



In [1]:

```
#importing the libararies required
from keras.layers import Input, Lambda, Dense, Flatten
from keras.models import Model
from keras.applications.vgg16 import VGG16
from keras.applications.vgg16 import preprocess_input
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, CS
VLogger, ReduceLROnPlateau
from keras.models import Sequential
import numpy as np
from glob import glob
import matplotlib.pyplot as plt
```

```
/opt/conda/lib/python3.10/site-packages/scipy/__init__.py:146: UserWar
ning: A NumPy version >=1.16.5 and <1.23.0 is required for this versio
n of SciPy (detected version 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversio
n}")
```

In [2]:

```
IMAGE_SIZE = [224, 224]
```

In [3]:

```
#path of the dataset
train_dir='/kaggle/input/stanford-dogs-dataset/images/Images'
```

In [4]:

```
# adding preprocessing layer to VGG16
vgg = VGG16(input_shape=IMAGE_SIZE + [3], weights='imagenet', include_top
=False)

# not train existing weights
for layer in vgg.layers:
    layer.trainable = False
```

Downloading data from [https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16\\_weights\\_tf\\_dim\\_ordering\\_tf\\_kernels\\_notop.h5](https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5)  
58889256/58889256 [=====] - 2s 0us/step

In [5]:

```
folders = glob('/kaggle/input/stanford-dogs-dataset/images/Images/*')
len(folders)
```

Out[5]:

120

In [6]:

```
x = Flatten()(vgg.output)
#activating dense layers tot he model
prediction = Dense(len(folders), activation='softmax')(x)
```

In [7]:

```
# create a model object
model = Model(inputs=vgg.input, outputs=prediction)

# structure of the model
model.summary()
```

Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808

block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 120)	3010680

=====

Total params: 17,725,368

Trainable params: 3,010,680

Non-trainable params: 14,714,688

-----

In [8]:

```
#compiling the model
model.compile(
    loss='binary_crossentropy',
    optimizer='adam',
    metrics=['accuracy']
)
```

In [9]:

```
from keras.preprocessing.image import ImageDataGenerator
#to rescale the training images
train_datagen = ImageDataGenerator(rescale = 1./255,
                                    shear_range = 0.2,
                                    zoom_range = 0.2,
                                    horizontal_flip = True)

#to rescale the validation images
valid_datagen = ImageDataGenerator(rescale = 1./255, validation_split=0.2)

#to create the training data
training_model = train_datagen.flow_from_directory(
    train_dir,
    target_size= (224, 224),
    color_mode= 'rgb',
    batch_size= 32,
    class_mode= 'categorical',
    subset='training',
    shuffle= True,
    seed= 1337
)

#to create the validating data
validating_model = valid_datagen.flow_from_directory(
    train_dir,
    target_size= (224, 224),
    color_mode= 'rgb',
    batch_size= 32,
    class_mode= 'categorical',
    subset='validation',
    shuffle= True,
    seed= 1337
)
```

Found 20580 images belonging to 120 classes.

Found 4072 images belonging to 120 classes.

In [10]:

```
print(len(training_model))  
print(len(validating_model))
```

644

128

In [11]:

```
from keras.callbacks import EarlyStopping, ModelCheckpoint  
  
# Checkpoint and monitoring val accuracy  
mc = ModelCheckpoint(  
    'dog_class_model.h5',  
    monitor='val_accuracy',  
    mode='max',  
    verbose=1,  
    save_best_only=True,  
)
```

In [12]:

```
#training the model
r = model.fit(
    training_model,
    epochs=100,
    steps_per_epoch=len(training_model),
    validation_data=validating_model,
    validation_steps=len(validating_model),
    verbose = 2,
    callbacks=[mc],
    shuffle = True
)
!mkdir -p saved_model
model.save('saved_model/dog_class_model.h5')
```



Epoch 1/100

Epoch 1: val\_accuracy improved from -inf to 0.31852, saving model to dog\_class\_model.h5

644/644 - 480s - loss: 0.0584 - accuracy: 0.1017 - val\_loss: 0.0390 - val\_accuracy: 0.3185 - 480s/epoch - 745ms/step

Epoch 2/100

Epoch 2: val\_accuracy improved from 0.31852 to 0.49361, saving model to dog\_class\_model.h5

644/644 - 337s - loss: 0.0427 - accuracy: 0.2758 - val\_loss: 0.0303 - val\_accuracy: 0.4936 - 337s/epoch - 523ms/step

Epoch 3/100

Epoch 3: val\_accuracy improved from 0.49361 to 0.54715, saving model to dog\_class\_model.h5

644/644 - 328s - loss: 0.0363 - accuracy: 0.3853 - val\_loss: 0.0320 - val\_accuracy: 0.5472 - 328s/epoch - 510ms/step

Epoch 4/100

Epoch 4: val\_accuracy improved from 0.54715 to 0.62083, saving model to dog\_class\_model.h5

