

# Modern convnet architecture patterns

## Modularity, hierarchy, and reuse

## Residual connections

### Residual block where the number of filters changes

```
from tensorflow import keras
from tensorflow.keras import layers

inputs = keras.Input(shape=(32, 32, 3))
x = layers.Conv2D(32, 3, activation="relu")(inputs)
residual = x
x = layers.Conv2D(64, 3, activation="relu", padding="same")(x)
residual = layers.Conv2D(64, 1)(residual)
x = layers.add([x, residual])
```

Case where target block includes a max pooling layer

```
inputs = keras.Input(shape=(32, 32, 3))
x = layers.Conv2D(32, 3, activation="relu")(inputs)
residual = x
x = layers.Conv2D(64, 3, activation="relu", padding="same")(x)
x = layers.MaxPooling2D(2, padding="same")(x)
residual = layers.Conv2D(64, 1, strides=2)(residual)
x = layers.add([x, residual])
```

```
inputs = keras.Input(shape=(32, 32, 3))
x = layers.Rescaling(1./255)(inputs)

def residual_block(x, filters, pooling=False):
    residual = x
    x = layers.Conv2D(filters, 3, activation="relu", padding="same")(x)
    x = layers.Conv2D(filters, 3, activation="relu", padding="same")(x)
    if pooling:
        x = layers.MaxPooling2D(2, padding="same")(x)
        residual = layers.Conv2D(filters, 1, strides=2)(residual)
    elif filters != residual.shape[-1]:
        residual = layers.Conv2D(filters, 1)(residual)
    x = layers.add([x, residual])
    return x

x = residual_block(x, filters=32, pooling=True)
x = residual_block(x, filters=64, pooling=True)
x = residual_block(x, filters=128, pooling=False)

x = layers.GlobalAveragePooling2D()(x)
outputs = layers.Dense(1, activation="sigmoid")(x)
model = keras.Model(inputs=inputs, outputs=outputs)
model.summary()
```

Model: "model\_9"

| Layer (type)  | Output Shape        | Param # | Connected to                                   |
|---|---------------------|---------|--|
| input_30 (InputLayer)                               | [(None, 32, 32, 3)] | 0       | []   |
| rescaling_9 (Rescaling)                             | (None, 32, 32, 3)   | 0       | ['input_30[0][0]']                             |
| conv2d_141 (Conv2D)                                 | (None, 32, 32, 32)  | 896     | ['rescaling_9[0][0]']                          |
| conv2d_142 (Conv2D)                                 | (None, 32, 32, 32)  | 9248    | ['conv2d_141[0][0]']                           |
| max_pooling2d_28 (MaxPooling2D)                     | (None, 16, 16, 32)  | 0       | ['conv2d_142[0][0]']                           |
| conv2d_143 (Conv2D)                                 | (None, 16, 16, 32)  | 128     | ['rescaling_9[0][0]']                          |
| add_47 (Add)  | (None, 16, 16, 32)  | 0       | ['max_pooling2d_28[0][0]', 'conv2d_143[0][0]'] |
| conv2d_144 (Conv2D)                                 | (None, 16, 16, 64)  | 18496   | ['add_47[0][0]']                               |
| conv2d_145 (Conv2D)                                 | (None, 16, 16, 64)  | 36928   | ['conv2d_144[0][0]']                           |
| max_pooling2d_29 (MaxPooling2D)                     | (None, 8, 8, 64)    | 0       | ['conv2d_145[0][0]']                           |
| conv2d_146 (Conv2D)                                 | (None, 8, 8, 64)    | 2112    | ['add_47[0][0]']                               |
| add_48 (Add)  | (None, 8, 8, 64)    | 0       | ['max_pooling2d_29[0][0]', 'conv2d_146[0][0]'] |
| conv2d_147 (Conv2D)                                 | (None, 8, 8, 128)   | 73856   | ['add_48[0][0]']                               |
| conv2d_148 (Conv2D)                                 | (None, 8, 8, 128)   | 147584  | ['conv2d_147[0][0]']                           |
| conv2d_149 (Conv2D)                                 | (None, 8, 8, 128)   | 8320    | ['add_48[0][0]']                               |
| add_49 (Add)  | (None, 8, 8, 128)   | 0       | ['conv2d_148[0][0]', 'conv2d_149[0][0]']       |
| global_average_pooling2d_9 (GlobalAveragePooling2D) | (None, 128)         | 0       | ['add_49[0][0]']                               |
| dense_9 (Dense)                                     | (None, 1)           | 129     | ['global_average_pooling2d_9[0][0]']           |

