - loss: 0.0109 - accuracy: 0.9878 - val_loss: 0.1689 - val_accuracy: 0.9645
Epoch 17/50
103/103 [=====] - 5s 44ms/step - loss: 0.0181 - accuracy: 0.9820 - val_loss: 0.4697 - val_accuracy: 0.9154
Epoch 18/50
103/103 [=====] - 5s 45ms/step - loss: 0.0084 - accuracy: 0.9930 - val_loss: 0.1632 - val_accuracy: 0.9522
Epoch 19/50
103/103 [=====] - 5s 44ms/step - loss: 0.0065 - ac

curacy: 0.9930 - val_loss: 0.1909 - val_accuracy: 0.9534
Epoch 20/50
103/103 [=====] - 5s 46ms/step - loss: 0.0072 - ac
curacy: 0.9930 - val_loss: 1.3219 - val_accuracy: 0.7941
Epoch 21/50
103/103 [=====] - 5s 44ms/step - loss: 0.0114 - ac
curacy: 0.9890 - val_loss: 0.3799 - val_accuracy: 0.9167
Epoch 22/50
103/103 [=====] - 5s 44ms/step - loss: 0.0081 - ac
curacy: 0.9905 - val_loss: 0.3125 - val_accuracy: 0.8983
Epoch 23/50
103/103 [=====] - 5s 44ms/step - loss: 0.0051 - ac
curacy: 0.9954 - val_loss: 0.5052 - val_accuracy: 0.9118
Epoch 24/50
103/103 [=====] - 5s 46ms/step - loss: 0.0031 - ac
curacy: 0.9963 - val_loss: 0.2252 - val_accuracy: 0.9547
Epoch 25/50
103/103 [=====] - 4s 44ms/step - loss: 8.9554e-04
- accuracy: 0.9997 - val_loss: 0.1945 - val_accuracy: 0.9645
Epoch 26/50
103/103 [=====] - 5s 44ms/step - loss: 2.0235e-04
- accuracy: 1.0000 - val_loss: 0.1989 - val_accuracy: 0.9596
Epoch 27/50
103/103 [=====] - 5s 44ms/step - loss: 2.2233e-04
- accuracy: 1.0000 - val_loss: 0.1998 - val_accuracy: 0.9608
Epoch 28/50
103/103 [=====] - 5s 44ms/step - loss: 2.6511e-04
- accuracy: 1.0000 - val_loss: 0.1991 - val_accuracy: 0.9522
Epoch 29/50
103/103 [=====] - 5s 45ms/step - loss: 2.5925e-04
- accuracy: 1.0000 - val_loss: 0.2690 - val_accuracy: 0.9534
Epoch 30/50
103/103 [=====] - 5s 45ms/step - loss: 0.0157 - ac
curacy: 0.9872 - val_loss: 1.1415 - val_accuracy: 0.8137
Epoch 31/50
103/103 [=====] - 5s 44ms/step - loss: 0.0142 - ac
curacy: 0.9869 - val_loss: 0.1444 - val_accuracy: 0.9559
Epoch 32/50
103/103 [=====] - 5s 45ms/step - loss: 0.0093 - ac
curacy: 0.9899 - val_loss: 0.2588 - val_accuracy: 0.9363
Epoch 33/50
103/103 [=====] - 5s 44ms/step - loss: 0.0041 - ac
curacy: 0.9960 - val_loss: 0.2428 - val_accuracy: 0.9473
Epoch 34/50
103/103 [=====] - 5s 46ms/step - loss: 0.0021 - ac
curacy: 0.9979 - val_loss: 0.2870 - val_accuracy: 0.9216
Epoch 35/50
103/103 [=====] - 5s 44ms/step - loss: 0.0045 - ac
curacy: 0.9957 - val_loss: 0.2662 - val_accuracy: 0.9240
Epoch 36/50
103/103 [=====] - 5s 44ms/step - loss: 0.0025 - ac
curacy: 0.9969 - val_loss: 0.2290 - val_accuracy: 0.9559
Epoch 37/50
103/103 [=====] - 5s 44ms/step - loss: 0.0031 - ac
curacy: 0.9969 - val_loss: 0.2775 - val_accuracy: 0.9350
Epoch 38/50

```

103/103 [=====] - 5s 44ms/step - loss: 0.0058 - ac
curacy: 0.9939 - val_loss: 0.1753 - val_accuracy: 0.9571
Epoch 39/50
103/103 [=====] - 5s 46ms/step - loss: 0.0019 - ac
curacy: 0.9988 - val_loss: 0.2912 - val_accuracy: 0.9522
Epoch 40/50
103/103 [=====] - 4s 44ms/step - loss: 7.2031e-04
- accuracy: 0.9997 - val_loss: 0.2204 - val_accuracy: 0.9657
Epoch 41/50
103/103 [=====] - 5s 44ms/step - loss: 3.0550e-04
- accuracy: 1.0000 - val_loss: 0.4322 - val_accuracy: 0.9449
Epoch 42/50
103/103 [=====] - 5s 44ms/step - loss: 6.2607e-05
- accuracy: 1.0000 - val_loss: 0.2477 - val_accuracy: 0.9583
Epoch 43/50
103/103 [=====] - 5s 46ms/step - loss: 4.3994e-05
- accuracy: 1.0000 - val_loss: 0.2637 - val_accuracy: 0.9608
Epoch 44/50
103/103 [=====] - 5s 44ms/step - loss: 3.6080e-05
- accuracy: 1.0000 - val_loss: 0.2589 - val_accuracy: 0.9596
Epoch 45/50
103/103 [=====] - 5s 44ms/step - loss: 3.0611e-05
- accuracy: 1.0000 - val_loss: 0.2599 - val_accuracy: 0.9583
Epoch 46/50
103/103 [=====] - 5s 44ms/step - loss: 2.2343e-05
- accuracy: 1.0000 - val_loss: 0.2646 - val_accuracy: 0.9608
Epoch 47/50
103/103 [=====] - 5s 44ms/step - loss: 2.4166e-05
- accuracy: 1.0000 - val_loss: 0.2671 - val_accuracy: 0.9620
Epoch 48/50
103/103 [=====] - 5s 44ms/step - loss: 1.9905e-05
- accuracy: 1.0000 - val_loss: 0.2761 - val_accuracy: 0.9608
Epoch 49/50
103/103 [=====] - 5s 46ms/step - loss: 1.3650e-05
- accuracy: 1.0000 - val_loss: 0.2730 - val_accuracy: 0.9583
Epoch 50/50
103/103 [=====] - 5s 44ms/step - loss: 1.3475e-05
- accuracy: 1.0000 - val_loss: 0.2778 - val_accuracy: 0.9608

```

```
In [83]: results_train = model4.evaluate(train_images, train_y)
```

```

103/103 [=====] - 1s 11ms/step - loss: 1.7107e-05
- accuracy: 1.0000

```

```
In [84]: results_test = model4.evaluate(test_images, test_y)
```

```

55/55 [=====] - 1s 11ms/step - loss: 0.3324 - accu
racy: 0.9555

```

```
In [85]: results_train
```

```
Out[85]: [1.7106991435866803e-05, 1.0]
```

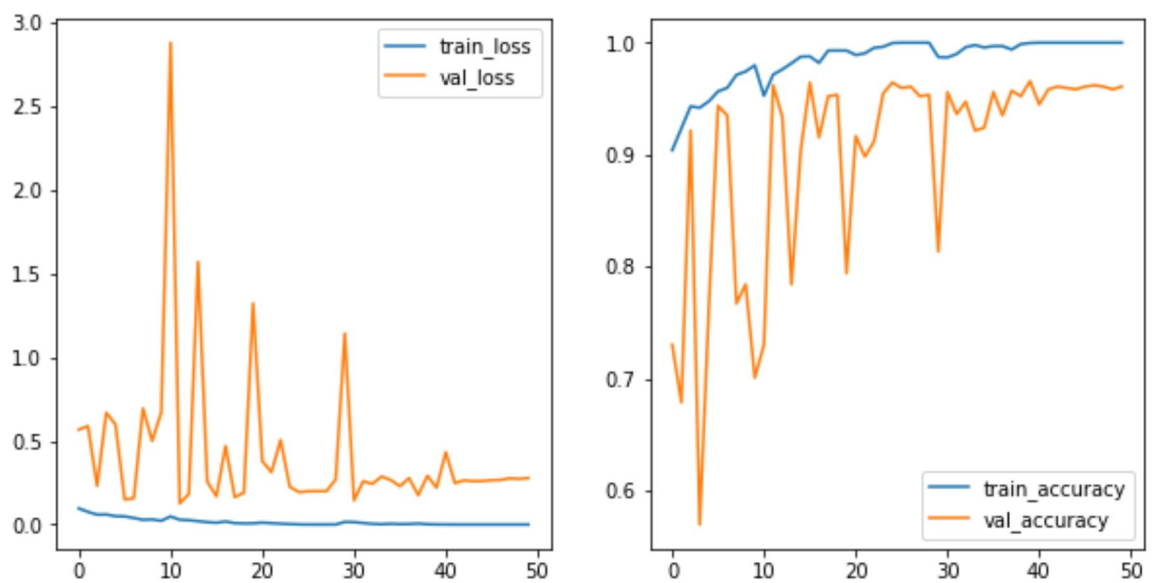
```
In [86]: results_test
```

```
Out[86]: [0.3323778510093689, 0.9554540514945984]
```

```
In [87]: train_loss = history4.history['loss']
train_acc = history4.history['accuracy']
val_loss = history4.history['val_loss']
val_acc = history4.history['val_accuracy']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=history4.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=history4.epoch, y=train_acc, ax=ax2, label='train_accuracy')
sns.lineplot(x=history4.epoch, y=val_loss, ax=ax1, label='val_loss')
sns.lineplot(x=history4.epoch, y=val_acc, ax=ax2, label='val_accuracy')
```

```
Out[87]: <AxesSubplot: >
```




```
In [90]: cvs = cross_val_score(keras_model5, train_images , train_y, cv=5)
```

Epoch 1/50
82/82 [=====] - 5s 48ms/step - loss: 0.2341 - accuracy: 0.9071 - val_loss: 0.6040 - val_accuracy: 0.7463
Epoch 2/50
82/82 [=====] - 4s 47ms/step - loss: 0.1642 - accuracy: 0.9381 - val_loss: 0.7065 - val_accuracy: 0.3946
Epoch 3/50
82/82 [=====] - 4s 46ms/step - loss: 0.1468 - accuracy: 0.9484 - val_loss: 0.5488 - val_accuracy: 0.7034
Epoch 4/50
82/82 [=====] - 4s 47ms/step - loss: 0.1152 - accuracy: 0.9564 - val_loss: 0.7290 - val_accuracy: 0.5527
Epoch 5/50
82/82 [=====] - 4s 47ms/step - loss: 0.0970 - accuracy: 0.9664 - val_loss: 1.4183 - val_accuracy: 0.7304
Epoch 6/50
82/82 [=====] - 4s 47ms/step - loss: 0.0884 - accuracy: 0.9694 - val_loss: 0.3700 - val_accuracy: 0.8370
Epoch 7/50
82/82 [=====] - 4s 47ms/step - loss: 0.0963 - accuracy: 0.9660 - val_loss: 0.2322 - val_accuracy: 0.8909
Epoch 8/50
82/82 [=====] - 4s 47ms/step - loss: 0.0960 - accuracy: 0.9614 - val_loss: 0.9839 - val_accuracy: 0.5172
Epoch 9/50
82/82 [=====] - 4s 47ms/step - loss: 0.0876 - accuracy: 0.9664 - val_loss: 0.2979 - val_accuracy: 0.8885
Epoch 10/50
82/82 [=====] - 4s 47ms/step - loss: 0.0745 - accuracy: 0.9740 - val_loss: 0.1822 - val_accuracy: 0.9326
Epoch 11/50
82/82 [=====] - 4s 46ms/step - loss: 0.0618 - accuracy: 0.9778 - val_loss: 0.2397 - val_accuracy: 0.9056
Epoch 12/50
82/82 [=====] - 4s 47ms/step - loss: 0.0566 - accuracy: 0.9782 - val_loss: 0.1313 - val_accuracy: 0.9485
Epoch 13/50
82/82 [=====] - 4s 47ms/step - loss: 0.0627 - accuracy: 0.9786 - val_loss: 0.1123 - val_accuracy: 0.9620
Epoch 14/50
82/82 [=====] - 4s 47ms/step - loss: 0.0449 - accuracy: 0.9836 - val_loss: 0.1676 - val_accuracy: 0.9363
Epoch 15/50
82/82 [=====] - 4s 47ms/step - loss: 0.0240 - accuracy: 0.9927 - val_loss: 0.1313 - val_accuracy: 0.9510
Epoch 16/50
82/82 [=====] - 4s 48ms/step - loss: 0.0204 - accuracy: 0.9939 - val_loss: 0.4465 - val_accuracy: 0.9007
Epoch 17/50
82/82 [=====] - 4s 48ms/step - loss: 0.0200 - accuracy: 0.9931 - val_loss: 0.1200 - val_accuracy: 0.9620
Epoch 18/50
82/82 [=====] - 4s 48ms/step - loss: 0.0188 - accuracy: 0.9946 - val_loss: 0.1777 - val_accuracy: 0.9436
Epoch 19/50
82/82 [=====] - 4s 47ms/step - loss: 0.0078 - accuracy:

racy: 0.9981 - val_loss: 0.9433 - val_accuracy: 0.8284
Epoch 20/50
82/82 [=====] - 4s 47ms/step - loss: 0.0163 - accuracy: 0.9946 - val_loss: 0.4568 - val_accuracy: 0.8689
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0523 - accuracy: 0.9801 - val_loss: 0.3632 - val_accuracy: 0.8787
Epoch 22/50
82/82 [=====] - 4s 49ms/step - loss: 0.0421 - accuracy: 0.9847 - val_loss: 0.1249 - val_accuracy: 0.9571
Epoch 23/50
82/82 [=====] - 4s 47ms/step - loss: 0.0355 - accuracy: 0.9889 - val_loss: 0.1068 - val_accuracy: 0.9669
Epoch 24/50
82/82 [=====] - 4s 47ms/step - loss: 0.0397 - accuracy: 0.9851 - val_loss: 0.2715 - val_accuracy: 0.9265
Epoch 25/50
82/82 [=====] - 4s 47ms/step - loss: 0.0139 - accuracy: 0.9962 - val_loss: 0.2379 - val_accuracy: 0.9436
Epoch 26/50
82/82 [=====] - 4s 47ms/step - loss: 0.0045 - accuracy: 0.9985 - val_loss: 0.1733 - val_accuracy: 0.9522
Epoch 27/50
82/82 [=====] - 4s 46ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 0.1787 - val_accuracy: 0.9571
Epoch 28/50
82/82 [=====] - 4s 51ms/step - loss: 4.9177e-04 - accuracy: 1.0000 - val_loss: 0.1607 - val_accuracy: 0.9620
Epoch 29/50
82/82 [=====] - 4s 47ms/step - loss: 3.3166e-04 - accuracy: 1.0000 - val_loss: 0.1589 - val_accuracy: 0.9681
Epoch 30/50
82/82 [=====] - 4s 47ms/step - loss: 3.1103e-04 - accuracy: 1.0000 - val_loss: 0.1841 - val_accuracy: 0.9596
Epoch 31/50
82/82 [=====] - 4s 47ms/step - loss: 1.7796e-04 - accuracy: 1.0000 - val_loss: 0.1669 - val_accuracy: 0.9657
Epoch 32/50
82/82 [=====] - 4s 47ms/step - loss: 1.5965e-04 - accuracy: 1.0000 - val_loss: 0.1665 - val_accuracy: 0.9657
Epoch 33/50
82/82 [=====] - 4s 47ms/step - loss: 2.0851e-04 - accuracy: 1.0000 - val_loss: 0.1709 - val_accuracy: 0.9657
Epoch 34/50
82/82 [=====] - 4s 49ms/step - loss: 1.1423e-04 - accuracy: 1.0000 - val_loss: 0.1680 - val_accuracy: 0.9718
Epoch 35/50
82/82 [=====] - 4s 46ms/step - loss: 8.3550e-05 - accuracy: 1.0000 - val_loss: 0.1679 - val_accuracy: 0.9669
Epoch 36/50
82/82 [=====] - 4s 48ms/step - loss: 8.7680e-05 - accuracy: 1.0000 - val_loss: 0.1732 - val_accuracy: 0.9694
Epoch 37/50
82/82 [=====] - 4s 48ms/step - loss: 6.9315e-05 - accuracy: 1.0000 - val_loss: 0.1783 - val_accuracy: 0.9645
Epoch 38/50