644/644 - 325s - loss: 0.0325 - accuracy: 0.4601 - val\_loss: 0.0245 - val\_accuracy: 0.6208 - 325s/epoch - 505ms/step

Epoch 5/100

Epoch 5: val\_accuracy improved from 0.62083 to 0.69106, saving model to dog\_class\_model.h5

644/644 - 326s - loss: 0.0276 - accuracy: 0.5371 - val\_loss: 0.0205 - val\_accuracy: 0.6911 - 326s/epoch - 507ms/step

Epoch 6/100

Epoch 6: val\_accuracy improved from 0.69106 to 0.73551, saving model to dog\_class\_model.h5

644/644 - 326s - loss: 0.0260 - accuracy: 0.5729 - val\_loss: 0.0199 - val\_accuracy: 0.7355 - 326s/epoch - 506ms/step

Epoch 7/100

Epoch 7: val\_accuracy improved from 0.73551 to 0.78340, saving model to dog\_class\_model.h5

644/644 - 324s - loss: 0.0242 - accuracy: 0.6237 - val\_loss: 0.0168 -  
val\_accuracy: 0.7834 - 324s/epoch - 503ms/step  
Epoch 8/100

Epoch 8: val\_accuracy did not improve from 0.78340  
644/644 - 329s - loss: 0.0211 - accuracy: 0.6694 - val\_loss: 0.0171 -  
val\_accuracy: 0.7827 - 329s/epoch - 511ms/step  
Epoch 9/100

Epoch 9: val\_accuracy improved from 0.78340 to 0.81508, saving model to  
dog\_class\_model.h5  
644/644 - 327s - loss: 0.0197 - accuracy: 0.6944 - val\_loss: 0.0156 -  
val\_accuracy: 0.8151 - 327s/epoch - 507ms/step  
Epoch 10/100

Epoch 10: val\_accuracy improved from 0.81508 to 0.84774, saving model  
to dog\_class\_model.h5  
644/644 - 325s - loss: 0.0190 - accuracy: 0.7159 - val\_loss: 0.0154 -  
val\_accuracy: 0.8477 - 325s/epoch - 505ms/step  
Epoch 11/100

Epoch 11: val\_accuracy improved from 0.84774 to 0.86076, saving model  
to dog\_class\_model.h5  
644/644 - 324s - loss: 0.0172 - accuracy: 0.7442 - val\_loss: 0.0128 -  
val\_accuracy: 0.8608 - 324s/epoch - 504ms/step  
Epoch 12/100

Epoch 12: val\_accuracy improved from 0.86076 to 0.87181, saving model  
to dog\_class\_model.h5  
644/644 - 323s - loss: 0.0168 - accuracy: 0.7564 - val\_loss: 0.0119 -  
val\_accuracy: 0.8718 - 323s/epoch - 501ms/step  
Epoch 13/100

Epoch 13: val\_accuracy improved from 0.87181 to 0.89514, saving model  
to dog\_class\_model.h5  
644/644 - 324s - loss: 0.0150 - accuracy: 0.7852 - val\_loss: 0.0108 -  
val\_accuracy: 0.8951 - 324s/epoch - 503ms/step  
Epoch 14/100

Epoch 14: val\_accuracy did not improve from 0.89514  
644/644 - 328s - loss: 0.0144 - accuracy: 0.7948 - val\_loss: 0.0117 -

val\_accuracy: 0.8829 - 328s/epoch - 509ms/step

Epoch 15/100

Epoch 15: val\_accuracy did not improve from 0.89514

644/644 - 334s - loss: 0.0145 - accuracy: 0.8010 - val\_loss: 0.0138 -

val\_accuracy: 0.8922 - 334s/epoch - 518ms/step

Epoch 16/100

Epoch 16: val\_accuracy did not improve from 0.89514

644/644 - 329s - loss: 0.0148 - accuracy: 0.8075 - val\_loss: 0.0115 -

val\_accuracy: 0.8892 - 329s/epoch - 511ms/step

Epoch 17/100

Epoch 17: val\_accuracy improved from 0.89514 to 0.90570, saving model to dog\_class\_model.h5

644/644 - 335s - loss: 0.0130 - accuracy: 0.8290 - val\_loss: 0.0117 -

val\_accuracy: 0.9057 - 335s/epoch - 520ms/step

Epoch 18/100

Epoch 18: val\_accuracy improved from 0.90570 to 0.93811, saving model to dog\_class\_model.h5

644/644 - 330s - loss: 0.0124 - accuracy: 0.8322 - val\_loss: 0.0078 -

val\_accuracy: 0.9381 - 330s/epoch - 513ms/step

Epoch 19/100

Epoch 19: val\_accuracy improved from 0.93811 to 0.94769, saving model to dog\_class\_model.h5