=====  
Total params: 297,697  
Trainable params: 297,697  
Non-trainable params: 0  
=====

```
import os, shutil, pathlib
from tensorflow.keras.utils import image_dataset_from_directory

original_dir = pathlib.Path("./dogs-vs-cats/train")
new_base_dir = pathlib.Path("./dogs-vs-cats/cats_vs_dogs_small")

def make_subset(subset_name, start_index, end_index):
    for category in ("cat", "dog"):
        dir = new_base_dir / subset_name / category
        os.makedirs(dir)
        fnames = [f"{category}.{i}.jpg" for i in range(start_index, end_index)]
        for fname in fnames:
            shutil.copyfile(src=original_dir / fname,
                           dst=dir / fname)

#make_subset("train", start_index=0, end_index=1000)
#make_subset("validation", start_index=1000, end_index=1500)
#make_subset("test", start_index=1500, end_index=2500)

train_dataset = image_dataset_from_directory(
    new_base_dir / "train",
    image_size=(180, 180),
    batch_size=32)
validation_dataset = image_dataset_from_directory(
    new_base_dir / "validation",
    image_size=(180, 180),
    batch_size=32)
test_dataset = image_dataset_from_directory(
    new_base_dir / "test",
    image_size=(180, 180),
    batch_size=32)
```

Found 2000 files belonging to 2 classes.  
Found 1000 files belonging to 2 classes.  
Found 2000 files belonging to 2 classes.

```
data_augmentation = keras.Sequential(
    [
        layers.RandomFlip("horizontal"),
        layers.RandomRotation(0.1),
        layers.RandomZoom(0.2),
    ]
)
```

```
inputs = keras.Input(shape=(180, 180, 3))
x = data_augmentation(inputs)

x = layers.Rescaling(1./255)(x)
x = layers.Conv2D(filters=32, kernel_size=5, use_bias=False)(x)

for size in [32, 64, 128, 256, 512]:
    residual = x

    x = layers.BatchNormalization()(x)
    x = layers.Activation("relu")(x)
    x = layers.SeparableConv2D(size, 3, padding="same", use_bias=False)(x)

    x = layers.BatchNormalization()(x)
    x = layers.Activation("relu")(x)
    x = layers.SeparableConv2D(size, 3, padding="same", use_bias=False)(x)

    x = layers.MaxPooling2D(3, strides=2, padding="same")(x)

    residual = layers.Conv2D(
        size, 1, strides=2, padding="same", use_bias=False)(residual)
    x = layers.add([x, residual])

x = layers.GlobalAveragePooling2D()(x)
x = layers.Dropout(0.5)(x)
outputs = layers.Dense(1, activation="sigmoid")(x)
model = keras.Model(inputs=inputs, outputs=outputs)
```

```
model.compile(loss="binary_crossentropy",
              optimizer="rmsprop",
              metrics=["accuracy"])
history = model.fit(
    train_dataset,
    epochs=100,
    validation_data=validation_dataset)
```

Epoch 1/100

63/63 [=====] - 41s 537ms/step - loss: 0.7135 - accuracy: 0.5725 - val\_loss: 0.7073 - val\_accuracy: 0.5000

Epoch 2/100

63/63 [=====] - 27s 422ms/step - loss: 0.6593 - accuracy: 0.5925 - val\_loss: 0.6987 - val\_accuracy: 0.5000

Epoch 3/100

63/63 [=====] - 32s 499ms/step - loss: 0.6521 - accuracy: 0.6200 - val\_loss: 0.7032 - val\_accuracy: 0.5000

Epoch 4/100

63/63 [=====] - 32s 507ms/step - loss: 0.6189 - accuracy: 0.6655 - val\_loss: 0.6993 - val\_accuracy: 0.5000

Epoch 5/100

63/63 [=====] - 31s 496ms/step - loss: 0.5919 - accuracy: 0.6930 - val\_loss: 0.7272 - val\_accuracy: 0.5000

Epoch 6/100

63/63 [=====] - 32s 500ms/step - loss: 0.5745 - accuracy: 0.7025 - val\_loss: 0.7131 - val\_accuracy: 0.4990

Epoch 7/100

63/63 [=====] - 32s 506ms/step - loss: 0.5500 - accuracy: 0.7305 - val\_loss: 0.9001 - val\_accuracy: 0.5000