82/82 [=====] - 4s 47ms/step - loss: 8.2282e-05 - accuracy: 1.0000 - val_loss: 0.1838 - val_accuracy: 0.9632
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 4.4054e-05 - accuracy: 1.0000 - val_loss: 0.1820 - val_accuracy: 0.9645
Epoch 40/50
82/82 [=====] - 4s 47ms/step - loss: 7.1033e-05 - accuracy: 1.0000 - val_loss: 0.1798 - val_accuracy: 0.9706
Epoch 41/50
82/82 [=====] - 4s 47ms/step - loss: 5.4551e-05 - accuracy: 1.0000 - val_loss: 0.1833 - val_accuracy: 0.9669
Epoch 42/50
82/82 [=====] - 4s 47ms/step - loss: 4.9364e-05 - accuracy: 1.0000 - val_loss: 0.1887 - val_accuracy: 0.9718
Epoch 43/50
82/82 [=====] - 4s 48ms/step - loss: 5.9455e-05 - accuracy: 1.0000 - val_loss: 0.1859 - val_accuracy: 0.9669
Epoch 44/50
82/82 [=====] - 4s 47ms/step - loss: 4.3198e-05 - accuracy: 1.0000 - val_loss: 0.1858 - val_accuracy: 0.9657
Epoch 45/50
82/82 [=====] - 4s 47ms/step - loss: 5.6293e-05 - accuracy: 1.0000 - val_loss: 0.1870 - val_accuracy: 0.9694
Epoch 46/50
82/82 [=====] - 4s 49ms/step - loss: 3.0196e-05 - accuracy: 1.0000 - val_loss: 0.1939 - val_accuracy: 0.9669
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 2.6717e-05 - accuracy: 1.0000 - val_loss: 0.1945 - val_accuracy: 0.9669
Epoch 48/50
82/82 [=====] - 4s 47ms/step - loss: 7.6135e-05 - accuracy: 1.0000 - val_loss: 0.2296 - val_accuracy: 0.9583
Epoch 49/50
82/82 [=====] - 4s 47ms/step - loss: 3.3941e-05 - accuracy: 1.0000 - val_loss: 0.1854 - val_accuracy: 0.9669
Epoch 50/50
82/82 [=====] - 4s 47ms/step - loss: 2.8431e-05 - accuracy: 1.0000 - val_loss: 0.1855 - val_accuracy: 0.9694
21/21 [=====] - 0s 11ms/step - loss: 0.2622 - accuracy: 0.9634
Epoch 1/50
82/82 [=====] - 5s 49ms/step - loss: 0.2590 - accuracy: 0.8972 - val_loss: 0.5019 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 48ms/step - loss: 0.1481 - accuracy: 0.9450 - val_loss: 0.7069 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 47ms/step - loss: 0.1473 - accuracy: 0.9526 - val_loss: 0.4342 - val_accuracy: 0.7328
Epoch 4/50
82/82 [=====] - 4s 47ms/step - loss: 0.1173 - accuracy: 0.9603 - val_loss: 1.3140 - val_accuracy: 0.7304
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.1035 - accuracy: 0.9626 - val_loss: 0.3372 - val_accuracy: 0.8309
Epoch 6/50

82/82 [=====] - 4s 48ms/step - loss: 0.1159 - accuracy: 0.9549 - val_loss: 0.3545 - val_accuracy: 0.8150
Epoch 7/50
82/82 [=====] - 4s 49ms/step - loss: 0.0863 - accuracy: 0.9713 - val_loss: 0.4771 - val_accuracy: 0.8199
Epoch 8/50
82/82 [=====] - 4s 47ms/step - loss: 0.0772 - accuracy: 0.9698 - val_loss: 0.1954 - val_accuracy: 0.9265
Epoch 9/50
82/82 [=====] - 4s 47ms/step - loss: 0.0793 - accuracy: 0.9729 - val_loss: 0.7396 - val_accuracy: 0.7512
Epoch 10/50
82/82 [=====] - 4s 46ms/step - loss: 0.0583 - accuracy: 0.9755 - val_loss: 0.1610 - val_accuracy: 0.9473
Epoch 11/50
82/82 [=====] - 4s 46ms/step - loss: 0.0448 - accuracy: 0.9843 - val_loss: 0.3015 - val_accuracy: 0.8946
Epoch 12/50
82/82 [=====] - 4s 46ms/step - loss: 0.0443 - accuracy: 0.9836 - val_loss: 0.2387 - val_accuracy: 0.9314
Epoch 13/50
82/82 [=====] - 4s 49ms/step - loss: 0.0290 - accuracy: 0.9916 - val_loss: 0.1406 - val_accuracy: 0.9547
Epoch 14/50
82/82 [=====] - 4s 46ms/step - loss: 0.0167 - accuracy: 0.9947 - val_loss: 0.1341 - val_accuracy: 0.9571
Epoch 15/50
82/82 [=====] - 4s 46ms/step - loss: 0.0264 - accuracy: 0.9912 - val_loss: 0.3113 - val_accuracy: 0.8873
Epoch 16/50
82/82 [=====] - 4s 48ms/step - loss: 0.0415 - accuracy: 0.9817 - val_loss: 0.4082 - val_accuracy: 0.8922
Epoch 17/50
82/82 [=====] - 4s 47ms/step - loss: 0.0164 - accuracy: 0.9950 - val_loss: 0.1742 - val_accuracy: 0.9522
Epoch 18/50
82/82 [=====] - 4s 47ms/step - loss: 0.0165 - accuracy: 0.9947 - val_loss: 0.1675 - val_accuracy: 0.9583
Epoch 19/50
82/82 [=====] - 4s 49ms/step - loss: 0.0113 - accuracy: 0.9962 - val_loss: 0.3627 - val_accuracy: 0.9289
Epoch 20/50
82/82 [=====] - 4s 46ms/step - loss: 0.0072 - accuracy: 0.9977 - val_loss: 0.1633 - val_accuracy: 0.9669
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 0.1754 - val_accuracy: 0.9534
Epoch 22/50
82/82 [=====] - 4s 46ms/step - loss: 5.3254e-04 - accuracy: 1.0000 - val_loss: 0.1628 - val_accuracy: 0.9632
Epoch 23/50
82/82 [=====] - 4s 46ms/step - loss: 3.9038e-04 - accuracy: 1.0000 - val_loss: 0.1792 - val_accuracy: 0.9632
Epoch 24/50
82/82 [=====] - 4s 46ms/step - loss: 2.4103e-04 - accuracy: 1.0000 - val_loss: 0.1820 - val_accuracy: 0.9657

Epoch 25/50
82/82 [=====] - 4s 47ms/step - loss: 2.5613e-04 - accuracy: 1.0000 - val_loss: 0.1817 - val_accuracy: 0.9669
Epoch 26/50
82/82 [=====] - 4s 48ms/step - loss: 1.1790e-04 - accuracy: 1.0000 - val_loss: 0.1914 - val_accuracy: 0.9669
Epoch 27/50
82/82 [=====] - 4s 46ms/step - loss: 1.3198e-04 - accuracy: 1.0000 - val_loss: 0.1915 - val_accuracy: 0.9657
Epoch 28/50
82/82 [=====] - 4s 46ms/step - loss: 1.4211e-04 - accuracy: 1.0000 - val_loss: 0.1919 - val_accuracy: 0.9657
Epoch 29/50
82/82 [=====] - 4s 47ms/step - loss: 1.5028e-04 - accuracy: 1.0000 - val_loss: 0.1934 - val_accuracy: 0.9632
Epoch 30/50
82/82 [=====] - 4s 46ms/step - loss: 1.2372e-04 - accuracy: 1.0000 - val_loss: 0.1958 - val_accuracy: 0.9681
Epoch 31/50
82/82 [=====] - 4s 49ms/step - loss: 6.2954e-05 - accuracy: 1.0000 - val_loss: 0.1999 - val_accuracy: 0.9669
Epoch 32/50
82/82 [=====] - 4s 46ms/step - loss: 6.7968e-05 - accuracy: 1.0000 - val_loss: 0.2036 - val_accuracy: 0.9645
Epoch 33/50
82/82 [=====] - 4s 46ms/step - loss: 4.4049e-05 - accuracy: 1.0000 - val_loss: 0.2087 - val_accuracy: 0.9657
Epoch 34/50
82/82 [=====] - 4s 48ms/step - loss: 5.9683e-05 - accuracy: 1.0000 - val_loss: 0.2137 - val_accuracy: 0.9632
Epoch 35/50
82/82 [=====] - 4s 46ms/step - loss: 3.5159e-05 - accuracy: 1.0000 - val_loss: 0.2119 - val_accuracy: 0.9645
Epoch 36/50
82/82 [=====] - 4s 47ms/step - loss: 4.1936e-05 - accuracy: 1.0000 - val_loss: 0.2126 - val_accuracy: 0.9657
Epoch 37/50
82/82 [=====] - 4s 49ms/step - loss: 5.0924e-05 - accuracy: 1.0000 - val_loss: 0.2132 - val_accuracy: 0.9657
Epoch 38/50
82/82 [=====] - 4s 46ms/step - loss: 3.2954e-05 - accuracy: 1.0000 - val_loss: 0.2167 - val_accuracy: 0.9657
Epoch 39/50
82/82 [=====] - 4s 47ms/step - loss: 2.7707e-05 - accuracy: 1.0000 - val_loss: 0.2159 - val_accuracy: 0.9657
Epoch 40/50
82/82 [=====] - 4s 47ms/step - loss: 2.1593e-05 - accuracy: 1.0000 - val_loss: 0.2147 - val_accuracy: 0.9657
Epoch 41/50
82/82 [=====] - 4s 47ms/step - loss: 3.8639e-05 - accuracy: 1.0000 - val_loss: 0.2180 - val_accuracy: 0.9632
Epoch 42/50
82/82 [=====] - 4s 47ms/step - loss: 3.2854e-05 - accuracy: 1.0000 - val_loss: 0.2221 - val_accuracy: 0.9632
Epoch 43/50
82/82 [=====] - 4s 48ms/step - loss: 1.4035e-04 -

accuracy: 1.0000 - val_loss: 0.2597 - val_accuracy: 0.9620
Epoch 44/50
82/82 [=====] - 4s 46ms/step - loss: 4.7511e-05 -
accuracy: 1.0000 - val_loss: 0.2198 - val_accuracy: 0.9645
Epoch 45/50
82/82 [=====] - 4s 47ms/step - loss: 4.1257e-05 -
accuracy: 1.0000 - val_loss: 0.2298 - val_accuracy: 0.9596
Epoch 46/50
82/82 [=====] - 4s 47ms/step - loss: 1.8327e-05 -
accuracy: 1.0000 - val_loss: 0.2323 - val_accuracy: 0.9645
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 2.4114e-05 -
accuracy: 1.0000 - val_loss: 0.2277 - val_accuracy: 0.9645
Epoch 48/50
82/82 [=====] - 4s 48ms/step - loss: 1.3570e-05 -
accuracy: 1.0000 - val_loss: 0.2294 - val_accuracy: 0.9645
Epoch 49/50
82/82 [=====] - 4s 47ms/step - loss: 2.0962e-05 -
accuracy: 1.0000 - val_loss: 0.2301 - val_accuracy: 0.9645
Epoch 50/50
82/82 [=====] - 4s 47ms/step - loss: 1.7229e-05 -
accuracy: 1.0000 - val_loss: 0.2373 - val_accuracy: 0.9645
21/21 [=====] - 0s 11ms/step - loss: 0.1169 - accu
racy: 0.9725
Epoch 1/50
82/82 [=====] - 5s 50ms/step - loss: 0.2636 - accu
racy: 0.8964 - val_loss: 0.5761 - val_accuracy: 0.7353
Epoch 2/50
82/82 [=====] - 4s 47ms/step - loss: 0.1641 - accu
racy: 0.9358 - val_loss: 0.4752 - val_accuracy: 0.7929
Epoch 3/50
82/82 [=====] - 4s 48ms/step - loss: 0.1277 - accu
racy: 0.9503 - val_loss: 0.5702 - val_accuracy: 0.6850
Epoch 4/50
82/82 [=====] - 4s 50ms/step - loss: 0.1176 - accu
racy: 0.9568 - val_loss: 0.5353 - val_accuracy: 0.7157
Epoch 5/50
82/82 [=====] - 4s 47ms/step - loss: 0.1001 - accu
racy: 0.9648 - val_loss: 0.2479 - val_accuracy: 0.8995
Epoch 6/50
82/82 [=====] - 4s 47ms/step - loss: 0.1018 - accu
racy: 0.9614 - val_loss: 0.2643 - val_accuracy: 0.8983
Epoch 7/50
82/82 [=====] - 4s 47ms/step - loss: 0.0947 - accu
racy: 0.9645 - val_loss: 0.6928 - val_accuracy: 0.7733
Epoch 8/50
82/82 [=====] - 4s 47ms/step - loss: 0.0714 - accu
racy: 0.9733 - val_loss: 0.9555 - val_accuracy: 0.7316
Epoch 9/50
82/82 [=====] - 4s 49ms/step - loss: 0.0602 - accu
racy: 0.9790 - val_loss: 0.1483 - val_accuracy: 0.9620
Epoch 10/50
82/82 [=====] - 4s 46ms/step - loss: 0.0492 - accu
racy: 0.9801 - val_loss: 0.8519 - val_accuracy: 0.6005
Epoch 11/50
82/82 [=====] - 4s 48ms/step - loss: 0.0449 - accu

racy: 0.9843 - val_loss: 0.6938 - val_accuracy: 0.8064
Epoch 12/50
82/82 [=====] - 4s 48ms/step - loss: 0.0492 - accuracy: 0.9832 - val_loss: 0.3914 - val_accuracy: 0.8615
Epoch 13/50
82/82 [=====] - 4s 47ms/step - loss: 0.0580 - accuracy: 0.9801 - val_loss: 2.8120 - val_accuracy: 0.7304
Epoch 14/50
82/82 [=====] - 4s 48ms/step - loss: 0.0421 - accuracy: 0.9836 - val_loss: 0.5392 - val_accuracy: 0.8860
Epoch 15/50
82/82 [=====] - 4s 47ms/step - loss: 0.0288 - accuracy: 0.9916 - val_loss: 3.0724 - val_accuracy: 0.7353
Epoch 16/50
82/82 [=====] - 4s 49ms/step - loss: 0.0540 - accuracy: 0.9797 - val_loss: 0.5357 - val_accuracy: 0.8738
Epoch 17/50
82/82 [=====] - 4s 47ms/step - loss: 0.0422 - accuracy: 0.9820 - val_loss: 0.1411 - val_accuracy: 0.9547
Epoch 18/50
82/82 [=====] - 4s 47ms/step - loss: 0.0342 - accuracy: 0.9855 - val_loss: 0.1703 - val_accuracy: 0.9461
Epoch 19/50
82/82 [=====] - 4s 47ms/step - loss: 0.0183 - accuracy: 0.9943 - val_loss: 0.3138 - val_accuracy: 0.8750
Epoch 20/50
82/82 [=====] - 4s 47ms/step - loss: 0.0113 - accuracy: 0.9969 - val_loss: 0.8243 - val_accuracy: 0.8640
Epoch 21/50
82/82 [=====] - 4s 49ms/step - loss: 0.0058 - accuracy: 0.9985 - val_loss: 0.1885 - val_accuracy: 0.9534
Epoch 22/50
82/82 [=====] - 4s 47ms/step - loss: 0.0018 - accuracy: 0.9996 - val_loss: 0.1447 - val_accuracy: 0.9657
Epoch 23/50
82/82 [=====] - 4s 47ms/step - loss: 0.0031 - accuracy: 0.9989 - val_loss: 0.3667 - val_accuracy: 0.9301
Epoch 24/50
82/82 [=====] - 4s 47ms/step - loss: 0.0011 - accuracy: 1.0000 - val_loss: 0.1758 - val_accuracy: 0.9669
Epoch 25/50
82/82 [=====] - 4s 48ms/step - loss: 2.9694e-04 - accuracy: 1.0000 - val_loss: 0.1747 - val_accuracy: 0.9694
Epoch 26/50
82/82 [=====] - 4s 48ms/step - loss: 2.3631e-04 - accuracy: 1.0000 - val_loss: 0.1843 - val_accuracy: 0.9657
Epoch 27/50
82/82 [=====] - 4s 47ms/step - loss: 1.8481e-04 - accuracy: 1.0000 - val_loss: 0.1832 - val_accuracy: 0.9681
Epoch 28/50
82/82 [=====] - 4s 49ms/step - loss: 1.4284e-04 - accuracy: 1.0000 - val_loss: 0.1956 - val_accuracy: 0.9681
Epoch 29/50
82/82 [=====] - 4s 47ms/step - loss: 9.4289e-05 - accuracy: 1.0000 - val_loss: 0.2092 - val_accuracy: 0.9694
Epoch 30/50