644/644 - 334s - loss: 0.0117 - accuracy: 0.8513 - val\_loss: 0.0081 -

val\_accuracy: 0.9477 - 334s/epoch - 519ms/step

Epoch 20/100

Epoch 20: val\_accuracy did not improve from 0.94769

644/644 - 336s - loss: 0.0119 - accuracy: 0.8525 - val\_loss: 0.0083 -

val\_accuracy: 0.9420 - 336s/epoch - 521ms/step

Epoch 21/100

Epoch 21: val\_accuracy did not improve from 0.94769

644/644 - 334s - loss: 0.0115 - accuracy: 0.8561 - val\_loss: 0.0104 -

val\_accuracy: 0.9271 - 334s/epoch - 519ms/step

Epoch 22/100

Epoch 22: val\_accuracy did not improve from 0.94769  
644/644 - 335s - loss: 0.0112 - accuracy: 0.8673 - val\_loss: 0.0081 -  
val\_accuracy: 0.9416 - 335s/epoch - 519ms/step  
Epoch 23/100

Epoch 23: val\_accuracy did not improve from 0.94769  
644/644 - 333s - loss: 0.0107 - accuracy: 0.8755 - val\_loss: 0.0176 -  
val\_accuracy: 0.8956 - 333s/epoch - 516ms/step  
Epoch 24/100

Epoch 24: val\_accuracy improved from 0.94769 to 0.95334, saving model  
to dog\_class\_model.h5  
644/644 - 333s - loss: 0.0106 - accuracy: 0.8775 - val\_loss: 0.0069 -  
val\_accuracy: 0.9533 - 333s/epoch - 517ms/step  
Epoch 25/100

Epoch 25: val\_accuracy did not improve from 0.95334  
644/644 - 327s - loss: 0.0096 - accuracy: 0.8892 - val\_loss: 0.0096 -  
val\_accuracy: 0.9344 - 327s/epoch - 507ms/step  
Epoch 26/100

Epoch 26: val\_accuracy did not improve from 0.95334  
644/644 - 331s - loss: 0.0105 - accuracy: 0.8908 - val\_loss: 0.0081 -  
val\_accuracy: 0.9511 - 331s/epoch - 515ms/step  
Epoch 27/100

Epoch 27: val\_accuracy improved from 0.95334 to 0.96439, saving model  
to dog\_class\_model.h5  
644/644 - 331s - loss: 0.0099 - accuracy: 0.8897 - val\_loss: 0.0062 -  
val\_accuracy: 0.9644 - 331s/epoch - 514ms/step  
Epoch 28/100

Epoch 28: val\_accuracy did not improve from 0.96439  
644/644 - 330s - loss: 0.0092 - accuracy: 0.8978 - val\_loss: 0.0066 -  
val\_accuracy: 0.9612 - 330s/epoch - 512ms/step  
Epoch 29/100

Epoch 29: val\_accuracy did not improve from 0.96439  
644/644 - 330s - loss: 0.0091 - accuracy: 0.9037 - val\_loss: 0.0076 -  
val\_accuracy: 0.9617 - 330s/epoch - 512ms/step  
Epoch 30/100

Epoch 30: val\_accuracy did not improve from 0.96439  
644/644 - 323s - loss: 0.0092 - accuracy: 0.9057 - val\_loss: 0.0069 -  
val\_accuracy: 0.9543 - 323s/epoch - 502ms/step  
Epoch 31/100

Epoch 31: val\_accuracy improved from 0.96439 to 0.97250, saving model  
to dog\_class\_model.h5  
644/644 - 331s - loss: 0.0086 - accuracy: 0.9088 - val\_loss: 0.0065 -  
val\_accuracy: 0.9725 - 331s/epoch - 515ms/step  
Epoch 32/100

Epoch 32: val\_accuracy improved from 0.97250 to 0.97765, saving model  
to dog\_class\_model.h5  
644/644 - 325s - loss: 0.0086 - accuracy: 0.9109 - val\_loss: 0.0055 -  
val\_accuracy: 0.9777 - 325s/epoch - 504ms/step  
Epoch 33/100

Epoch 33: val\_accuracy did not improve from 0.97765  
644/644 - 322s - loss: 0.0088 - accuracy: 0.9094 - val\_loss: 0.0076 -  
val\_accuracy: 0.9612 - 322s/epoch - 501ms/step  
Epoch 34/100

Epoch 34: val\_accuracy did not improve from 0.97765  
644/644 - 324s - loss: 0.0087 - accuracy: 0.9128 - val\_loss: 0.0071 -  
val\_accuracy: 0.9732 - 324s/epoch - 504ms/step  
Epoch 35/100

Epoch 35: val\_accuracy did not improve from 0.97765  
644/644 - 326s - loss: 0.0086 - accuracy: 0.9178 - val\_loss: 0.0076 -  
val\_accuracy: 0.9585 - 326s/epoch - 507ms/step  
Epoch 36/100