Epoch 8/100

63/63 [=====] - 31s 492ms/step - loss: 0.5496 - accuracy: 0.7400 - val\_loss: 0.7476 - val\_accuracy: 0.5140

Epoch 9/100

63/63 [=====] - 22s 353ms/step - loss: 0.5228 - accuracy: 0.7425 - val\_loss: 0.7280 - val\_accuracy: 0.5660

Epoch 10/100

63/63 [=====] - 31s 494ms/step - loss: 0.5138 - accuracy: 0.7505 - val\_loss: 0.8662 - val\_accuracy: 0.5830

Epoch 11/100

63/63 [=====] - 31s 493ms/step - loss: 0.4970 - accuracy: 0.7635 - val\_loss: 0.6105 - val\_accuracy: 0.6790

Epoch 12/100

63/63 [=====] - 32s 510ms/step - loss: 0.4962 - accuracy: 0.7545 - val\_loss: 0.6766 - val\_accuracy: 0.6260

Epoch 13/100

63/63 [=====] - 32s 508ms/step - loss: 0.4838 - accuracy: 0.7680 - val\_loss: 0.5809 - val\_accuracy: 0.6990

Epoch 14/100

63/63 [=====] - 32s 508ms/step - loss: 0.4709 - accuracy: 0.7810 - val\_loss: 1.3401 - val\_accuracy: 0.5340

Epoch 15/100

63/63 [=====] - 31s 491ms/step - loss: 0.4568 - accuracy: 0.7855 - val\_loss: 0.7847 - val\_accuracy: 0.6750

Epoch 16/100

63/63 [=====] - 32s 512ms/step - loss: 0.4315 - accuracy: 0.8130 - val\_loss: 0.5319 - val\_accuracy: 0.7680

Epoch 17/100

63/63 [=====] - 27s 433ms/step - loss: 0.4277 - accuracy: 0.8120 - val\_loss: 0.9704 - val\_accuracy: 0.6530

Epoch 18/100

63/63 [=====] - 31s 500ms/step - loss: 0.4118 - accuracy: 0.8200 - val\_loss: 1.1865 - val\_accuracy: 0.5350

Epoch 19/100

63/63 [=====] - 32s 501ms/step - loss: 0.4054 - accuracy: 0.8105 - val\_loss: 0.5281 - val\_accuracy: 0.7780

Epoch 20/100

63/63 [=====] - 32s 505ms/step - loss: 0.4111 - accuracy: 0.8090 - val\_loss: 0.8618 - val\_accuracy: 0.6230

Epoch 21/100

63/63 [=====] - 32s 500ms/step - loss: 0.3938 - accuracy: 0.8260 - val\_loss: 0.9996 - val\_accuracy: 0.5900

Epoch 22/100

63/63 [=====] - 32s 499ms/step - loss: 0.3814 - accuracy: 0.8335 - val\_loss: 0.6669 - val\_accuracy: 0.6810

Epoch 23/100

63/63 [=====] - 32s 509ms/step - loss: 0.3887 - accuracy: 0.8315 - val\_loss: 0.5620 - val\_accuracy: 0.7330

Epoch 24/100

63/63 [=====] - 32s 503ms/step - loss: 0.3706 - accuracy: 0.8450 - val\_loss: 0.4635 - val\_accuracy: 0.7900

Epoch 25/100

63/63 [=====] - 32s 507ms/step - loss: 0.3626 - accuracy: 0.8390 - val\_loss: 0.5070 - val\_accuracy: 0.7720

Epoch 26/100

63/63 [=====] - 32s 508ms/step - loss: 0.3682 - accuracy: 0.8365 - val\_loss: 0.5605 - val\_accuracy: 0.7470