82/82 [=====] - 4s 47ms/step - loss: 9.4302e-05 -
accuracy: 1.0000 - val_loss: 0.2100 - val_accuracy: 0.9669
Epoch 31/50
82/82 [=====] - 4s 47ms/step - loss: 9.1536e-05 -
accuracy: 1.0000 - val_loss: 0.2160 - val_accuracy: 0.9718
Epoch 32/50
82/82 [=====] - 4s 47ms/step - loss: 5.9545e-05 -
accuracy: 1.0000 - val_loss: 0.2113 - val_accuracy: 0.9730
Epoch 33/50
82/82 [=====] - 4s 47ms/step - loss: 8.9381e-05 -
accuracy: 1.0000 - val_loss: 0.2250 - val_accuracy: 0.9681
Epoch 34/50
82/82 [=====] - 4s 47ms/step - loss: 4.4019e-05 -
accuracy: 1.0000 - val_loss: 0.2164 - val_accuracy: 0.9718
Epoch 35/50
82/82 [=====] - 4s 47ms/step - loss: 1.9194e-04 -
accuracy: 1.0000 - val_loss: 0.2135 - val_accuracy: 0.9694
Epoch 36/50
82/82 [=====] - 4s 47ms/step - loss: 1.1054e-04 -
accuracy: 1.0000 - val_loss: 0.2219 - val_accuracy: 0.9657
Epoch 37/50
82/82 [=====] - 4s 47ms/step - loss: 4.4745e-05 -
accuracy: 1.0000 - val_loss: 0.2223 - val_accuracy: 0.9694
Epoch 38/50
82/82 [=====] - 4s 49ms/step - loss: 4.3262e-05 -
accuracy: 1.0000 - val_loss: 0.2354 - val_accuracy: 0.9645
Epoch 39/50
82/82 [=====] - 4s 46ms/step - loss: 3.1588e-05 -
accuracy: 1.0000 - val_loss: 0.2301 - val_accuracy: 0.9669
Epoch 40/50
82/82 [=====] - 4s 48ms/step - loss: 2.1277e-05 -
accuracy: 1.0000 - val_loss: 0.2283 - val_accuracy: 0.9730
Epoch 41/50
82/82 [=====] - 4s 46ms/step - loss: 2.5957e-05 -
accuracy: 1.0000 - val_loss: 0.2305 - val_accuracy: 0.9730
Epoch 42/50
82/82 [=====] - 4s 47ms/step - loss: 2.5125e-05 -
accuracy: 1.0000 - val_loss: 0.2419 - val_accuracy: 0.9657
Epoch 43/50
82/82 [=====] - 4s 49ms/step - loss: 4.5500e-05 -
accuracy: 1.0000 - val_loss: 0.2399 - val_accuracy: 0.9694
Epoch 44/50
82/82 [=====] - 4s 48ms/step - loss: 3.8561e-05 -
accuracy: 1.0000 - val_loss: 0.2388 - val_accuracy: 0.9706
Epoch 45/50
82/82 [=====] - 4s 47ms/step - loss: 2.1902e-05 -
accuracy: 1.0000 - val_loss: 0.2339 - val_accuracy: 0.9718
Epoch 46/50
82/82 [=====] - 4s 47ms/step - loss: 1.1269e-05 -
accuracy: 1.0000 - val_loss: 0.2378 - val_accuracy: 0.9706
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 1.9812e-05 -
accuracy: 1.0000 - val_loss: 0.2401 - val_accuracy: 0.9718
Epoch 48/50
82/82 [=====] - 4s 48ms/step - loss: 1.1047e-05 -
accuracy: 1.0000 - val_loss: 0.2421 - val_accuracy: 0.9730

Epoch 49/50
82/82 [=====] - 4s 49ms/step - loss: 1.2265e-05 - accuracy: 1.0000 - val_loss: 0.2438 - val_accuracy: 0.9706
Epoch 50/50
82/82 [=====] - 4s 47ms/step - loss: 1.3162e-05 - accuracy: 1.0000 - val_loss: 0.2474 - val_accuracy: 0.9718
21/21 [=====] - 0s 12ms/step - loss: 0.3326 - accuracy: 0.9526
Epoch 1/50
82/82 [=====] - 5s 50ms/step - loss: 0.2476 - accuracy: 0.8949 - val_loss: 0.5278 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 47ms/step - loss: 0.1618 - accuracy: 0.9419 - val_loss: 0.5940 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 46ms/step - loss: 0.1417 - accuracy: 0.9480 - val_loss: 0.4598 - val_accuracy: 0.7426
Epoch 4/50
82/82 [=====] - 4s 47ms/step - loss: 0.1172 - accuracy: 0.9561 - val_loss: 0.2053 - val_accuracy: 0.9363
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.1057 - accuracy: 0.9599 - val_loss: 0.7020 - val_accuracy: 0.7610
Epoch 6/50
82/82 [=====] - 4s 46ms/step - loss: 0.0909 - accuracy: 0.9664 - val_loss: 0.3411 - val_accuracy: 0.8578
Epoch 7/50
82/82 [=====] - 4s 47ms/step - loss: 0.0769 - accuracy: 0.9736 - val_loss: 0.3344 - val_accuracy: 0.8235
Epoch 8/50
82/82 [=====] - 4s 47ms/step - loss: 0.0956 - accuracy: 0.9641 - val_loss: 1.4264 - val_accuracy: 0.7304
Epoch 9/50
82/82 [=====] - 4s 47ms/step - loss: 0.0745 - accuracy: 0.9694 - val_loss: 0.4215 - val_accuracy: 0.8051
Epoch 10/50
82/82 [=====] - 4s 48ms/step - loss: 0.0442 - accuracy: 0.9859 - val_loss: 0.2034 - val_accuracy: 0.9191
Epoch 11/50
82/82 [=====] - 4s 47ms/step - loss: 0.0448 - accuracy: 0.9828 - val_loss: 0.7321 - val_accuracy: 0.8260
Epoch 12/50
82/82 [=====] - 4s 47ms/step - loss: 0.0504 - accuracy: 0.9847 - val_loss: 0.6035 - val_accuracy: 0.8186
Epoch 13/50
82/82 [=====] - 4s 47ms/step - loss: 0.0306 - accuracy: 0.9889 - val_loss: 0.3042 - val_accuracy: 0.9167
Epoch 14/50
82/82 [=====] - 4s 47ms/step - loss: 0.0280 - accuracy: 0.9893 - val_loss: 0.3033 - val_accuracy: 0.9216
Epoch 15/50
82/82 [=====] - 4s 46ms/step - loss: 0.0408 - accuracy: 0.9870 - val_loss: 1.9542 - val_accuracy: 0.7463
Epoch 16/50
82/82 [=====] - 4s 49ms/step - loss: 0.0331 - accuracy: 0.9866 - val_loss: 0.4088 - val_accuracy: 0.8284

Epoch 17/50
82/82 [=====] - 4s 47ms/step - loss: 0.0289 - accuracy: 0.9901 - val_loss: 0.2311 - val_accuracy: 0.9142
Epoch 18/50
82/82 [=====] - 4s 48ms/step - loss: 0.0260 - accuracy: 0.9912 - val_loss: 1.1814 - val_accuracy: 0.5588
Epoch 19/50
82/82 [=====] - 4s 47ms/step - loss: 0.0155 - accuracy: 0.9927 - val_loss: 0.0966 - val_accuracy: 0.9706
Epoch 20/50
82/82 [=====] - 4s 46ms/step - loss: 0.0091 - accuracy: 0.9969 - val_loss: 0.1644 - val_accuracy: 0.9485
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0080 - accuracy: 0.9973 - val_loss: 0.2915 - val_accuracy: 0.9387
Epoch 22/50
82/82 [=====] - 4s 49ms/step - loss: 0.0058 - accuracy: 0.9981 - val_loss: 0.1406 - val_accuracy: 0.9608
Epoch 23/50
82/82 [=====] - 4s 46ms/step - loss: 0.0086 - accuracy: 0.9977 - val_loss: 0.2827 - val_accuracy: 0.9449
Epoch 24/50
82/82 [=====] - 4s 47ms/step - loss: 0.0141 - accuracy: 0.9954 - val_loss: 0.2455 - val_accuracy: 0.9167
Epoch 25/50
82/82 [=====] - 4s 47ms/step - loss: 0.0213 - accuracy: 0.9947 - val_loss: 0.3000 - val_accuracy: 0.9265
Epoch 26/50
82/82 [=====] - 4s 48ms/step - loss: 0.0284 - accuracy: 0.9874 - val_loss: 0.1230 - val_accuracy: 0.9681
Epoch 27/50
82/82 [=====] - 4s 49ms/step - loss: 0.0036 - accuracy: 0.9992 - val_loss: 0.2785 - val_accuracy: 0.9424
Epoch 28/50
82/82 [=====] - 4s 47ms/step - loss: 0.0015 - accuracy: 1.0000 - val_loss: 0.2371 - val_accuracy: 0.9559
Epoch 29/50
82/82 [=====] - 4s 47ms/step - loss: 9.0321e-04 - accuracy: 1.0000 - val_loss: 0.2599 - val_accuracy: 0.9522
Epoch 30/50
82/82 [=====] - 4s 47ms/step - loss: 4.5722e-04 - accuracy: 1.0000 - val_loss: 0.1271 - val_accuracy: 0.9706
Epoch 31/50
82/82 [=====] - 4s 47ms/step - loss: 1.6490e-04 - accuracy: 1.0000 - val_loss: 0.1355 - val_accuracy: 0.9730
Epoch 32/50
82/82 [=====] - 4s 47ms/step - loss: 2.7070e-04 - accuracy: 1.0000 - val_loss: 0.1422 - val_accuracy: 0.9743
Epoch 33/50
82/82 [=====] - 4s 48ms/step - loss: 1.4790e-04 - accuracy: 1.0000 - val_loss: 0.1490 - val_accuracy: 0.9743
Epoch 34/50
82/82 [=====] - 4s 47ms/step - loss: 7.9734e-05 - accuracy: 1.0000 - val_loss: 0.1517 - val_accuracy: 0.9755
Epoch 35/50
82/82 [=====] - 4s 47ms/step - loss: 1.4092e-04 -

accuracy: 1.0000 - val_loss: 0.1653 - val_accuracy: 0.9743
Epoch 36/50
82/82 [=====] - 4s 47ms/step - loss: 6.5759e-05 -
accuracy: 1.0000 - val_loss: 0.1549 - val_accuracy: 0.9730
Epoch 37/50
82/82 [=====] - 4s 47ms/step - loss: 5.2638e-05 -
accuracy: 1.0000 - val_loss: 0.1584 - val_accuracy: 0.9706
Epoch 38/50
82/82 [=====] - 4s 47ms/step - loss: 4.8810e-05 -
accuracy: 1.0000 - val_loss: 0.1605 - val_accuracy: 0.9730
Epoch 39/50
82/82 [=====] - 4s 49ms/step - loss: 3.4591e-05 -
accuracy: 1.0000 - val_loss: 0.1619 - val_accuracy: 0.9743
Epoch 40/50
82/82 [=====] - 4s 48ms/step - loss: 3.8044e-05 -
accuracy: 1.0000 - val_loss: 0.1634 - val_accuracy: 0.9743
Epoch 41/50
82/82 [=====] - 4s 48ms/step - loss: 3.6914e-05 -
accuracy: 1.0000 - val_loss: 0.1657 - val_accuracy: 0.9730
Epoch 42/50
82/82 [=====] - 4s 47ms/step - loss: 4.0159e-05 -
accuracy: 1.0000 - val_loss: 0.1687 - val_accuracy: 0.9743
Epoch 43/50
82/82 [=====] - 4s 47ms/step - loss: 2.9145e-05 -
accuracy: 1.0000 - val_loss: 0.1732 - val_accuracy: 0.9743
Epoch 44/50
82/82 [=====] - 4s 49ms/step - loss: 3.8662e-05 -
accuracy: 1.0000 - val_loss: 0.1688 - val_accuracy: 0.9743
Epoch 45/50
82/82 [=====] - 4s 47ms/step - loss: 2.5317e-05 -
accuracy: 1.0000 - val_loss: 0.1712 - val_accuracy: 0.9743
Epoch 46/50
82/82 [=====] - 4s 47ms/step - loss: 7.9608e-05 -
accuracy: 1.0000 - val_loss: 0.1708 - val_accuracy: 0.9706
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 4.0859e-05 -
accuracy: 1.0000 - val_loss: 0.1716 - val_accuracy: 0.9706
Epoch 48/50
82/82 [=====] - 4s 47ms/step - loss: 2.2057e-05 -
accuracy: 1.0000 - val_loss: 0.1751 - val_accuracy: 0.9730
Epoch 49/50
82/82 [=====] - 4s 47ms/step - loss: 2.4112e-05 -
accuracy: 1.0000 - val_loss: 0.1778 - val_accuracy: 0.9730
Epoch 50/50
82/82 [=====] - 4s 49ms/step - loss: 2.7340e-05 -
accuracy: 1.0000 - val_loss: 0.1786 - val_accuracy: 0.9718
21/21 [=====] - 0s 11ms/step - loss: 0.1773 - accu
racy: 0.9648
Epoch 1/50
82/82 [=====] - 5s 50ms/step - loss: 0.2083 - accu
racy: 0.9159 - val_loss: 0.6035 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 47ms/step - loss: 0.1353 - accu
racy: 0.9507 - val_loss: 0.9052 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 47ms/step - loss: 0.1077 - accu

racy: 0.9637 - val_loss: 1.3493 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 47ms/step - loss: 0.1025 - accu
racy: 0.9641 - val_loss: 0.6380 - val_accuracy: 0.7341
Epoch 5/50
82/82 [=====] - 4s 49ms/step - loss: 0.0664 - accu
racy: 0.9759 - val_loss: 0.9661 - val_accuracy: 0.7304
Epoch 6/50
82/82 [=====] - 4s 46ms/step - loss: 0.0666 - accu
racy: 0.9740 - val_loss: 0.7366 - val_accuracy: 0.7623
Epoch 7/50
82/82 [=====] - 4s 48ms/step - loss: 0.0431 - accu
racy: 0.9855 - val_loss: 0.2146 - val_accuracy: 0.9203
Epoch 8/50
82/82 [=====] - 4s 46ms/step - loss: 0.0299 - accu
racy: 0.9904 - val_loss: 0.5882 - val_accuracy: 0.8505
Epoch 9/50
82/82 [=====] - 4s 47ms/step - loss: 0.0398 - accu
racy: 0.9878 - val_loss: 0.2009 - val_accuracy: 0.9314
Epoch 10/50
82/82 [=====] - 4s 47ms/step - loss: 0.0256 - accu
racy: 0.9912 - val_loss: 0.5297 - val_accuracy: 0.8211
Epoch 11/50
82/82 [=====] - 4s 49ms/step - loss: 0.0311 - accu
racy: 0.9897 - val_loss: 0.4960 - val_accuracy: 0.8836
Epoch 12/50
82/82 [=====] - 4s 46ms/step - loss: 0.0558 - accu
racy: 0.9813 - val_loss: 0.3241 - val_accuracy: 0.9203
Epoch 13/50
82/82 [=====] - 4s 47ms/step - loss: 0.0190 - accu
racy: 0.9924 - val_loss: 0.2003 - val_accuracy: 0.9449
Epoch 14/50
82/82 [=====] - 4s 47ms/step - loss: 0.0132 - accu
racy: 0.9966 - val_loss: 0.1481 - val_accuracy: 0.9596
Epoch 15/50
82/82 [=====] - 4s 47ms/step - loss: 0.0145 - accu
racy: 0.9958 - val_loss: 0.7162 - val_accuracy: 0.8787
Epoch 16/50
82/82 [=====] - 4s 48ms/step - loss: 0.0234 - accu
racy: 0.9920 - val_loss: 0.7619 - val_accuracy: 0.8676
Epoch 17/50
82/82 [=====] - 4s 47ms/step - loss: 0.0127 - accu
racy: 0.9943 - val_loss: 0.2038 - val_accuracy: 0.9436
Epoch 18/50
82/82 [=====] - 4s 47ms/step - loss: 0.0053 - accu
racy: 0.9977 - val_loss: 0.1670 - val_accuracy: 0.9583
Epoch 19/50
82/82 [=====] - 4s 47ms/step - loss: 0.0169 - accu
racy: 0.9943 - val_loss: 0.3434 - val_accuracy: 0.9093
Epoch 20/50
82/82 [=====] - 4s 48ms/step - loss: 0.0091 - accu
racy: 0.9969 - val_loss: 0.2567 - val_accuracy: 0.9424
Epoch 21/50
82/82 [=====] - 4s 47ms/step - loss: 0.0041 - accu
racy: 0.9992 - val_loss: 0.1748 - val_accuracy: 0.9559
Epoch 22/50