Epoch 36: val\_accuracy did not improve from 0.97765  
644/644 - 323s - loss: 0.0080 - accuracy: 0.9236 - val\_loss: 0.0072 -  
val\_accuracy: 0.9629 - 323s/epoch - 501ms/step  
Epoch 37/100

Epoch 37: val\_accuracy did not improve from 0.97765  
644/644 - 328s - loss: 0.0075 - accuracy: 0.9269 - val\_loss: 0.0070 -  
val\_accuracy: 0.9641 - 328s/epoch - 509ms/step

Epoch 38/100

Epoch 38: val\_accuracy did not improve from 0.97765

644/644 - 336s - loss: 0.0076 - accuracy: 0.9296 - val\_loss: 0.0054 -  
val\_accuracy: 0.9764 - 336s/epoch - 522ms/step

Epoch 39/100

Epoch 39: val\_accuracy improved from 0.97765 to 0.97814, saving model  
to dog\_class\_model.h5

644/644 - 330s - loss: 0.0079 - accuracy: 0.9269 - val\_loss: 0.0057 -  
val\_accuracy: 0.9781 - 330s/epoch - 512ms/step

Epoch 40/100

Epoch 40: val\_accuracy improved from 0.97814 to 0.97888, saving model  
to dog\_class\_model.h5

644/644 - 336s - loss: 0.0078 - accuracy: 0.9289 - val\_loss: 0.0066 -  
val\_accuracy: 0.9789 - 336s/epoch - 523ms/step

Epoch 41/100

Epoch 41: val\_accuracy improved from 0.97888 to 0.98011, saving model  
to dog\_class\_model.h5

644/644 - 336s - loss: 0.0071 - accuracy: 0.9354 - val\_loss: 0.0054 -  
val\_accuracy: 0.9801 - 336s/epoch - 522ms/step

Epoch 42/100

Epoch 42: val\_accuracy did not improve from 0.98011

644/644 - 328s - loss: 0.0074 - accuracy: 0.9357 - val\_loss: 0.0050 -  
val\_accuracy: 0.9801 - 328s/epoch - 510ms/step

Epoch 43/100

Epoch 43: val\_accuracy improved from 0.98011 to 0.98207, saving model  
to dog\_class\_model.h5

644/644 - 336s - loss: 0.0071 - accuracy: 0.9346 - val\_loss: 0.0066 -  
val\_accuracy: 0.9821 - 336s/epoch - 522ms/step

Epoch 44/100

Epoch 44: val\_accuracy did not improve from 0.98207

644/644 - 332s - loss: 0.0068 - accuracy: 0.9419 - val\_loss: 0.0052 -  
val\_accuracy: 0.9774 - 332s/epoch - 516ms/step

Epoch 45/100

Epoch 45: val\_accuracy improved from 0.98207 to 0.98281, saving model to dog\_class\_model.h5

644/644 - 330s - loss: 0.0070 - accuracy: 0.9402 - val\_loss: 0.0051 - val\_accuracy: 0.9828 - 330s/epoch - 512ms/step

Epoch 46/100

Epoch 46: val\_accuracy improved from 0.98281 to 0.98502, saving model to dog\_class\_model.h5

644/644 - 329s - loss: 0.0067 - accuracy: 0.9430 - val\_loss: 0.0050 - val\_accuracy: 0.9850 - 329s/epoch - 510ms/step

Epoch 47/100

Epoch 47: val\_accuracy did not improve from 0.98502

644/644 - 332s - loss: 0.0069 - accuracy: 0.9404 - val\_loss: 0.0048 - val\_accuracy: 0.9831 - 332s/epoch - 516ms/step

Epoch 48/100

Epoch 48: val\_accuracy did not improve from 0.98502

644/644 - 329s - loss: 0.0062 - accuracy: 0.9469 - val\_loss: 0.0059 - val\_accuracy: 0.9715 - 329s/epoch - 511ms/step

Epoch 49/100

Epoch 49: val\_accuracy did not improve from 0.98502

644/644 - 342s - loss: 0.0070 - accuracy: 0.9417 - val\_loss: 0.0063 - val\_accuracy: 0.9774 - 342s/epoch - 531ms/step

Epoch 50/100

Epoch 50: val\_accuracy did not improve from 0.98502

644/644 - 332s - loss: 0.0065 - accuracy: 0.9466 - val\_loss: 0.0051 - val\_accuracy: 0.9835 - 332s/epoch - 515ms/step

Epoch 51/100

Epoch 51: val\_accuracy did not improve from 0.98502

644/644 - 335s - loss: 0.0065 - accuracy: 0.9452 - val\_loss: 0.0049 - val\_accuracy: 0.9845 - 335s/epoch - 520ms/step

Epoch 52/100

Epoch 52: val\_accuracy improved from 0.98502 to 0.98674, saving model to dog\_class\_model.h5

644/644 - 333s - loss: 0.0065 - accuracy: 0.9504 - val\_loss: 0.0055 - val\_accuracy: 0.9867 - 333s/epoch - 517ms/step