Epoch 27/100  
63/63 [=====] - 31s 495ms/step - loss: 0.3540 - accuracy: 0.8445 - val\_loss: 1.8395 - val\_accuracy: 0.5800  
Epoch 28/100  
63/63 [=====] - 32s 500ms/step - loss: 0.3509 - accuracy: 0.8460 - val\_loss: 0.4632 - val\_accuracy: 0.7960  
Epoch 29/100  
63/63 [=====] - 31s 495ms/step - loss: 0.3335 - accuracy: 0.8620 - val\_loss: 0.5023 - val\_accuracy: 0.7800  
Epoch 30/100  
63/63 [=====] - 32s 506ms/step - loss: 0.3247 - accuracy: 0.8570 - val\_loss: 0.4386 - val\_accuracy: 0.8040  
Epoch 31/100  
63/63 [=====] - 31s 496ms/step - loss: 0.3302 - accuracy: 0.8485 - val\_loss: 0.4410 - val\_accuracy: 0.7970  
Epoch 32/100  
63/63 [=====] - 31s 498ms/step - loss: 0.3034 - accuracy: 0.8680 - val\_loss: 0.6504 - val\_accuracy: 0.7600  
Epoch 33/100  
63/63 [=====] - 31s 491ms/step - loss: 0.3127 - accuracy: 0.8665 - val\_loss: 0.5403 - val\_accuracy: 0.7880  
Epoch 34/100  
63/63 [=====] - 31s 498ms/step - loss: 0.3063 - accuracy: 0.8655 - val\_loss: 0.4322 - val\_accuracy: 0.8370  
Epoch 35/100  
63/63 [=====] - 32s 503ms/step - loss: 0.3030 - accuracy: 0.8740 - val\_loss: 0.4207 - val\_accuracy: 0.8030  
Epoch 36/100  
63/63 [=====] - 32s 505ms/step - loss: 0.3014 - accuracy: 0.8660 - val\_loss: 0.6295 - val\_accuracy: 0.7380  
Epoch 37/100  
63/63 [=====] - 32s 504ms/step - loss: 0.2926 - accuracy: 0.8740 - val\_loss: 0.3865 - val\_accuracy: 0.8410  
Epoch 38/100  
63/63 [=====] - 31s 495ms/step - loss: 0.2665 - accuracy: 0.8910 - val\_loss: 0.5948 - val\_accuracy: 0.7940  
Epoch 39/100  
63/63 [=====] - 28s 434ms/step - loss: 0.2872 - accuracy: 0.8840 - val\_loss: 0.5283 - val\_accuracy: 0.7760  
Epoch 40/100  
63/63 [=====] - 32s 506ms/step - loss: 0.2753 - accuracy: 0.8835 - val\_loss: 0.3631 - val\_accuracy: 0.8580  
Epoch 41/100  
63/63 [=====] - 32s 501ms/step - loss: 0.2753 - accuracy: 0.8895 - val\_loss: 0.6495 - val\_accuracy: 0.7250  
Epoch 42/100  
63/63 [=====] - 32s 507ms/step - loss: 0.2569 - accuracy: 0.8930 - val\_loss: 0.5856 - val\_accuracy: 0.7930  
Epoch 43/100  
63/63 [=====] - 32s 503ms/step - loss: 0.2579 - accuracy: 0.8945 - val\_loss: 0.5390 - val\_accuracy: 0.8120  
Epoch 44/100  
63/63 [=====] - 31s 495ms/step - loss: 0.2531 - accuracy: 0.8870 - val\_loss: 2.0241 - val\_accuracy: 0.5690  
Epoch 45/100  
63/63 [=====] - 32s 505ms/step - loss: 0.2366 - accuracy: 0.9065 - val\_loss: 0.5129 - val\_accuracy: 0.8080  
Epoch 46/100  
63/63 [=====] - 32s 503ms/step - loss: 0.2431 - accuracy: 0.8970 - val\_loss: 0.6172 - val\_accuracy: 0.7570  
Epoch 47/100  
63/63 [=====] - 31s 497ms/step - loss: 0.2467 - accuracy: 0.9000 - val\_loss: 0.6998 - val\_accuracy: 0.7630  
Epoch 48/100  
63/63 [=====] - 30s 482ms/step - loss: 0.2476 - accuracy: 0.8980 - val\_loss: 1.0030 - val\_accuracy: 0.7130  
Epoch 49/100  
63/63 [=====] - 31s 496ms/step - loss: 0.2207 - accuracy: 0.9055 - val\_loss: 0.4378 - val\_accuracy: 0.8350  
Epoch 50/100  
63/63 [=====] - 31s 496ms/step - loss: 0.2121 - accuracy: 0.9125 - val\_loss: 0.5769 - val\_accuracy: 0.7740  
  