82/82 [=====] - 4s 49ms/step - loss: 0.0149 - accuracy: 0.9954 - val_loss: 4.7237 - val_accuracy: 0.7377
Epoch 23/50
82/82 [=====] - 4s 47ms/step - loss: 0.0128 - accuracy: 0.9962 - val_loss: 0.3768 - val_accuracy: 0.8836
Epoch 24/50
82/82 [=====] - 4s 47ms/step - loss: 0.0108 - accuracy: 0.9969 - val_loss: 0.3112 - val_accuracy: 0.9191
Epoch 25/50
82/82 [=====] - 4s 47ms/step - loss: 0.0108 - accuracy: 0.9969 - val_loss: 1.7111 - val_accuracy: 0.8076
Epoch 26/50
82/82 [=====] - 4s 47ms/step - loss: 0.0113 - accuracy: 0.9954 - val_loss: 0.8660 - val_accuracy: 0.8444
Epoch 27/50
82/82 [=====] - 4s 49ms/step - loss: 0.0136 - accuracy: 0.9954 - val_loss: 0.2355 - val_accuracy: 0.9632
Epoch 28/50
82/82 [=====] - 4s 47ms/step - loss: 0.0314 - accuracy: 0.9897 - val_loss: 4.2900 - val_accuracy: 0.7328
Epoch 29/50
82/82 [=====] - 4s 47ms/step - loss: 0.0185 - accuracy: 0.9947 - val_loss: 0.2155 - val_accuracy: 0.9485
Epoch 30/50
82/82 [=====] - 4s 47ms/step - loss: 0.0090 - accuracy: 0.9977 - val_loss: 0.1855 - val_accuracy: 0.9596
Epoch 31/50
82/82 [=====] - 4s 47ms/step - loss: 0.0127 - accuracy: 0.9958 - val_loss: 0.1872 - val_accuracy: 0.9559
Epoch 32/50
82/82 [=====] - 4s 47ms/step - loss: 0.0136 - accuracy: 0.9954 - val_loss: 1.6878 - val_accuracy: 0.7770
Epoch 33/50
82/82 [=====] - 4s 49ms/step - loss: 0.0183 - accuracy: 0.9947 - val_loss: 0.2303 - val_accuracy: 0.9632
Epoch 34/50
82/82 [=====] - 4s 47ms/step - loss: 0.0043 - accuracy: 0.9985 - val_loss: 0.2627 - val_accuracy: 0.9449
Epoch 35/50
82/82 [=====] - 4s 47ms/step - loss: 0.0012 - accuracy: 0.9996 - val_loss: 0.1974 - val_accuracy: 0.9571
Epoch 36/50
82/82 [=====] - 4s 47ms/step - loss: 0.0092 - accuracy: 0.9958 - val_loss: 0.2293 - val_accuracy: 0.9571
Epoch 37/50
82/82 [=====] - 4s 47ms/step - loss: 0.0026 - accuracy: 0.9992 - val_loss: 0.2199 - val_accuracy: 0.9534
Epoch 38/50
82/82 [=====] - 4s 49ms/step - loss: 6.7588e-04 - accuracy: 1.0000 - val_loss: 0.2270 - val_accuracy: 0.9583
Epoch 39/50
82/82 [=====] - 4s 47ms/step - loss: 2.3470e-04 - accuracy: 1.0000 - val_loss: 0.2239 - val_accuracy: 0.9620
Epoch 40/50
82/82 [=====] - 4s 47ms/step - loss: 1.0197e-04 - accuracy: 1.0000 - val_loss: 0.2267 - val_accuracy: 0.9620

```

Epoch 41/50
82/82 [=====] - 4s 47ms/step - loss: 7.4948e-05 -
accuracy: 1.0000 - val_loss: 0.2330 - val_accuracy: 0.9645
Epoch 42/50
82/82 [=====] - 4s 46ms/step - loss: 7.5998e-05 -
accuracy: 1.0000 - val_loss: 0.2317 - val_accuracy: 0.9632
Epoch 43/50
82/82 [=====] - 4s 47ms/step - loss: 7.3421e-05 -
accuracy: 1.0000 - val_loss: 0.2417 - val_accuracy: 0.9608
Epoch 44/50
82/82 [=====] - 4s 49ms/step - loss: 4.2832e-05 -
accuracy: 1.0000 - val_loss: 0.2383 - val_accuracy: 0.9632
Epoch 45/50
82/82 [=====] - 4s 47ms/step - loss: 3.1059e-05 -
accuracy: 1.0000 - val_loss: 0.2402 - val_accuracy: 0.9632
Epoch 46/50
82/82 [=====] - 4s 47ms/step - loss: 2.8110e-05 -
accuracy: 1.0000 - val_loss: 0.2396 - val_accuracy: 0.9632
Epoch 47/50
82/82 [=====] - 4s 47ms/step - loss: 3.1674e-05 -
accuracy: 1.0000 - val_loss: 0.2398 - val_accuracy: 0.9632
Epoch 48/50
82/82 [=====] - 4s 47ms/step - loss: 3.7738e-05 -
accuracy: 1.0000 - val_loss: 0.2404 - val_accuracy: 0.9632
Epoch 49/50
82/82 [=====] - 4s 49ms/step - loss: 3.2291e-05 -
accuracy: 1.0000 - val_loss: 0.2659 - val_accuracy: 0.9608

```

```
In [91]: save_result('CNN #4', results_train[1], results_test[1], cvs[0], cvs[1], cvs[2], cvs[3], cvs[4])
```

Out[91]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	CNN #4	1.000	0.955	0.963	0.972	0.953	0.965	0.950	0.009	0.961
0	CNN #3	1.000	0.947	0.957	0.969	0.957	0.965	0.948	0.008	0.959
0	CNN #2	1.000	0.949	0.957	0.969	0.956	0.965	0.945	0.009	0.958
0	CNN #1	0.947	0.931	0.933	0.943	0.934	0.943	0.917	0.011	0.934
0	Initial Model	0.730	0.730	0.937	0.732	0.735	0.709	0.725	0.095	0.768

```
In [149]: predictions = model_4.predict(x = test_images, steps = 10, verbose=0)
```

```
In [150]: pred_check = np.round(predictions)
```

```
In [151]: pred_check = pred_check[:]
pred_check = pred_check.flatten()
pred_check
```

Out[151]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)

```
In [152]: test_check = test_labels[:,0]
test_check
```

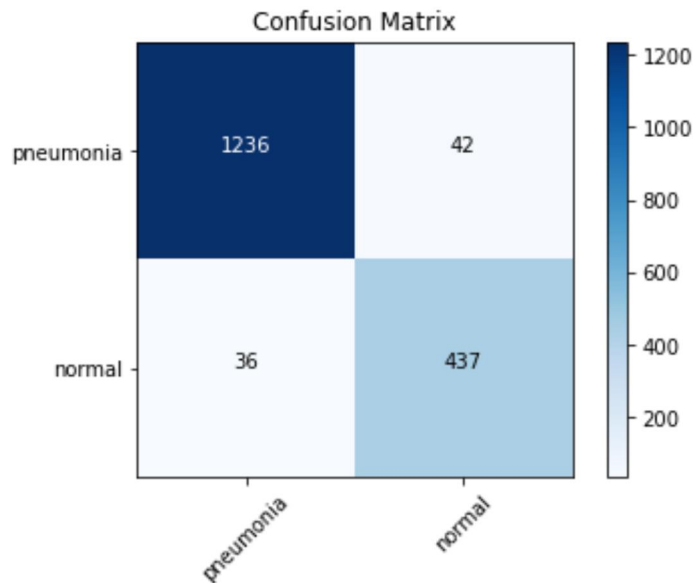
```
Out[152]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [153]: cm = confusion_matrix(y_true=test_check, y_pred=pred_check)
```

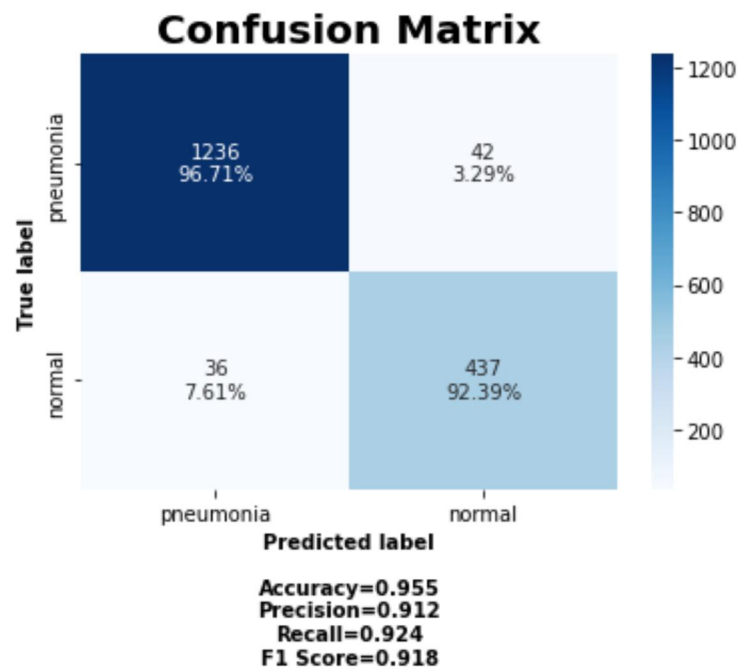
```
In [154]: cm_plot_labels = ['pneumonia', 'normal']
plot_confusion_matrix(cm=cm, classes=cm_plot_labels, title='Confusion Matrix')
```

Confusion matrix, without normalization

```
[[1236  42]
 [ 36 437]]
```



```
In [155]: cm_plot_labels = ['pneumonia', 'normal']  
make_confusion_matrix(cm, categories = cm_plot_labels, title='Confusion Matrix')
```



CNN Model 5: Model 3 with added layer

After observing how the first five models ran, CNN #2 was the best model due to low standard deviation for the cross validation and high test and train accuracy. I will try adding dropout to help the little bit of overtraining that is occurring

```

In [98]: model 6 = models.Sequential()

model 6.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
model 6.add(layers.MaxPooling2D((2, 2)))

model 6.add(layers.Conv2D(32, (4, 4), activation='relu'))
model 6.add(BatchNormalization())
model 6.add(layers.MaxPooling2D((2, 2)))

model 6.add(layers.Conv2D(64, (3, 3), activation='relu'))
model 6.add(BatchNormalization())
model 6.add(layers.MaxPooling2D((2, 2)))

model 6.add(layers.Conv2D(96, (3, 3), activation='relu', padding='same')) # new
model 6.add(BatchNormalization()) # new
model 6.add(Dropout(0.1)) # new
model 6.add(layers.MaxPooling2D((2, 2))) # new

model 6.add(layers.Conv2D(128, (3, 3), activation='relu', padding='same'))
model 6.add(BatchNormalization())
model 6.add(layers.MaxPooling2D((2, 2)))

model 6.add(layers.Flatten())
model 6.add(layers.Dense(64, activation='relu'))
model 6.add(Dropout(0.1))
model 6.add(layers.Dense(1, activation='sigmoid'))

model 6.compile(loss='binary_crossentropy',
                optimizer='adam',
                metrics=['accuracy'])

```

[illegible]

Epoch 1/50
103/103 [=====] - 6s 52ms/step - loss: 0.2311 - accuracy: 0.9089 - val_loss: 1.0388 - val_accuracy: 0.7304

Epoch 2/50
103/103 [=====] - 5s 48ms/step - loss: 0.1591 - accuracy: 0.9395 - val_loss: 1.2372 - val_accuracy: 0.7304

Epoch 3/50
103/103 [=====] - 5s 49ms/step - loss: 0.1301 - accuracy: 0.9514 - val_loss: 0.4789 - val_accuracy: 0.7623

Epoch 4/50
103/103 [=====] - 5s 49ms/step - loss: 0.1148 - accuracy: 0.9584 - val_loss: 1.4005 - val_accuracy: 0.7341

Epoch 5/50
103/103 [=====] - 5s 49ms/step - loss: 0.1077 - accuracy: 0.9612 - val_loss: 1.1663 - val_accuracy: 0.7304

Epoch 6/50
103/103 [=====] - 5s 51ms/step - loss: 0.0958 - accuracy: 0.9642 - val_loss: 1.0817 - val_accuracy: 0.4498

Epoch 7/50
103/103 [=====] - 5s 49ms/step - loss: 0.0840 - accuracy: 0.9673 - val_loss: 0.1275 - val_accuracy: 0.9534

Epoch 8/50
103/103 [=====] - 5s 49ms/step - loss: 0.0922 - accuracy: 0.9670 - val_loss: 0.2087 - val_accuracy: 0.9093

Epoch 9/50
103/103 [=====] - 5s 49ms/step - loss: 0.0757 - accuracy: 0.9728 - val_loss: 0.7503 - val_accuracy: 0.7892

Epoch 10/50
103/103 [=====] - 5s 50ms/step - loss: 0.0907 - accuracy: 0.9682 - val_loss: 0.2925 - val_accuracy: 0.8676

Epoch 11/50
103/103 [=====] - 5s 49ms/step - loss: 0.0650 - accuracy: 0.9752 - val_loss: 0.1731 - val_accuracy: 0.9534

Epoch 12/50
103/103 [=====] - 5s 49ms/step - loss: 0.0484 - accuracy: 0.9829 - val_loss: 0.1940 - val_accuracy: 0.9350

Epoch 13/50
103/103 [=====] - 5s 50ms/step - loss: 0.0494 - accuracy: 0.9810 - val_loss: 0.1163 - val_accuracy: 0.9645

Epoch 14/50
103/103 [=====] - 5s 50ms/step - loss: 0.0381 - accuracy: 0.9856 - val_loss: 0.1317 - val_accuracy: 0.9547

Epoch 15/50
103/103 [=====] - 5s 49ms/step - loss: 0.0417 - accuracy: 0.9844 - val_loss: 0.1432 - val_accuracy: 0.9522

Epoch 16/50
103/103 [=====] - 5s 49ms/step - loss: 0.0341 - accuracy: 0.9872 - val_loss: 0.2145 - val_accuracy: 0.9228

Epoch 17/50
103/103 [=====] - 5s 49ms/step - loss: 0.0398 - accuracy: 0.9872 - val_loss: 0.2305 - val_accuracy: 0.9167

Epoch 18/50
103/103 [=====] - 5s 49ms/step - loss: 0.0232 - accuracy: 0.9908 - val_loss: 0.2723 - val_accuracy: 0.9032