Epoch 53/100

Epoch 53: val\_accuracy did not improve from 0.98674  
644/644 - 331s - loss: 0.0064 - accuracy: 0.9490 - val\_loss: 0.0063 -  
val\_accuracy: 0.9796 - 331s/epoch - 515ms/step  
Epoch 54/100

Epoch 54: val\_accuracy did not improve from 0.98674  
644/644 - 329s - loss: 0.0063 - accuracy: 0.9501 - val\_loss: 0.0058 -  
val\_accuracy: 0.9867 - 329s/epoch - 510ms/step  
Epoch 55/100

Epoch 55: val\_accuracy improved from 0.98674 to 0.98969, saving model  
to dog\_class\_model.h5  
644/644 - 330s - loss: 0.0060 - accuracy: 0.9527 - val\_loss: 0.0043 -  
val\_accuracy: 0.9897 - 330s/epoch - 512ms/step  
Epoch 56/100

Epoch 56: val\_accuracy did not improve from 0.98969  
644/644 - 329s - loss: 0.0062 - accuracy: 0.9516 - val\_loss: 0.0054 -  
val\_accuracy: 0.9811 - 329s/epoch - 511ms/step  
Epoch 57/100

Epoch 57: val\_accuracy did not improve from 0.98969  
644/644 - 327s - loss: 0.0061 - accuracy: 0.9529 - val\_loss: 0.0060 -  
val\_accuracy: 0.9850 - 327s/epoch - 508ms/step  
Epoch 58/100

Epoch 58: val\_accuracy did not improve from 0.98969  
644/644 - 327s - loss: 0.0066 - accuracy: 0.9529 - val\_loss: 0.0058 -  
val\_accuracy: 0.9804 - 327s/epoch - 508ms/step  
Epoch 59/100

Epoch 59: val\_accuracy did not improve from 0.98969  
644/644 - 325s - loss: 0.0059 - accuracy: 0.9571 - val\_loss: 0.0069 -  
val\_accuracy: 0.9745 - 325s/epoch - 505ms/step  
Epoch 60/100

Epoch 60: val\_accuracy did not improve from 0.98969  
644/644 - 325s - loss: 0.0057 - accuracy: 0.9580 - val\_loss: 0.0048 -  
val\_accuracy: 0.9838 - 325s/epoch - 505ms/step



Epoch 61/100

Epoch 61: val\_accuracy did not improve from 0.98969

644/644 - 325s - loss: 0.0056 - accuracy: 0.9573 - val\_loss: 0.0045 -  
val\_accuracy: 0.9885 - 325s/epoch - 505ms/step

Epoch 62/100

Epoch 62: val\_accuracy did not improve from 0.98969

644/644 - 320s - loss: 0.0059 - accuracy: 0.9585 - val\_loss: 0.0058 -  
val\_accuracy: 0.9823 - 320s/epoch - 498ms/step

Epoch 63/100

Epoch 63: val\_accuracy did not improve from 0.98969

644/644 - 323s - loss: 0.0056 - accuracy: 0.9598 - val\_loss: 0.0048 -  
val\_accuracy: 0.9889 - 323s/epoch - 501ms/step

Epoch 64/100

Epoch 64: val\_accuracy did not improve from 0.98969

644/644 - 321s - loss: 0.0057 - accuracy: 0.9584 - val\_loss: 0.0054 -  
val\_accuracy: 0.9880 - 321s/epoch - 498ms/step

Epoch 65/100

Epoch 65: val\_accuracy did not improve from 0.98969

644/644 - 322s - loss: 0.0056 - accuracy: 0.9583 - val\_loss: 0.0042 -  
val\_accuracy: 0.9885 - 322s/epoch - 500ms/step

Epoch 66/100

Epoch 66: val\_accuracy did not improve from 0.98969

644/644 - 314s - loss: 0.0057 - accuracy: 0.9623 - val\_loss: 0.0064 -  
val\_accuracy: 0.9892 - 314s/epoch - 488ms/step

Epoch 67/100

Epoch 67: val\_accuracy did not improve from 0.98969

644/644 - 318s - loss: 0.0052 - accuracy: 0.9648 - val\_loss: 0.0050 -  
val\_accuracy: 0.9882 - 318s/epoch - 494ms/step

Epoch 68/100

Epoch 68: val\_accuracy did not improve from 0.98969

644/644 - 319s - loss: 0.0060 - accuracy: 0.9569 - val\_loss: 0.0049 -  
val\_accuracy: 0.9875 - 319s/epoch - 495ms/step

Epoch 69/100

Epoch 69: val\_accuracy did not improve from 0.98969  
644/644 - 319s - loss: 0.0052 - accuracy: 0.9642 - val\_loss: 0.0054 -  
val\_accuracy: 0.9882 - 319s/epoch - 495ms/step  
Epoch 70/100

Epoch 70: val\_accuracy did not improve from 0.98969  
644/644 - 319s - loss: 0.0055 - accuracy: 0.9638 - val\_loss: 0.0045 -  
val\_accuracy: 0.9894 - 319s/epoch - 495ms/step  
Epoch 71/100

Epoch 71: val\_accuracy did not improve from 0.98969  
644/644 - 322s - loss: 0.0050 - accuracy: 0.9683 - val\_loss: 0.0050 -  
val\_accuracy: 0.9862 - 322s/epoch - 501ms/step  
Epoch 72/100

Epoch 72: val\_accuracy improved from 0.98969 to 0.99312, saving model  
to dog\_class\_model.h5  
644/644 - 321s - loss: 0.0052 - accuracy: 0.9630 - val\_loss: 0.0058 -  
val\_accuracy: 0.9931 - 321s/epoch - 498ms/step  
Epoch 73/100

Epoch 73: val\_accuracy did not improve from 0.99312  
644/644 - 321s - loss: 0.0054 - accuracy: 0.9626 - val\_loss: 0.0049 -  
val\_accuracy: 0.9904 - 321s/epoch - 498ms/step  
Epoch 74/100

Epoch 74: val\_accuracy did not improve from 0.99312  
644/644 - 321s - loss: 0.0050 - accuracy: 0.9660 - val\_loss: 0.0042 -  
val\_accuracy: 0.9919 - 321s/epoch - 498ms/step  
Epoch 75/100

Epoch 75: val\_accuracy did not improve from 0.99312  
644/644 - 325s - loss: 0.0048 - accuracy: 0.9679 - val\_loss: 0.0045 -  
val\_accuracy: 0.9921 - 325s/epoch - 504ms/step  
Epoch 76/100

Epoch 76: val\_accuracy did not improve from 0.99312  
644/644 - 322s - loss: 0.0052 - accuracy: 0.9668 - val\_loss: 0.0043 -  
val\_accuracy: 0.9889 - 322s/epoch - 500ms/step  
Epoch 77/100

Epoch 77: val\_accuracy did not improve from 0.99312  
644/644 - 322s - loss: 0.0053 - accuracy: 0.9661 - val\_loss: 0.0060 -  
val\_accuracy: 0.9818 - 322s/epoch - 500ms/step  
Epoch 78/100

Epoch 78: val\_accuracy did not improve from 0.99312  
644/644 - 319s - loss: 0.0053 - accuracy: 0.9669 - val\_loss: 0.0056 -  
val\_accuracy: 0.9872 - 319s/epoch - 495ms/step  
Epoch 79/100

Epoch 79: val\_accuracy did not improve from 0.99312  
644/644 - 321s - loss: 0.0048 - accuracy: 0.9683 - val\_loss: 0.0053 -  
val\_accuracy: 0.9907 - 321s/epoch - 498ms/step  
Epoch 80/100

Epoch 80: val\_accuracy did not improve from 0.99312  
644/644 - 322s - loss: 0.0050 - accuracy: 0.9683 - val\_loss: 0.0049 -  
val\_accuracy: 0.9887 - 322s/epoch - 501ms/step  
Epoch 81/100

Epoch 81: val\_accuracy did not improve from 0.99312  
644/644 - 321s - loss: 0.0051 - accuracy: 0.9675 - val\_loss: 0.0046 -  
val\_accuracy: 0.9921 - 321s/epoch - 498ms/step  
Epoch 82/100

Epoch 82: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0050 - accuracy: 0.9690 - val\_loss: 0.0054 -  
val\_accuracy: 0.9862 - 318s/epoch - 494ms/step  
Epoch 83/100

Epoch 83: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0049 - accuracy: 0.9696 - val\_loss: 0.0048 -  
val\_accuracy: 0.9907 - 318s/epoch - 494ms/step  
Epoch 84/100

Epoch 84: val\_accuracy did not improve from 0.99312  
644/644 - 319s - loss: 0.0052 - accuracy: 0.9669 - val\_loss: 0.0052 -  
val\_accuracy: 0.9924 - 319s/epoch - 496ms/step  
Epoch 85/100

Epoch 85: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0056 - accuracy: 0.9663 - val\_loss: 0.0040 -  
val\_accuracy: 0.9921 - 318s/epoch - 493ms/step  
Epoch 86/100

Epoch 86: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0046 - accuracy: 0.9706 - val\_loss: 0.0042 -  
val\_accuracy: 0.9914 - 318s/epoch - 495ms/step  
Epoch 87/100

Epoch 87: val\_accuracy did not improve from 0.99312  
644/644 - 320s - loss: 0.0048 - accuracy: 0.9698 - val\_loss: 0.0051 -  
val\_accuracy: 0.9904 - 320s/epoch - 496ms/step  
Epoch 88/100

Epoch 88: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0048 - accuracy: 0.9718 - val\_loss: 0.0045 -  
val\_accuracy: 0.9899 - 318s/epoch - 494ms/step  
Epoch 89/100