Epoch 19/50
103/103 [=====] - 5s 49ms/step - loss: 0.0189 - accuracy: 0.9908 - val_loss: 0.2723 - val_accuracy: 0.9032

curacy: 0.9942 - val_loss: 0.1256 - val_accuracy: 0.9632
Epoch 20/50
103/103 [=====] - 5s 49ms/step - loss: 0.0196 - ac
curacy: 0.9927 - val_loss: 0.1154 - val_accuracy: 0.9694
Epoch 21/50
103/103 [=====] - 5s 49ms/step - loss: 0.0276 - ac
curacy: 0.9899 - val_loss: 0.2875 - val_accuracy: 0.9338
Epoch 22/50
103/103 [=====] - 5s 49ms/step - loss: 0.0147 - ac
curacy: 0.9957 - val_loss: 0.1644 - val_accuracy: 0.9449
Epoch 23/50
103/103 [=====] - 5s 50ms/step - loss: 0.0299 - ac
curacy: 0.9890 - val_loss: 0.2606 - val_accuracy: 0.9118
Epoch 24/50
103/103 [=====] - 5s 49ms/step - loss: 0.0149 - ac
curacy: 0.9930 - val_loss: 0.3881 - val_accuracy: 0.9044
Epoch 25/50
103/103 [=====] - 5s 49ms/step - loss: 0.0187 - ac
curacy: 0.9939 - val_loss: 2.1235 - val_accuracy: 0.7537
Epoch 26/50
103/103 [=====] - 5s 49ms/step - loss: 0.0147 - ac
curacy: 0.9945 - val_loss: 0.1644 - val_accuracy: 0.9632
Epoch 27/50
103/103 [=====] - 5s 50ms/step - loss: 0.0138 - ac
curacy: 0.9951 - val_loss: 0.2430 - val_accuracy: 0.9547
Epoch 28/50
103/103 [=====] - 5s 49ms/step - loss: 0.0202 - ac
curacy: 0.9924 - val_loss: 0.3019 - val_accuracy: 0.9228
Epoch 29/50
103/103 [=====] - 5s 49ms/step - loss: 0.0264 - ac
curacy: 0.9914 - val_loss: 0.1855 - val_accuracy: 0.9461
Epoch 30/50
103/103 [=====] - 5s 49ms/step - loss: 0.0134 - ac
curacy: 0.9951 - val_loss: 0.1541 - val_accuracy: 0.9608
Epoch 31/50
103/103 [=====] - 5s 49ms/step - loss: 0.0029 - ac
curacy: 0.9991 - val_loss: 0.1589 - val_accuracy: 0.9681
Epoch 32/50
103/103 [=====] - 5s 49ms/step - loss: 0.0035 - ac
curacy: 0.9994 - val_loss: 0.1956 - val_accuracy: 0.9583
Epoch 33/50
103/103 [=====] - 5s 49ms/step - loss: 4.3671e-04
- accuracy: 1.0000 - val_loss: 0.1781 - val_accuracy: 0.9669
Epoch 34/50
103/103 [=====] - 5s 50ms/step - loss: 3.2410e-04
- accuracy: 1.0000 - val_loss: 0.1789 - val_accuracy: 0.9718
Epoch 35/50
103/103 [=====] - 5s 49ms/step - loss: 2.5489e-04
- accuracy: 1.0000 - val_loss: 0.1872 - val_accuracy: 0.9657
Epoch 36/50
103/103 [=====] - 5s 49ms/step - loss: 1.5903e-04
- accuracy: 1.0000 - val_loss: 0.1933 - val_accuracy: 0.9645
Epoch 37/50
103/103 [=====] - 5s 49ms/step - loss: 1.6927e-04
- accuracy: 1.0000 - val_loss: 0.2029 - val_accuracy: 0.9657
Epoch 38/50

```

103/103 [=====] - 5s 50ms/step - loss: 9.2095e-05
- accuracy: 1.0000 - val_loss: 0.2032 - val_accuracy: 0.9669
Epoch 39/50
103/103 [=====] - 5s 49ms/step - loss: 1.3282e-04
- accuracy: 1.0000 - val_loss: 0.2280 - val_accuracy: 0.9730
Epoch 40/50
103/103 [=====] - 5s 49ms/step - loss: 9.0506e-05
- accuracy: 1.0000 - val_loss: 0.2118 - val_accuracy: 0.9681
Epoch 41/50
103/103 [=====] - 5s 49ms/step - loss: 6.2496e-05
- accuracy: 1.0000 - val_loss: 0.2157 - val_accuracy: 0.9669
Epoch 42/50
103/103 [=====] - 5s 49ms/step - loss: 5.1160e-05
- accuracy: 1.0000 - val_loss: 0.2144 - val_accuracy: 0.9669
Epoch 43/50
103/103 [=====] - 5s 49ms/step - loss: 0.0298 - ac
curacy: 0.9921 - val_loss: 0.2718 - val_accuracy: 0.9032
Epoch 44/50
103/103 [=====] - 5s 50ms/step - loss: 0.0413 - ac
curacy: 0.9841 - val_loss: 1.5274 - val_accuracy: 0.7855
Epoch 45/50
103/103 [=====] - 5s 49ms/step - loss: 0.0243 - ac
curacy: 0.9921 - val_loss: 0.3327 - val_accuracy: 0.9363
Epoch 46/50
103/103 [=====] - 5s 49ms/step - loss: 0.0285 - ac
curacy: 0.9908 - val_loss: 0.7221 - val_accuracy: 0.7892
Epoch 47/50
103/103 [=====] - 5s 49ms/step - loss: 0.0225 - ac
curacy: 0.9927 - val_loss: 2.9416 - val_accuracy: 0.7304
Epoch 48/50
103/103 [=====] - 5s 50ms/step - loss: 0.0163 - ac
curacy: 0.9930 - val_loss: 0.1716 - val_accuracy: 0.9522
Epoch 49/50
103/103 [=====] - 5s 51ms/step - loss: 0.0034 - ac
curacy: 0.9988 - val_loss: 0.2305 - val_accuracy: 0.9596
Epoch 50/50
103/103 [=====] - 5s 49ms/step - loss: 0.0064 - ac
curacy: 0.9976 - val_loss: 0.3521 - val_accuracy: 0.9032

```

```
In [100]: results_train = model6.evaluate(train_images, train_y)
```

```

103/103 [=====] - 1s 12ms/step - loss: 0.1507 - ac
curacy: 0.9389

```

```
In [101]: results_test = model6.evaluate(test_images, test_y)
```

```

55/55 [=====] - 1s 12ms/step - loss: 0.4150 - accu
racy: 0.8932

```

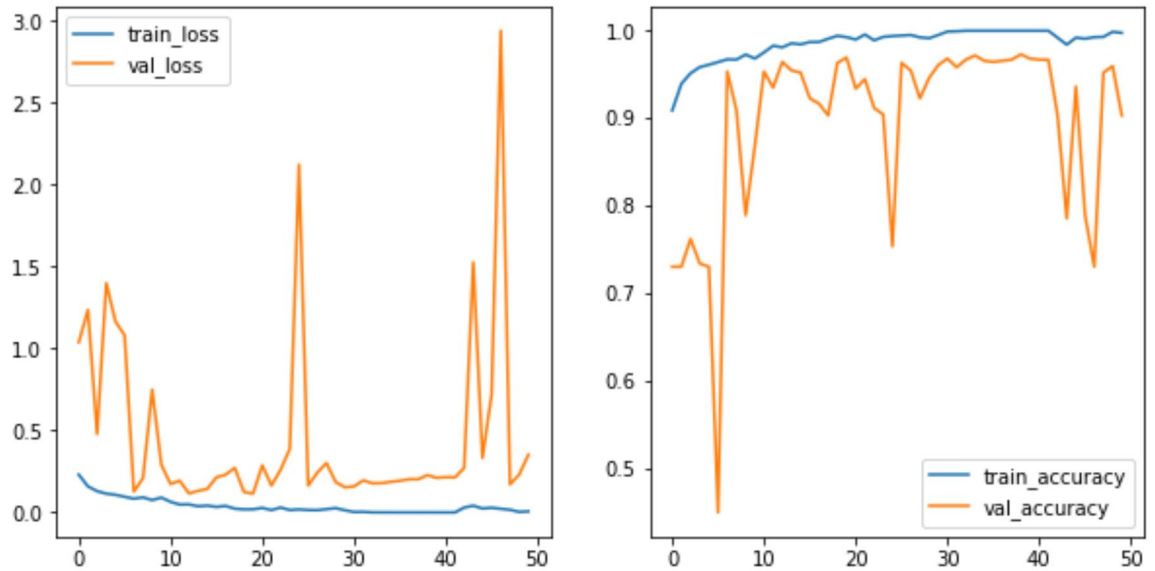
```

In [102]: train_loss = history.history['loss']
train_acc = history.history['accuracy']
val_loss = history.history['val_loss']
val_acc = history.history['val_accuracy']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
sns.lineplot(x=history.epoch, y=train_loss, ax=ax1, label='train_loss')
sns.lineplot(x=history.epoch, y=train_acc, ax=ax2, label='train_accuracy')
sns.lineplot(x=history.epoch, y=val_loss, ax=ax1, label='val_loss')
sns.lineplot(x=history.epoch, y=val_acc, ax=ax2, label='val_accuracy')

```

Out[102]: <AxesSubplot: >



```
In [103]: def build_cnn6():  
    model_6 = models.Sequential()  
  
    model_6.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(64,  
    , 64, 3)))  
    model_6.add(layers.MaxPooling2D((2, 2)))  
  
    model_6.add(layers.Conv2D(32, (4, 4), activation='relu'))  
    model_6.add(BatchNormalization())  
    model_6.add(layers.MaxPooling2D((2, 2)))  
  
    model_6.add(layers.Conv2D(64, (3, 3), activation='relu'))  
    model_6.add(BatchNormalization())  
    model_6.add(layers.MaxPooling2D((2, 2)))  
  
    model_6.add(layers.Conv2D(96, (3, 3), activation='relu', padding='same'  
    )) # new  
    model_6.add(BatchNormalization())  
    # new  
    model_6.add(Dropout(0.1))  
    # new  
    model_6.add(layers.MaxPooling2D((2, 2)))  
    # new  
  
    model_6.add(layers.Conv2D(128, (3, 3), activation='relu', padding='same'  
    ))  
    model_6.add(BatchNormalization())  
    model_6.add(layers.MaxPooling2D((2, 2)))  
  
    model_6.add(layers.Flatten())  
    model_6.add(layers.Dense(64, activation='relu'))  
    model_6.add(Dropout(0.1))  
    model_6.add(layers.Dense(1, activation='sigmoid'))  
  
    model_6.compile(loss='binary_crossentropy',  
                    optimizer="adam",  
                    metrics=['accuracy'])  
  
    return model_6
```

[illegible]

```
In [105]: cvs = cross_val_score(keras_model6, train_images , train_y, cv=5)
```

Epoch 1/50
82/82 [=====] - 6s 56ms/step - loss: 0.2369 - accuracy: 0.9083 - val_loss: 0.5335 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 53ms/step - loss: 0.1652 - accuracy: 0.9411 - val_loss: 1.1278 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 53ms/step - loss: 0.1388 - accuracy: 0.9507 - val_loss: 0.6078 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 53ms/step - loss: 0.1183 - accuracy: 0.9560 - val_loss: 0.3297 - val_accuracy: 0.8603
Epoch 5/50
82/82 [=====] - 4s 53ms/step - loss: 0.1051 - accuracy: 0.9595 - val_loss: 0.4812 - val_accuracy: 0.7782
Epoch 6/50
82/82 [=====] - 4s 53ms/step - loss: 0.0873 - accuracy: 0.9675 - val_loss: 0.5795 - val_accuracy: 0.8039
Epoch 7/50
82/82 [=====] - 4s 53ms/step - loss: 0.0739 - accuracy: 0.9721 - val_loss: 0.1368 - val_accuracy: 0.9498
Epoch 8/50
82/82 [=====] - 4s 53ms/step - loss: 0.0745 - accuracy: 0.9729 - val_loss: 0.3322 - val_accuracy: 0.8799
Epoch 9/50
82/82 [=====] - 4s 53ms/step - loss: 0.0519 - accuracy: 0.9832 - val_loss: 0.1514 - val_accuracy: 0.9461
Epoch 10/50
82/82 [=====] - 4s 53ms/step - loss: 0.0401 - accuracy: 0.9855 - val_loss: 0.3193 - val_accuracy: 0.9179
Epoch 11/50
82/82 [=====] - 4s 53ms/step - loss: 0.0377 - accuracy: 0.9855 - val_loss: 0.2175 - val_accuracy: 0.9338
Epoch 12/50
82/82 [=====] - 4s 53ms/step - loss: 0.0365 - accuracy: 0.9878 - val_loss: 0.3863 - val_accuracy: 0.8150
Epoch 13/50
82/82 [=====] - 4s 53ms/step - loss: 0.0379 - accuracy: 0.9862 - val_loss: 0.5202 - val_accuracy: 0.7586
Epoch 14/50
82/82 [=====] - 4s 53ms/step - loss: 0.0558 - accuracy: 0.9797 - val_loss: 2.0328 - val_accuracy: 0.7341
Epoch 15/50
82/82 [=====] - 4s 53ms/step - loss: 0.0424 - accuracy: 0.9855 - val_loss: 0.1710 - val_accuracy: 0.9485
Epoch 16/50
82/82 [=====] - 4s 53ms/step - loss: 0.0355 - accuracy: 0.9866 - val_loss: 2.5578 - val_accuracy: 0.7316
Epoch 17/50
82/82 [=====] - 4s 53ms/step - loss: 0.0285 - accuracy: 0.9904 - val_loss: 0.5187 - val_accuracy: 0.8860
Epoch 18/50
82/82 [=====] - 4s 53ms/step - loss: 0.0170 - accuracy: 0.9946 - val_loss: 0.4369 - val_accuracy: 0.8444
Epoch 19/50
82/82 [=====] - 4s 53ms/step - loss: 0.0272 - accuracy:

racy: 0.9878 - val_loss: 1.2385 - val_accuracy: 0.7880
Epoch 20/50
82/82 [=====] - 4s 53ms/step - loss: 0.0252 - accu
racy: 0.9916 - val_loss: 0.1709 - val_accuracy: 0.9424
Epoch 21/50
82/82 [=====] - 4s 53ms/step - loss: 0.0200 - accu
racy: 0.9943 - val_loss: 0.3416 - val_accuracy: 0.8860
Epoch 22/50
82/82 [=====] - 4s 53ms/step - loss: 0.0178 - accu
racy: 0.9943 - val_loss: 0.1735 - val_accuracy: 0.9522
Epoch 23/50
82/82 [=====] - 4s 52ms/step - loss: 0.0086 - accu
racy: 0.9977 - val_loss: 0.1519 - val_accuracy: 0.9571
Epoch 24/50
82/82 [=====] - 4s 53ms/step - loss: 0.0132 - accu
racy: 0.9958 - val_loss: 0.2170 - val_accuracy: 0.9203
Epoch 25/50
82/82 [=====] - 5s 56ms/step - loss: 0.0301 - accu
racy: 0.9897 - val_loss: 1.8573 - val_accuracy: 0.7512
Epoch 26/50
82/82 [=====] - 4s 53ms/step - loss: 0.0304 - accu
racy: 0.9885 - val_loss: 0.2455 - val_accuracy: 0.9473
Epoch 27/50
82/82 [=====] - 4s 52ms/step - loss: 0.0351 - accu
racy: 0.9855 - val_loss: 4.9432 - val_accuracy: 0.7365
Epoch 28/50
82/82 [=====] - 4s 52ms/step - loss: 0.0236 - accu
racy: 0.9912 - val_loss: 0.4876 - val_accuracy: 0.8456
Epoch 29/50
82/82 [=====] - 4s 52ms/step - loss: 0.0095 - accu
racy: 0.9977 - val_loss: 0.2839 - val_accuracy: 0.9228
Epoch 30/50
82/82 [=====] - 4s 54ms/step - loss: 0.0040 - accu
racy: 0.9981 - val_loss: 0.3372 - val_accuracy: 0.9081
Epoch 31/50
82/82 [=====] - 4s 53ms/step - loss: 0.0061 - accu
racy: 0.9977 - val_loss: 0.2955 - val_accuracy: 0.9326
Epoch 32/50
82/82 [=====] - 4s 53ms/step - loss: 0.0050 - accu
racy: 0.9989 - val_loss: 0.1414 - val_accuracy: 0.9669
Epoch 33/50
82/82 [=====] - 4s 53ms/step - loss: 0.0028 - accu
racy: 0.9989 - val_loss: 0.1551 - val_accuracy: 0.9657
Epoch 34/50
82/82 [=====] - 4s 53ms/step - loss: 0.0015 - accu
racy: 0.9996 - val_loss: 0.4561 - val_accuracy: 0.9326
Epoch 35/50
82/82 [=====] - 4s 53ms/step - loss: 0.0407 - accu
racy: 0.9866 - val_loss: 0.2740 - val_accuracy: 0.9375
Epoch 36/50
82/82 [=====] - 4s 55ms/step - loss: 0.0130 - accu
racy: 0.9958 - val_loss: 0.1907 - val_accuracy: 0.9559
Epoch 37/50
82/82 [=====] - 4s 52ms/step - loss: 0.0131 - accu
racy: 0.9958 - val_loss: 0.2205 - val_accuracy: 0.9547
Epoch 38/50

82/82 [=====] - 4s 53ms/step - loss: 0.0042 - accuracy: 0.9989 - val_loss: 0.2442 - val_accuracy: 0.9559
Epoch 39/50
82/82 [=====] - 4s 53ms/step - loss: 0.0181 - accuracy: 0.9935 - val_loss: 0.1871 - val_accuracy: 0.9498
Epoch 40/50
82/82 [=====] - 4s 53ms/step - loss: 0.0308 - accuracy: 0.9870 - val_loss: 0.1944 - val_accuracy: 0.9350
Epoch 41/50
82/82 [=====] - 4s 54ms/step - loss: 0.0111 - accuracy: 0.9958 - val_loss: 0.1574 - val_accuracy: 0.9681
Epoch 42/50
82/82 [=====] - 4s 53ms/step - loss: 0.0065 - accuracy: 0.9973 - val_loss: 0.1775 - val_accuracy: 0.9706
Epoch 43/50
82/82 [=====] - 4s 53ms/step - loss: 0.0031 - accuracy: 0.9992 - val_loss: 0.1849 - val_accuracy: 0.9681
Epoch 44/50
82/82 [=====] - 4s 52ms/step - loss: 0.0090 - accuracy: 0.9981 - val_loss: 0.1711 - val_accuracy: 0.9596
Epoch 45/50
82/82 [=====] - 4s 53ms/step - loss: 0.0265 - accuracy: 0.9893 - val_loss: 0.7702 - val_accuracy: 0.8235
Epoch 46/50
82/82 [=====] - 4s 53ms/step - loss: 0.0156 - accuracy: 0.9943 - val_loss: 0.9570 - val_accuracy: 0.8591
Epoch 47/50
82/82 [=====] - 4s 55ms/step - loss: 0.0174 - accuracy: 0.9931 - val_loss: 0.8341 - val_accuracy: 0.7990
Epoch 48/50
82/82 [=====] - 4s 53ms/step - loss: 0.0052 - accuracy: 0.9989 - val_loss: 0.1974 - val_accuracy: 0.9559
Epoch 49/50
82/82 [=====] - 4s 53ms/step - loss: 0.0038 - accuracy: 0.9981 - val_loss: 0.4734 - val_accuracy: 0.9069
Epoch 50/50
82/82 [=====] - 4s 53ms/step - loss: 0.0082 - accuracy: 0.9973 - val_loss: 0.3808 - val_accuracy: 0.9093
21/21 [=====] - 1s 13ms/step - loss: 0.5435 - accuracy: 0.8901
Epoch 1/50
82/82 [=====] - 5s 55ms/step - loss: 0.2285 - accuracy: 0.9064 - val_loss: 0.5551 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 54ms/step - loss: 0.1682 - accuracy: 0.9354 - val_loss: 0.6731 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 53ms/step - loss: 0.1167 - accuracy: 0.9576 - val_loss: 0.9154 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 53ms/step - loss: 0.1103 - accuracy: 0.9610 - val_loss: 0.1811 - val_accuracy: 0.9314
Epoch 5/50
82/82 [=====] - 4s 52ms/step - loss: 0.1058 - accuracy: 0.9595 - val_loss: 1.9771 - val_accuracy: 0.3309
Epoch 6/50

82/82 [=====] - 4s 53ms/step - loss: 0.0905 - accuracy: 0.9671 - val_loss: 0.1942 - val_accuracy: 0.9289
Epoch 7/50
82/82 [=====] - 4s 54ms/step - loss: 0.0838 - accuracy: 0.9717 - val_loss: 0.2939 - val_accuracy: 0.9449
Epoch 8/50
82/82 [=====] - 4s 53ms/step - loss: 0.0764 - accuracy: 0.9721 - val_loss: 0.1192 - val_accuracy: 0.9608
Epoch 9/50
82/82 [=====] - 4s 53ms/step - loss: 0.0734 - accuracy: 0.9725 - val_loss: 0.3467 - val_accuracy: 0.8493
Epoch 10/50
82/82 [=====] - 4s 53ms/step - loss: 0.0713 - accuracy: 0.9736 - val_loss: 1.0387 - val_accuracy: 0.7635
Epoch 11/50
82/82 [=====] - 4s 52ms/step - loss: 0.0541 - accuracy: 0.9775 - val_loss: 0.2500 - val_accuracy: 0.9020
Epoch 12/50
82/82 [=====] - 4s 53ms/step - loss: 0.0442 - accuracy: 0.9855 - val_loss: 0.4579 - val_accuracy: 0.7855
Epoch 13/50
82/82 [=====] - 4s 55ms/step - loss: 0.0349 - accuracy: 0.9855 - val_loss: 0.6775 - val_accuracy: 0.7537
Epoch 14/50
82/82 [=====] - 4s 53ms/step - loss: 0.0453 - accuracy: 0.9840 - val_loss: 0.2918 - val_accuracy: 0.8958
Epoch 15/50
82/82 [=====] - 4s 53ms/step - loss: 0.0251 - accuracy: 0.9920 - val_loss: 0.1145 - val_accuracy: 0.9571
Epoch 16/50
82/82 [=====] - 4s 52ms/step - loss: 0.0196 - accuracy: 0.9924 - val_loss: 0.1492 - val_accuracy: 0.9620
Epoch 17/50
82/82 [=====] - 4s 53ms/step - loss: 0.0404 - accuracy: 0.9851 - val_loss: 0.1660 - val_accuracy: 0.9498
Epoch 18/50
82/82 [=====] - 4s 53ms/step - loss: 0.0420 - accuracy: 0.9840 - val_loss: 0.1505 - val_accuracy: 0.9559
Epoch 19/50
82/82 [=====] - 4s 54ms/step - loss: 0.0382 - accuracy: 0.9862 - val_loss: 1.1799 - val_accuracy: 0.7794
Epoch 20/50
82/82 [=====] - 4s 53ms/step - loss: 0.0238 - accuracy: 0.9904 - val_loss: 0.3530 - val_accuracy: 0.9130
Epoch 21/50
82/82 [=====] - 4s 53ms/step - loss: 0.0361 - accuracy: 0.9859 - val_loss: 0.2308 - val_accuracy: 0.9326
Epoch 22/50
82/82 [=====] - 4s 53ms/step - loss: 0.0338 - accuracy: 0.9885 - val_loss: 0.2910 - val_accuracy: 0.9167
Epoch 23/50
82/82 [=====] - 4s 53ms/step - loss: 0.0140 - accuracy: 0.9943 - val_loss: 0.3850 - val_accuracy: 0.8983
Epoch 24/50
82/82 [=====] - 4s 53ms/step - loss: 0.0045 - accuracy: 0.9989 - val_loss: 0.2718 - val_accuracy: 0.9498

Epoch 25/50
82/82 [=====] - 5s 55ms/step - loss: 0.0032 - accuracy: 0.9989 - val_loss: 0.2701 - val_accuracy: 0.9547
Epoch 26/50
82/82 [=====] - 4s 53ms/step - loss: 0.0083 - accuracy: 0.9977 - val_loss: 0.3492 - val_accuracy: 0.9375
Epoch 27/50
82/82 [=====] - 4s 53ms/step - loss: 0.0254 - accuracy: 0.9912 - val_loss: 0.8873 - val_accuracy: 0.7770
Epoch 28/50
82/82 [=====] - 4s 53ms/step - loss: 0.0180 - accuracy: 0.9939 - val_loss: 0.5542 - val_accuracy: 0.8505
Epoch 29/50
82/82 [=====] - 4s 53ms/step - loss: 0.0065 - accuracy: 0.9973 - val_loss: 1.5766 - val_accuracy: 0.7855
Epoch 30/50
82/82 [=====] - 4s 55ms/step - loss: 0.0076 - accuracy: 0.9977 - val_loss: 0.2312 - val_accuracy: 0.9669
Epoch 31/50
82/82 [=====] - 4s 52ms/step - loss: 0.0327 - accuracy: 0.9874 - val_loss: 5.0547 - val_accuracy: 0.7304
Epoch 32/50
82/82 [=====] - 4s 53ms/step - loss: 0.0259 - accuracy: 0.9901 - val_loss: 1.4602 - val_accuracy: 0.7696
Epoch 33/50
82/82 [=====] - 4s 52ms/step - loss: 0.0109 - accuracy: 0.9947 - val_loss: 0.1685 - val_accuracy: 0.9571
Epoch 34/50
82/82 [=====] - 4s 52ms/step - loss: 0.0193 - accuracy: 0.9939 - val_loss: 0.2667 - val_accuracy: 0.9510
Epoch 35/50
82/82 [=====] - 4s 54ms/step - loss: 0.0109 - accuracy: 0.9954 - val_loss: 0.2146 - val_accuracy: 0.9400
Epoch 36/50
82/82 [=====] - 4s 52ms/step - loss: 0.0089 - accuracy: 0.9966 - val_loss: 0.2155 - val_accuracy: 0.9694
Epoch 37/50
82/82 [=====] - 4s 53ms/step - loss: 0.0052 - accuracy: 0.9973 - val_loss: 0.2869 - val_accuracy: 0.9375
Epoch 38/50
82/82 [=====] - 4s 53ms/step - loss: 0.0067 - accuracy: 0.9977 - val_loss: 0.2911 - val_accuracy: 0.9350
Epoch 39/50
82/82 [=====] - 4s 53ms/step - loss: 0.0046 - accuracy: 0.9981 - val_loss: 1.0184 - val_accuracy: 0.8076
Epoch 40/50
82/82 [=====] - 4s 53ms/step - loss: 0.0042 - accuracy: 0.9989 - val_loss: 0.3286 - val_accuracy: 0.9534
Epoch 41/50
82/82 [=====] - 4s 54ms/step - loss: 0.0015 - accuracy: 1.0000 - val_loss: 0.2328 - val_accuracy: 0.9645
Epoch 42/50
82/82 [=====] - 4s 53ms/step - loss: 2.5114e-04 - accuracy: 1.0000 - val_loss: 0.2176 - val_accuracy: 0.9657
Epoch 43/50
82/82 [=====] - 4s 53ms/step - loss: 2.6517e-04 -

accuracy: 1.0000 - val_loss: 0.2271 - val_accuracy: 0.9645
Epoch 44/50
82/82 [=====] - 4s 53ms/step - loss: 1.6115e-04 -
accuracy: 1.0000 - val_loss: 0.2217 - val_accuracy: 0.9645
Epoch 45/50
82/82 [=====] - 4s 52ms/step - loss: 9.6979e-05 -
accuracy: 1.0000 - val_loss: 0.2323 - val_accuracy: 0.9657
Epoch 46/50
82/82 [=====] - 4s 54ms/step - loss: 8.8980e-05 -
accuracy: 1.0000 - val_loss: 0.2188 - val_accuracy: 0.9657
Epoch 47/50
82/82 [=====] - 4s 53ms/step - loss: 3.0057e-04 -
accuracy: 1.0000 - val_loss: 0.2167 - val_accuracy: 0.9694
Epoch 48/50
82/82 [=====] - 4s 53ms/step - loss: 0.0185 - accu
racy: 0.9916 - val_loss: 2.9815 - val_accuracy: 0.7451
Epoch 49/50
82/82 [=====] - 4s 53ms/step - loss: 0.0426 - accu
racy: 0.9862 - val_loss: 0.3437 - val_accuracy: 0.9056
Epoch 50/50
82/82 [=====] - 4s 53ms/step - loss: 0.0416 - accu
racy: 0.9859 - val_loss: 0.4073 - val_accuracy: 0.9118
21/21 [=====] - 1s 13ms/step - loss: 0.3851 - accu
racy: 0.9174
Epoch 1/50
82/82 [=====] - 5s 54ms/step - loss: 0.2265 - accu
racy: 0.9133 - val_loss: 0.6538 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 54ms/step - loss: 0.1445 - accu
racy: 0.9457 - val_loss: 0.6684 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 52ms/step - loss: 0.1033 - accu
racy: 0.9599 - val_loss: 0.5928 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 53ms/step - loss: 0.0890 - accu
racy: 0.9652 - val_loss: 0.3203 - val_accuracy: 0.9007
Epoch 5/50
82/82 [=====] - 4s 52ms/step - loss: 0.0980 - accu
racy: 0.9622 - val_loss: 0.2179 - val_accuracy: 0.9142
Epoch 6/50
82/82 [=====] - 4s 52ms/step - loss: 0.0783 - accu
racy: 0.9736 - val_loss: 0.9539 - val_accuracy: 0.4044
Epoch 7/50
82/82 [=====] - 4s 54ms/step - loss: 0.0686 - accu
racy: 0.9755 - val_loss: 0.2370 - val_accuracy: 0.9105
Epoch 8/50
82/82 [=====] - 4s 52ms/step - loss: 0.0536 - accu
racy: 0.9782 - val_loss: 1.2327 - val_accuracy: 0.7304
Epoch 9/50
82/82 [=====] - 4s 52ms/step - loss: 0.0743 - accu
racy: 0.9710 - val_loss: 1.1591 - val_accuracy: 0.7304
Epoch 10/50
82/82 [=====] - 4s 52ms/step - loss: 0.0650 - accu
racy: 0.9740 - val_loss: 0.1720 - val_accuracy: 0.9449
Epoch 11/50
82/82 [=====] - 4s 52ms/step - loss: 0.0514 - accu

racy: 0.9809 - val_loss: 0.1841 - val_accuracy: 0.9228
Epoch 12/50
82/82 [=====] - 4s 52ms/step - loss: 0.0346 - accu
racy: 0.9866 - val_loss: 1.5441 - val_accuracy: 0.6066
Epoch 13/50
82/82 [=====] - 4s 54ms/step - loss: 0.0505 - accu
racy: 0.9817 - val_loss: 0.1770 - val_accuracy: 0.9350
Epoch 14/50
82/82 [=====] - 4s 52ms/step - loss: 0.0453 - accu
racy: 0.9843 - val_loss: 0.1525 - val_accuracy: 0.9559
Epoch 15/50
82/82 [=====] - 4s 53ms/step - loss: 0.0251 - accu
racy: 0.9908 - val_loss: 0.2642 - val_accuracy: 0.9118
Epoch 16/50
82/82 [=====] - 4s 52ms/step - loss: 0.0137 - accu
racy: 0.9958 - val_loss: 0.1425 - val_accuracy: 0.9485
Epoch 17/50
82/82 [=====] - 4s 52ms/step - loss: 0.0279 - accu
racy: 0.9920 - val_loss: 0.1399 - val_accuracy: 0.9608
Epoch 18/50
82/82 [=====] - 4s 53ms/step - loss: 0.0175 - accu
racy: 0.9939 - val_loss: 0.1689 - val_accuracy: 0.9424
Epoch 19/50
82/82 [=====] - 4s 54ms/step - loss: 0.0323 - accu
racy: 0.9885 - val_loss: 0.1362 - val_accuracy: 0.9547
Epoch 20/50
82/82 [=====] - 4s 52ms/step - loss: 0.0149 - accu
racy: 0.9943 - val_loss: 0.1816 - val_accuracy: 0.9436
Epoch 21/50
82/82 [=====] - 4s 53ms/step - loss: 0.0095 - accu
racy: 0.9973 - val_loss: 0.2211 - val_accuracy: 0.9289
Epoch 22/50
82/82 [=====] - 4s 52ms/step - loss: 0.0154 - accu
racy: 0.9954 - val_loss: 0.4122 - val_accuracy: 0.8995
Epoch 23/50
82/82 [=====] - 4s 52ms/step - loss: 0.0104 - accu
racy: 0.9969 - val_loss: 0.4306 - val_accuracy: 0.8946
Epoch 24/50
82/82 [=====] - 4s 52ms/step - loss: 0.0556 - accu
racy: 0.9786 - val_loss: 0.8501 - val_accuracy: 0.7463
Epoch 25/50
82/82 [=====] - 4s 54ms/step - loss: 0.0231 - accu
racy: 0.9908 - val_loss: 0.8682 - val_accuracy: 0.8174
Epoch 26/50
82/82 [=====] - 4s 52ms/step - loss: 0.0161 - accu
racy: 0.9927 - val_loss: 0.9743 - val_accuracy: 0.8370
Epoch 27/50
82/82 [=====] - 4s 53ms/step - loss: 0.0335 - accu
racy: 0.9870 - val_loss: 0.1571 - val_accuracy: 0.9522
Epoch 28/50
82/82 [=====] - 4s 53ms/step - loss: 0.0129 - accu
racy: 0.9958 - val_loss: 0.1730 - val_accuracy: 0.9473
Epoch 29/50
82/82 [=====] - 4s 53ms/step - loss: 0.0210 - accu
racy: 0.9920 - val_loss: 0.2137 - val_accuracy: 0.9375
Epoch 30/50