Epoch 89: val\_accuracy did not improve from 0.99312  
644/644 - 318s - loss: 0.0048 - accuracy: 0.9717 - val\_loss: 0.0046 -  
val\_accuracy: 0.9912 - 318s/epoch - 493ms/step  
Epoch 90/100

Epoch 90: val\_accuracy did not improve from 0.99312  
644/644 - 315s - loss: 0.0048 - accuracy: 0.9712 - val\_loss: 0.0046 -  
val\_accuracy: 0.9926 - 315s/epoch - 490ms/step  
Epoch 91/100

Epoch 91: val\_accuracy improved from 0.99312 to 0.99386, saving model  
to dog\_class\_model.h5  
644/644 - 327s - loss: 0.0044 - accuracy: 0.9761 - val\_loss: 0.0050 -  
val\_accuracy: 0.9939 - 327s/epoch - 507ms/step  
Epoch 92/100

Epoch 92: val\_accuracy did not improve from 0.99386  
644/644 - 320s - loss: 0.0052 - accuracy: 0.9693 - val\_loss: 0.0043 -  
val\_accuracy: 0.9917 - 320s/epoch - 497ms/step  
Epoch 93/100

Epoch 93: val\_accuracy did not improve from 0.99386  
644/644 - 321s - loss: 0.0045 - accuracy: 0.9729 - val\_loss: 0.0051 -  
val\_accuracy: 0.9914 - 321s/epoch - 498ms/step  
Epoch 94/100

Epoch 94: val\_accuracy did not improve from 0.99386  
644/644 - 318s - loss: 0.0042 - accuracy: 0.9768 - val\_loss: 0.0045 -  
val\_accuracy: 0.9934 - 318s/epoch - 495ms/step  
Epoch 95/100

Epoch 95: val\_accuracy did not improve from 0.99386  
644/644 - 317s - loss: 0.0048 - accuracy: 0.9727 - val\_loss: 0.0049 -  
val\_accuracy: 0.9934 - 317s/epoch - 493ms/step  
Epoch 96/100

Epoch 96: val\_accuracy did not improve from 0.99386  
644/644 - 317s - loss: 0.0044 - accuracy: 0.9737 - val\_loss: 0.0034 -  
val\_accuracy: 0.9926 - 317s/epoch - 492ms/step  
Epoch 97/100

Epoch 97: val\_accuracy did not improve from 0.99386  
644/644 - 316s - loss: 0.0048 - accuracy: 0.9709 - val\_loss: 0.0039 -  
val\_accuracy: 0.9934 - 316s/epoch - 491ms/step  
Epoch 98/100

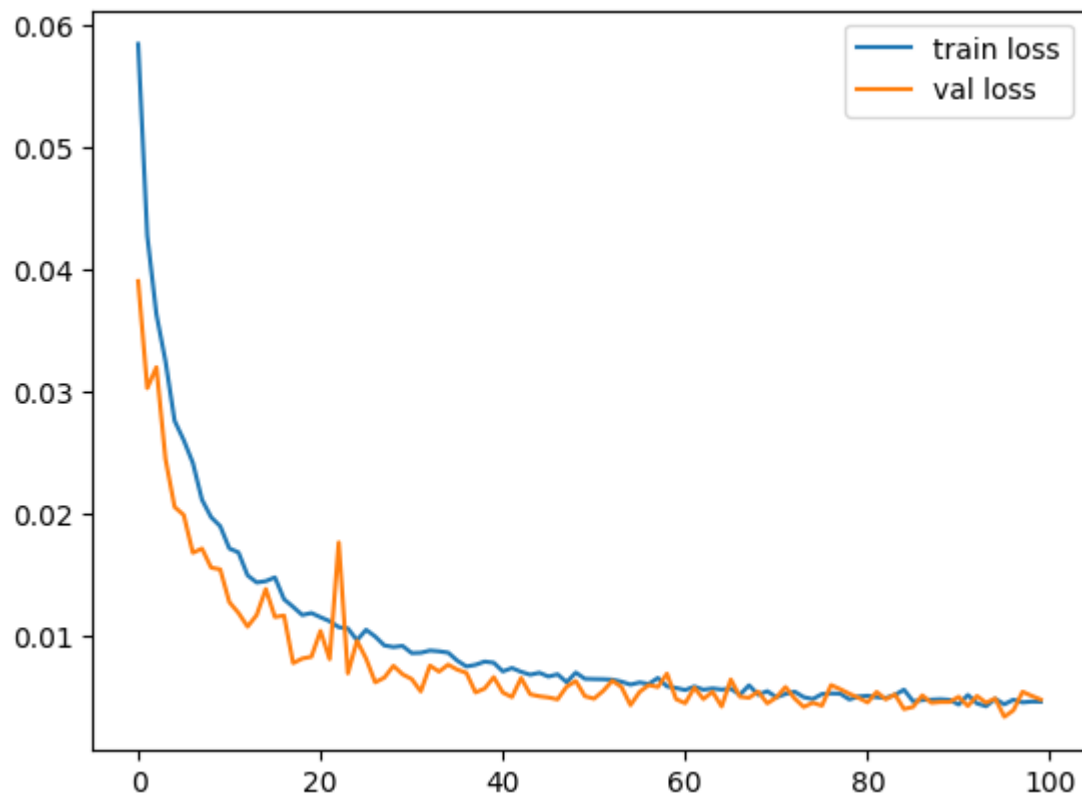
Epoch 98: val\_accuracy did not improve from 0.99386  
644/644 - 316s - loss: 0.0046 - accuracy: 0.9745 - val\_loss: 0.0054 -  
val\_accuracy: 0.9919 - 316s/epoch - 491ms/step  
Epoch 99/100

Epoch 99: val\_accuracy did not improve from 0.99386  
644/644 - 314s - loss: 0.0046 - accuracy: 0.9739 - val\_loss: 0.0051 -  
val\_accuracy: 0.9880 - 314s/epoch - 488ms/step  
Epoch 100/100

Epoch 100: val\_accuracy did not improve from 0.99386  
644/644 - 324s - loss: 0.0046 - accuracy: 0.9753 - val\_loss: 0.0048 -  
val\_accuracy: 0.9912 - 324s/epoch - 503ms/step

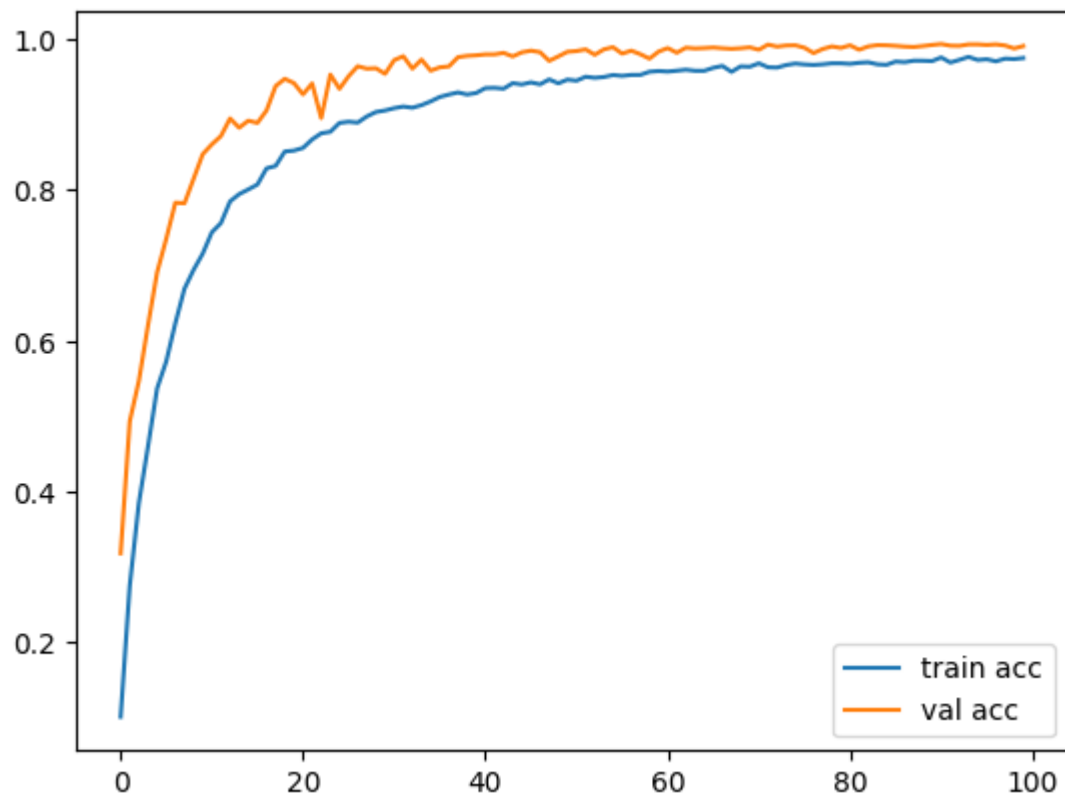
In [13]:

```
# plotting loss occurred  
plt.plot(r.history['loss'], label='train loss')  
plt.plot(r.history['val_loss'], label='val loss')  
plt.legend()  
plt.show()
```



In [14]:

```
# plotting the accuracy  
plt.plot(r.history['accuracy'], label='train acc')  
plt.plot(r.history['val_accuracy'], label='val acc')  
plt.legend()  
plt.show()
```



In [15]:

```
from keras.models import load_model
#loading the model and to check the structure
model1 = load_model('./dog_class_model.h5', compile=False)
model1.summary()
```



Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808

block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 120)	3010680

=====

Total params: 17,725,368

Trainable params: 3,010,680

Non-trainable params: 14,714,688

-----