82/82 [=====] - 4s 54ms/step - loss: 0.0100 - accuracy: 0.9973 - val_loss: 0.2553 - val_accuracy: 0.9387
Epoch 31/50
82/82 [=====] - 4s 52ms/step - loss: 0.0118 - accuracy: 0.9950 - val_loss: 0.1716 - val_accuracy: 0.9510
Epoch 32/50
82/82 [=====] - 4s 53ms/step - loss: 0.0106 - accuracy: 0.9969 - val_loss: 0.2092 - val_accuracy: 0.9301
Epoch 33/50
82/82 [=====] - 4s 53ms/step - loss: 0.0039 - accuracy: 0.9985 - val_loss: 0.1804 - val_accuracy: 0.9632
Epoch 34/50
82/82 [=====] - 4s 53ms/step - loss: 0.0099 - accuracy: 0.9969 - val_loss: 0.1747 - val_accuracy: 0.9596
Epoch 35/50
82/82 [=====] - 4s 54ms/step - loss: 0.0185 - accuracy: 0.9935 - val_loss: 0.5598 - val_accuracy: 0.8137
Epoch 36/50
82/82 [=====] - 4s 53ms/step - loss: 0.0133 - accuracy: 0.9958 - val_loss: 0.1816 - val_accuracy: 0.9424
Epoch 37/50
82/82 [=====] - 4s 52ms/step - loss: 0.0077 - accuracy: 0.9973 - val_loss: 0.6523 - val_accuracy: 0.9020
Epoch 38/50
82/82 [=====] - 4s 52ms/step - loss: 0.0101 - accuracy: 0.9950 - val_loss: 0.2776 - val_accuracy: 0.9338
Epoch 39/50
82/82 [=====] - 4s 53ms/step - loss: 0.0119 - accuracy: 0.9950 - val_loss: 0.6160 - val_accuracy: 0.8971
Epoch 40/50
82/82 [=====] - 4s 52ms/step - loss: 0.0036 - accuracy: 0.9981 - val_loss: 0.2963 - val_accuracy: 0.9400
Epoch 41/50
82/82 [=====] - 4s 55ms/step - loss: 6.3689e-04 - accuracy: 1.0000 - val_loss: 0.1713 - val_accuracy: 0.9657
Epoch 42/50
82/82 [=====] - 4s 52ms/step - loss: 0.0035 - accuracy: 0.9996 - val_loss: 0.2469 - val_accuracy: 0.9547
Epoch 43/50
82/82 [=====] - 4s 53ms/step - loss: 0.0059 - accuracy: 0.9981 - val_loss: 1.1212 - val_accuracy: 0.6238
Epoch 44/50
82/82 [=====] - 4s 52ms/step - loss: 0.0392 - accuracy: 0.9878 - val_loss: 0.3331 - val_accuracy: 0.8983
Epoch 45/50
82/82 [=====] - 4s 53ms/step - loss: 0.0133 - accuracy: 0.9954 - val_loss: 0.2697 - val_accuracy: 0.9449
Epoch 46/50
82/82 [=====] - 4s 53ms/step - loss: 0.0066 - accuracy: 0.9969 - val_loss: 0.2254 - val_accuracy: 0.9400
Epoch 47/50
82/82 [=====] - 4s 55ms/step - loss: 0.0040 - accuracy: 0.9992 - val_loss: 0.1623 - val_accuracy: 0.9706
Epoch 48/50
82/82 [=====] - 4s 53ms/step - loss: 0.0068 - accuracy: 0.9981 - val_loss: 0.1473 - val_accuracy: 0.9669

Epoch 49/50
82/82 [=====] - 4s 54ms/step - loss: 0.0033 - accuracy: 0.9996 - val_loss: 0.1600 - val_accuracy: 0.9657
Epoch 50/50
82/82 [=====] - 4s 53ms/step - loss: 7.4960e-04 - accuracy: 1.0000 - val_loss: 0.1832 - val_accuracy: 0.9620
21/21 [=====] - 0s 12ms/step - loss: 0.2191 - accuracy: 0.9511
Epoch 1/50
82/82 [=====] - 5s 54ms/step - loss: 0.2172 - accuracy: 0.9198 - val_loss: 0.9543 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 54ms/step - loss: 0.1411 - accuracy: 0.9438 - val_loss: 1.5855 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 52ms/step - loss: 0.1197 - accuracy: 0.9583 - val_loss: 1.3822 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 52ms/step - loss: 0.1085 - accuracy: 0.9591 - val_loss: 1.5540 - val_accuracy: 0.7304
Epoch 5/50
82/82 [=====] - 4s 53ms/step - loss: 0.0843 - accuracy: 0.9721 - val_loss: 0.5083 - val_accuracy: 0.8064
Epoch 6/50
82/82 [=====] - 4s 53ms/step - loss: 0.1144 - accuracy: 0.9610 - val_loss: 0.3614 - val_accuracy: 0.8775
Epoch 7/50
82/82 [=====] - 4s 55ms/step - loss: 0.0816 - accuracy: 0.9690 - val_loss: 0.3583 - val_accuracy: 0.8738
Epoch 8/50
82/82 [=====] - 4s 52ms/step - loss: 0.0708 - accuracy: 0.9729 - val_loss: 0.3866 - val_accuracy: 0.8701
Epoch 9/50
82/82 [=====] - 4s 52ms/step - loss: 0.0536 - accuracy: 0.9820 - val_loss: 0.1151 - val_accuracy: 0.9620
Epoch 10/50
82/82 [=====] - 4s 52ms/step - loss: 0.0450 - accuracy: 0.9824 - val_loss: 0.3824 - val_accuracy: 0.9020
Epoch 11/50
82/82 [=====] - 4s 52ms/step - loss: 0.0763 - accuracy: 0.9729 - val_loss: 0.1271 - val_accuracy: 0.9608
Epoch 12/50
82/82 [=====] - 4s 53ms/step - loss: 0.0599 - accuracy: 0.9790 - val_loss: 0.2116 - val_accuracy: 0.9179
Epoch 13/50
82/82 [=====] - 4s 54ms/step - loss: 0.0368 - accuracy: 0.9870 - val_loss: 0.2520 - val_accuracy: 0.9265
Epoch 14/50
82/82 [=====] - 4s 53ms/step - loss: 0.0278 - accuracy: 0.9901 - val_loss: 0.2994 - val_accuracy: 0.9154
Epoch 15/50
82/82 [=====] - 4s 52ms/step - loss: 0.0315 - accuracy: 0.9878 - val_loss: 0.1840 - val_accuracy: 0.9534
Epoch 16/50
82/82 [=====] - 4s 52ms/step - loss: 0.0444 - accuracy: 0.9828 - val_loss: 1.5337 - val_accuracy: 0.7341

Epoch 17/50
82/82 [=====] - 4s 52ms/step - loss: 0.0470 - accuracy: 0.9828 - val_loss: 0.4301 - val_accuracy: 0.8186
Epoch 18/50
82/82 [=====] - 4s 55ms/step - loss: 0.0261 - accuracy: 0.9912 - val_loss: 0.3070 - val_accuracy: 0.8848
Epoch 19/50
82/82 [=====] - 4s 52ms/step - loss: 0.0312 - accuracy: 0.9889 - val_loss: 0.3487 - val_accuracy: 0.9056
Epoch 20/50
82/82 [=====] - 4s 52ms/step - loss: 0.0248 - accuracy: 0.9889 - val_loss: 0.1821 - val_accuracy: 0.9547
Epoch 21/50
82/82 [=====] - 4s 52ms/step - loss: 0.0332 - accuracy: 0.9878 - val_loss: 0.1364 - val_accuracy: 0.9620
Epoch 22/50
82/82 [=====] - 4s 53ms/step - loss: 0.0274 - accuracy: 0.9908 - val_loss: 0.4033 - val_accuracy: 0.9142
Epoch 23/50
82/82 [=====] - 4s 53ms/step - loss: 0.0331 - accuracy: 0.9885 - val_loss: 0.3664 - val_accuracy: 0.9252
Epoch 24/50
82/82 [=====] - 4s 54ms/step - loss: 0.0081 - accuracy: 0.9977 - val_loss: 0.1388 - val_accuracy: 0.9669
Epoch 25/50
82/82 [=====] - 4s 52ms/step - loss: 0.0053 - accuracy: 0.9977 - val_loss: 1.6517 - val_accuracy: 0.8002
Epoch 26/50
82/82 [=====] - 4s 52ms/step - loss: 0.0143 - accuracy: 0.9939 - val_loss: 0.2114 - val_accuracy: 0.9534
Epoch 27/50
82/82 [=====] - 4s 52ms/step - loss: 0.0019 - accuracy: 0.9996 - val_loss: 0.1832 - val_accuracy: 0.9681
Epoch 28/50
82/82 [=====] - 4s 53ms/step - loss: 6.8130e-04 - accuracy: 1.0000 - val_loss: 0.1647 - val_accuracy: 0.9681
Epoch 29/50
82/82 [=====] - 4s 55ms/step - loss: 3.8563e-04 - accuracy: 1.0000 - val_loss: 0.1765 - val_accuracy: 0.9657
Epoch 30/50
82/82 [=====] - 4s 52ms/step - loss: 4.8282e-04 - accuracy: 1.0000 - val_loss: 0.1666 - val_accuracy: 0.9620
Epoch 31/50
82/82 [=====] - 4s 52ms/step - loss: 5.9485e-04 - accuracy: 1.0000 - val_loss: 0.2036 - val_accuracy: 0.9657
Epoch 32/50
82/82 [=====] - 4s 53ms/step - loss: 0.0402 - accuracy: 0.9874 - val_loss: 1.0349 - val_accuracy: 0.8235
Epoch 33/50
82/82 [=====] - 4s 52ms/step - loss: 0.0444 - accuracy: 0.9851 - val_loss: 0.3741 - val_accuracy: 0.8995
Epoch 34/50
82/82 [=====] - 4s 53ms/step - loss: 0.0203 - accuracy: 0.9931 - val_loss: 0.1867 - val_accuracy: 0.9571
Epoch 35/50
82/82 [=====] - 4s 54ms/step - loss: 0.0292 - accuracy:

racy: 0.9897 - val_loss: 0.5378 - val_accuracy: 0.8627
Epoch 36/50
82/82 [=====] - 4s 52ms/step - loss: 0.0245 - accu
racy: 0.9920 - val_loss: 0.2493 - val_accuracy: 0.9216
Epoch 37/50
82/82 [=====] - 4s 52ms/step - loss: 0.0074 - accu
racy: 0.9969 - val_loss: 0.1609 - val_accuracy: 0.9534
Epoch 38/50
82/82 [=====] - 4s 52ms/step - loss: 0.0022 - accu
racy: 0.9996 - val_loss: 0.1669 - val_accuracy: 0.9583
Epoch 39/50
82/82 [=====] - 4s 52ms/step - loss: 0.0018 - accu
racy: 0.9992 - val_loss: 0.2047 - val_accuracy: 0.9669
Epoch 40/50
82/82 [=====] - 4s 52ms/step - loss: 0.0073 - accu
racy: 0.9973 - val_loss: 0.7750 - val_accuracy: 0.8995
Epoch 41/50
82/82 [=====] - 4s 52ms/step - loss: 0.0259 - accu
racy: 0.9916 - val_loss: 0.2367 - val_accuracy: 0.9510
Epoch 42/50
82/82 [=====] - 4s 54ms/step - loss: 0.0025 - accu
racy: 0.9996 - val_loss: 0.1787 - val_accuracy: 0.9657
Epoch 43/50
82/82 [=====] - 4s 52ms/step - loss: 0.0013 - accu
racy: 0.9996 - val_loss: 0.1804 - val_accuracy: 0.9596
Epoch 44/50
82/82 [=====] - 4s 53ms/step - loss: 2.4902e-04 -
accuracy: 1.0000 - val_loss: 0.1740 - val_accuracy: 0.9657
Epoch 45/50
82/82 [=====] - 4s 52ms/step - loss: 0.0035 - accu
racy: 0.9985 - val_loss: 0.2084 - val_accuracy: 0.9632
Epoch 46/50
82/82 [=====] - 4s 52ms/step - loss: 0.0112 - accu
racy: 0.9969 - val_loss: 0.2354 - val_accuracy: 0.9498
Epoch 47/50
82/82 [=====] - 4s 52ms/step - loss: 0.0231 - accu
racy: 0.9920 - val_loss: 0.2109 - val_accuracy: 0.9412
Epoch 48/50
82/82 [=====] - 4s 54ms/step - loss: 0.0095 - accu
racy: 0.9966 - val_loss: 0.3101 - val_accuracy: 0.9179
Epoch 49/50
82/82 [=====] - 4s 52ms/step - loss: 0.0071 - accu
racy: 0.9985 - val_loss: 0.7672 - val_accuracy: 0.8897
Epoch 50/50
82/82 [=====] - 4s 52ms/step - loss: 0.0081 - accu
racy: 0.9977 - val_loss: 0.9254 - val_accuracy: 0.8051
21/21 [=====] - 0s 12ms/step - loss: 0.8211 - accu
racy: 0.8043
Epoch 1/50
82/82 [=====] - 6s 55ms/step - loss: 0.2299 - accu
racy: 0.9071 - val_loss: 0.5721 - val_accuracy: 0.7304
Epoch 2/50
82/82 [=====] - 4s 53ms/step - loss: 0.1480 - accu
racy: 0.9492 - val_loss: 0.6506 - val_accuracy: 0.7304
Epoch 3/50
82/82 [=====] - 4s 55ms/step - loss: 0.1266 - accu

racy: 0.9538 - val_loss: 1.1161 - val_accuracy: 0.7304
Epoch 4/50
82/82 [=====] - 4s 53ms/step - loss: 0.1108 - accu
racy: 0.9618 - val_loss: 1.0343 - val_accuracy: 0.7304
Epoch 5/50
82/82 [=====] - 4s 53ms/step - loss: 0.0890 - accu
racy: 0.9668 - val_loss: 2.8976 - val_accuracy: 0.7304
Epoch 6/50
82/82 [=====] - 4s 53ms/step - loss: 0.0927 - accu
racy: 0.9683 - val_loss: 0.3111 - val_accuracy: 0.8652
Epoch 7/50
82/82 [=====] - 4s 53ms/step - loss: 0.0769 - accu
racy: 0.9725 - val_loss: 0.4605 - val_accuracy: 0.8529
Epoch 8/50
82/82 [=====] - 4s 53ms/step - loss: 0.0574 - accu
racy: 0.9801 - val_loss: 0.6019 - val_accuracy: 0.7414
Epoch 9/50
82/82 [=====] - 4s 55ms/step - loss: 0.0567 - accu
racy: 0.9813 - val_loss: 1.1140 - val_accuracy: 0.7537
Epoch 10/50
82/82 [=====] - 4s 53ms/step - loss: 0.0395 - accu
racy: 0.9859 - val_loss: 0.1091 - val_accuracy: 0.9632
Epoch 11/50
82/82 [=====] - 4s 53ms/step - loss: 0.0305 - accu
racy: 0.9889 - val_loss: 1.1930 - val_accuracy: 0.5968
Epoch 12/50
82/82 [=====] - 4s 53ms/step - loss: 0.0419 - accu
racy: 0.9855 - val_loss: 0.2941 - val_accuracy: 0.8689
Epoch 13/50
82/82 [=====] - 4s 53ms/step - loss: 0.0393 - accu
racy: 0.9843 - val_loss: 0.6528 - val_accuracy: 0.7708
Epoch 14/50
82/82 [=====] - 4s 53ms/step - loss: 0.0460 - accu
racy: 0.9828 - val_loss: 0.3387 - val_accuracy: 0.8578
Epoch 15/50
82/82 [=====] - 4s 53ms/step - loss: 0.0355 - accu
racy: 0.9889 - val_loss: 0.1723 - val_accuracy: 0.9449
Epoch 16/50
82/82 [=====] - 4s 53ms/step - loss: 0.0358 - accu
racy: 0.9893 - val_loss: 3.9434 - val_accuracy: 0.7316
Epoch 17/50
82/82 [=====] - 4s 52ms/step - loss: 0.0328 - accu
racy: 0.9866 - val_loss: 0.4489 - val_accuracy: 0.8971
Epoch 18/50
82/82 [=====] - 4s 55ms/step - loss: 0.0094 - accu
racy: 0.9985 - val_loss: 0.1745 - val_accuracy: 0.9620
Epoch 19/50
82/82 [=====] - 4s 53ms/step - loss: 0.0119 - accu
racy: 0.9950 - val_loss: 0.1984 - val_accuracy: 0.9510
Epoch 20/50
82/82 [=====] - 4s 53ms/step - loss: 0.0315 - accu
racy: 0.9882 - val_loss: 0.2612 - val_accuracy: 0.9118
Epoch 21/50
82/82 [=====] - 4s 53ms/step - loss: 0.0195 - accu
racy: 0.9924 - val_loss: 0.2562 - val_accuracy: 0.9387
Epoch 22/50

82/82 [=====] - 4s 53ms/step - loss: 0.0125 - accuracy: 0.9958 - val_loss: 0.1508 - val_accuracy: 0.9669
Epoch 23/50
82/82 [=====] - 4s 53ms/step - loss: 0.0135 - accuracy: 0.9954 - val_loss: 0.3670 - val_accuracy: 0.9265
Epoch 24/50
82/82 [=====] - 5s 55ms/step - loss: 0.0177 - accuracy: 0.9947 - val_loss: 0.2576 - val_accuracy: 0.9118
Epoch 25/50
82/82 [=====] - 4s 53ms/step - loss: 0.0083 - accuracy: 0.9969 - val_loss: 2.3821 - val_accuracy: 0.7451
Epoch 26/50
82/82 [=====] - 4s 53ms/step - loss: 0.0365 - accuracy: 0.9878 - val_loss: 1.1250 - val_accuracy: 0.7402
Epoch 27/50
82/82 [=====] - 4s 53ms/step - loss: 0.0266 - accuracy: 0.9912 - val_loss: 0.1604 - val_accuracy: 0.9522
Epoch 28/50
82/82 [=====] - 4s 53ms/step - loss: 0.0128 - accuracy: 0.9950 - val_loss: 0.1714 - val_accuracy: 0.9583
Epoch 29/50
82/82 [=====] - 4s 55ms/step - loss: 0.0047 - accuracy: 0.9985 - val_loss: 0.3412 - val_accuracy: 0.9326
Epoch 30/50
82/82 [=====] - 4s 53ms/step - loss: 0.0016 - accuracy: 0.9992 - val_loss: 0.1740 - val_accuracy: 0.9657
Epoch 31/50
82/82 [=====] - 4s 54ms/step - loss: 0.0079 - accuracy: 0.9977 - val_loss: 0.1750 - val_accuracy: 0.9681
Epoch 32/50
82/82 [=====] - 4s 53ms/step - loss: 0.0186 - accuracy: 0.9935 - val_loss: 0.1622 - val_accuracy: 0.9632
Epoch 33/50
82/82 [=====] - 4s 55ms/step - loss: 0.0087 - accuracy: 0.9977 - val_loss: 0.1663 - val_accuracy: 0.9657
Epoch 34/50
82/82 [=====] - 4s 52ms/step - loss: 0.0152 - accuracy: 0.9943 - val_loss: 0.8044 - val_accuracy: 0.7451
Epoch 35/50
82/82 [=====] - 4s 53ms/step - loss: 0.0136 - accuracy: 0.9966 - val_loss: 0.2380 - val_accuracy: 0.9289
Epoch 36/50
82/82 [=====] - 4s 53ms/step - loss: 0.0108 - accuracy: 0.9958 - val_loss: 0.7104 - val_accuracy: 0.8713
Epoch 37/50
82/82 [=====] - 4s 53ms/step - loss: 0.0089 - accuracy: 0.9969 - val_loss: 0.2572 - val_accuracy: 0.9216
Epoch 38/50
82/82 [=====] - 4s 53ms/step - loss: 0.0140 - accuracy: 0.9950 - val_loss: 0.4356 - val_accuracy: 0.9338
Epoch 39/50
82/82 [=====] - 4s 55ms/step - loss: 0.0188 - accuracy: 0.9924 - val_loss: 0.2121 - val_accuracy: 0.9583
Epoch 40/50
82/82 [=====] - 4s 53ms/step - loss: 0.0072 - accuracy: 0.9977 - val_loss: 0.1793 - val_accuracy: 0.9657

```

Epoch 41/50
82/82 [=====] - 4s 53ms/step - loss: 0.0010 - accu
racy: 1.0000 - val_loss: 0.1972 - val_accuracy: 0.9694
Epoch 42/50
82/82 [=====] - 4s 54ms/step - loss: 0.0042 - accu
racy: 0.9985 - val_loss: 0.2267 - val_accuracy: 0.9510
Epoch 43/50
82/82 [=====] - 4s 53ms/step - loss: 0.0082 - accu
racy: 0.9966 - val_loss: 0.1362 - val_accuracy: 0.9632
Epoch 44/50
82/82 [=====] - 4s 55ms/step - loss: 0.0359 - accu
racy: 0.9912 - val_loss: 0.2320 - val_accuracy: 0.9191
Epoch 45/50
82/82 [=====] - 4s 53ms/step - loss: 0.0153 - accu
racy: 0.9954 - val_loss: 0.2813 - val_accuracy: 0.9534
Epoch 46/50
82/82 [=====] - 4s 53ms/step - loss: 0.0015 - accu
racy: 1.0000 - val_loss: 0.2603 - val_accuracy: 0.9632
Epoch 47/50
82/82 [=====] - 4s 52ms/step - loss: 7.8274e-04 -
accuracy: 0.9996 - val_loss: 0.5136 - val_accuracy: 0.8873
Epoch 48/50
82/82 [=====] - 4s 53ms/step - loss: 0.0029 - accu
racy: 0.9989 - val_loss: 0.2047 - val_accuracy: 0.9669
Epoch 49/50
82/82 [=====] - 4s 54ms/step - loss: 9.7579e-04 -
accuracy: 0.9996 - val_loss: 0.3279 - val_accuracy: 0.9632

```

```
In [106]: save_result('CNN #6', results_train[1], results_test[1], cvs[0], cvs[1], cv
s[2], cvs[3], cvs[4])
```

Out[106]:

	model_name	Train Accuracy	Test Accuracy	CV1	CV2	CV3	CV4	CV5	CV_Std	CV_avg
0	CNN #4	1.000	0.955	0.963	0.972	0.953	0.965	0.950	0.009	0.961
0	CNN #3	1.000	0.947	0.957	0.969	0.957	0.965	0.948	0.008	0.959
0	CNN #2	1.000	0.949	0.957	0.969	0.956	0.965	0.945	0.009	0.958
0	CNN #1	0.947	0.931	0.933	0.943	0.934	0.943	0.917	0.011	0.934
0	CNN #6	0.939	0.893	0.890	0.917	0.951	0.804	0.902	0.055	0.893
0	Initial Model	0.730	0.730	0.937	0.732	0.735	0.709	0.725	0.095	0.768

Prediction for Confusion Matrix

```
In [107]: predictions = model6.predict(x = test_images, steps = 10, verbose=0)
```

```
In [108]: pred_check = np.round(predictions)
```

```
In [109]: pred_check = pred_check[:]  
pred_check = pred_check.flatten()  
pred_check
```

```
Out[109]: array([1., 1., 1., ..., 0., 0., 1.], dtype=float32)
```

```
In [110]: test_labels = test_labels[:,0]  
test_labels
```

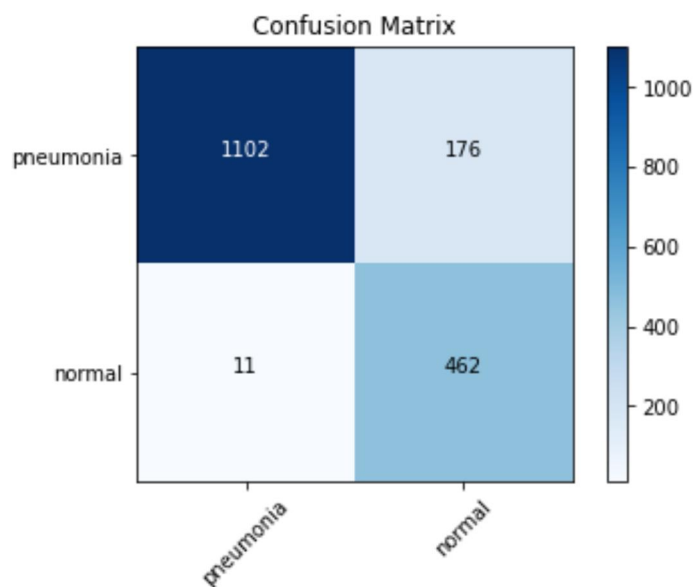
```
Out[110]: array([1., 1., 1., ..., 0., 0., 0.], dtype=float32)
```

```
In [111]: cm = confusion_matrix(y_true=test_labels, y_pred=pred_check)
```

```
In [112]: cm_plot_labels = ['pneumonia', 'normal']  
plot_confusion_matrix(cm=cm, classes=cm_plot_labels, title='Confusion Matrix')
```

Confusion matrix, without normalization

```
[[1102  176]  
 [  11  462]]
```



```

In [120]: # SOURCE: The origin of this confusion matrix code was found on medium, '
# from https://medium.com/@dtuk81/confusion-matrix-visualization-fc31e3f30f
ea
def make_confusion_matrix(cf,
                           group_names=None,
                           categories='auto',
                           count=True,
                           percent=True,
                           cbar=True,
                           xyticks=True,
                           xyplotlabels=True,
                           sum_stats=True,
                           figsize=None,
                           cmap='Blues',
                           title=None):

    # CODE TO GENERATE SUMMARY STATISTICS & TEXT FOR SUMMARY STATS
    if sum_stats:
        #Accuracy is sum of diagonal divided by total observations
        accuracy = np.trace(cf) / float(np.sum(cf))

        #if it is a binary confusion matrix, show some more stats
        if len(cf)==2:
            #Metrics for Binary Confusion Matrices
            a = cf[0,0]
            b = cf[0,1]
            c = cf[1,0]
            d = cf[1,1]
            tn = ((a / (a+b))*100).round(2).astype(str) + '%'
            fp = ((b / (a+b))*100).round(2).astype(str) + '%'
            fn = ((c / (c+d))*100).round(2).astype(str) + '%'
            tp = ((d / (c+d))*100).round(2).astype(str) + '%'
            precision = cf[1,1] / sum(cf[:,1])
            recall = cf[1,1] / sum(cf[1,:])
            f1_score = 2*precision*recall / (precision + recall)
            rwf_score = 2*precision* (recall*2) /(precision + (recall*2))
            stats_text = "\n\nAccuracy={: 0.3f}\nPrecision={: 0.3f}\nRecall =
{: 0.3f}\nF1 Score={: 0.3f}\n\nRecall-Weighted F Score={: 0.3f}".format(
                accuracy,precision,recall,f1_score,rwf_score)
        else:
            stats_text = "\n\nAccuracy={: 0.3f}".format(accuracy)
    else:
        stats_text = ""

    # CODE TO GENERATE TEXT INSIDE EACH SQUARE
    blanks = ['' for i in range(cf.size)]

    if group_names and len(group_names)==cf.size:
        group_labels = ["{}\n".format(value) for value in group_names]
    else:
        group_labels = blanks

    if count:
        group_counts = ["{0:0.0f}\n".format(value) for value in cf.flatten
()]

```

```

else:
    group_counts = blanks

if percent:
    group_percentages = [tn, fp, fn, tp]
    # old = group_percentages = ["{0:.2%}".format(value) for value in c
f.flatten()/np.sum(cf)]
else:
    group_percentages = blanks

box_labels = [f"{v1}{v2}{v3}".strip() for v1, v2, v3 in zip(group_labels, group_counts, group_percentages)]
box_labels = np.asarray(box_labels).reshape(cf.shape[0], cf.shape[1])

# SET FIGURE PARAMETERS ACCORDING TO OTHER ARGUMENTS
if figsize==None:
    #Get default figure size if not set
    figsize = plt.rcParams.get('figure.figsize')

if xyticks==False:
    #Do not show categories if xyticks is False
    categories=False

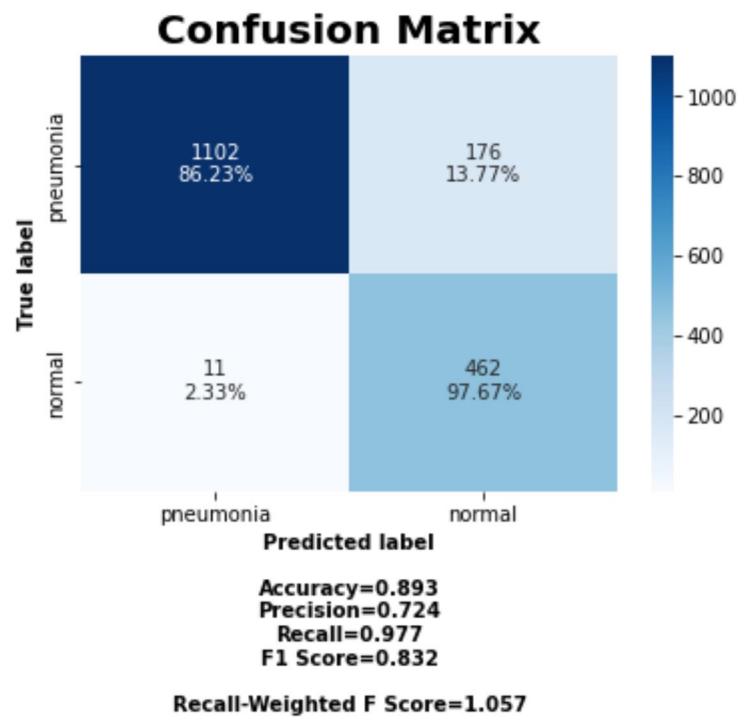
# MAKE THE HEATMAP VISUALIZATION
plt.figure(figsize=figsize)
sns.heatmap(cf, annot=box_labels, fmt="", cmap=cmap, cbar=cbar, xticklabels=
categories, yticklabels=categories)

if xyplotlabels:
    plt.ylabel('True Label', weight = 'bold')
    plt.xlabel('Predicted Label' + stats_text, weight = 'bold')
else:
    plt.xlabel(stats_text)

if title:
    plt.title(title, size = 20, weight = 'bold')

```

```
In [124]: cm_plot_labels = ['pneumonia', 'normal']  
make_confusion_matrix(cm, categories = cm_plot_labels, title='Confusion Matrix')
```



Conclusion

I was unable to improve performance past CNN #4, making CNN #4 the final model for this project.


```
In [119]: # Model 4 Summary
model 4. summary()
```

Model: "sequential_19"

Layer (type)	Output Shape	Param #
conv2d_66 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_66 (MaxPooling)	(None, 31, 31, 32)	0
conv2d_67 (Conv2D)	(None, 28, 28, 32)	16416
batch_normalization_12 (Batch Normalization)	(None, 28, 28, 32)	128
max_pooling2d_67 (MaxPooling)	(None, 14, 14, 32)	0
conv2d_68 (Conv2D)	(None, 12, 12, 64)	18496
max_pooling2d_68 (MaxPooling)	(None, 6, 6, 64)	0
conv2d_69 (Conv2D)	(None, 4, 4, 128)	73856
batch_normalization_13 (Batch Normalization)	(None, 4, 4, 128)	512
max_pooling2d_69 (MaxPooling)	(None, 2, 2, 128)	0
flatten_18 (Flatten)	(None, 512)	0
dense_60 (Dense)	(None, 64)	32832
dropout_6 (Dropout)	(None, 64)	0
dense_61 (Dense)	(None, 1)	65
Total params: 143,201		
Trainable params: 142,881		
Non-trainable params: 320		

Final Results

The final result included 1,236 true positives, 437 true negatives, 36 false negatives, and 42 false positives.

The total accuracy was 95.5 percent.

Recall is the number of true positives divided by the total number of elements that actually belong to the positive class -i.e., true positives plus false negatives.

- Recall equaled .93.
- Precision equaled .912.
- The F1 Score equaled .918.

In product terms, this means that we could expect the model to correctly pick if an individual has pneumonia based on their x-ray 95.5 percent of the time.

If I could further this project, I would try other neural network models to see if there is a possibly better techniques for the modeling, such as LIME (locally interpretable model-agnostic explanations).

I would also attempt further data augmentation. The data augmentation I attempted did not improve the performance of the model, although that portion of the model was cut out for brevity, along with numerous other versions of the CNNs.

Thank you.