Epoch 51/100  
63/63 [=====] - 32s 506ms/step - loss: 0.2444 - accuracy: 0.9005 - val\_loss: 0.3978 - val\_accuracy: 0.8430  
Epoch 52/100  
63/63 [=====] - 31s 499ms/step - loss: 0.2028 - accuracy: 0.9155 - val\_loss: 0.3784 - val\_accuracy: 0.8560  
Epoch 53/100  
63/63 [=====] - 31s 497ms/step - loss: 0.2154 - accuracy: 0.9025 - val\_loss: 0.6082 - val\_accuracy: 0.8210  
Epoch 54/100  
63/63 [=====] - 31s 495ms/step - loss: 0.2138 - accuracy: 0.9115 - val\_loss: 0.5440 - val\_accuracy: 0.8130  
Epoch 55/100  
63/63 [=====] - 32s 502ms/step - loss: 0.1916 - accuracy: 0.9150 - val\_loss: 0.6288 - val\_accuracy: 0.7980  
Epoch 56/100  
63/63 [=====] - 32s 499ms/step - loss: 0.2090 - accuracy: 0.9155 - val\_loss: 0.5030 - val\_accuracy: 0.8340  
Epoch 57/100  
63/63 [=====] - 31s 490ms/step - loss: 0.1922 - accuracy: 0.9170 - val\_loss: 0.4073 - val\_accuracy: 0.8510  
Epoch 58/100  
63/63 [=====] - 32s 505ms/step - loss: 0.1813 - accuracy: 0.9240 - val\_loss: 0.7630 - val\_accuracy: 0.7010  
Epoch 59/100  
63/63 [=====] - 32s 503ms/step - loss: 0.1976 - accuracy: 0.9160 - val\_loss: 0.3536 - val\_accuracy: 0.8720  
Epoch 60/100  
63/63 [=====] - 32s 501ms/step - loss: 0.1821 - accuracy: 0.9245 - val\_loss: 0.4454 - val\_accuracy: 0.8610  
Epoch 61/100  
63/63 [=====] - 32s 501ms/step - loss: 0.1818 - accuracy: 0.9265 - val\_loss: 0.3777 - val\_accuracy: 0.8670  
Epoch 62/100  
63/63 [=====] - 32s 511ms/step - loss: 0.1879 - accuracy: 0.9220 - val\_loss: 1.4254 - val\_accuracy: 0.6570  
Epoch 63/100  
63/63 [=====] - 32s 513ms/step - loss: 0.1918 - accuracy: 0.9280 - val\_loss: 0.4841 - val\_accuracy: 0.8360  
Epoch 64/100  
63/63 [=====] - 33s 528ms/step - loss: 0.1704 - accuracy: 0.9335 - val\_loss: 0.6832 - val\_accuracy: 0.8100  
Epoch 65/100  
63/63 [=====] - 34s 535ms/step - loss: 0.1691 - accuracy: 0.9295 - val\_loss: 0.7121 - val\_accuracy: 0.8210  
Epoch 66/100  
63/63 [=====] - 33s 527ms/step - loss: 0.1680 - accuracy: 0.9330 - val\_loss: 0.4298 - val\_accuracy: 0.8330  
Epoch 67/100  
63/63 [=====] - 33s 527ms/step - loss: 0.1514 - accuracy: 0.9410 - val\_loss: 0.3925 - val\_accuracy: 0.8730  
Epoch 68/100  
63/63 [=====] - 34s 536ms/step - loss: 0.1784 - accuracy: 0.9315 - val\_loss: 0.3346 - val\_accuracy: 0.8680  
Epoch 69/100  
63/63 [=====] - 34s 546ms/step - loss: 0.1643 - accuracy: 0.9360 - val\_loss: 0.6059 - val\_accuracy: 0.8160  
Epoch 70/100  
63/63 [=====] - 36s 568ms/step - loss: 0.1672 - accuracy: 0.9340 - val\_loss: 0.4374 - val\_accuracy: 0.8550  
Epoch 71/100  
63/63 [=====] - 33s 517ms/step - loss: 0.1370 - accuracy: 0.9450 - val\_loss: 0.4639 - val\_accuracy: 0.8520  
Epoch 72/100  
63/63 [=====] - 33s 516ms/step - loss: 0.1368 - accuracy: 0.9455 - val\_loss: 0.4796 - val\_accuracy: 0.8450  
Epoch 73/100  
63/63 [=====] - 33s 516ms/step - loss: 0.1584 - accuracy: 0.9350 - val\_loss: 0.4497 - val\_accuracy: 0.8300  
Epoch 74/100  
63/63 [=====] - 32s 509ms/step - loss: 0.1607 - accuracy: 0.9400 - val\_loss: 0.4988 - val\_accuracy: 0.8490  
Epoch 75/100  
63/63 [=====] - 32s 508ms/step - loss: 0.1515 - accuracy: 0.9340 - val\_loss: 0.4139 - val\_accuracy: 0.8610

Epoch 76/100  
63/63 [=====] - 33s 519ms/step - loss: 0.1350 - accuracy: 0.9470 - val\_loss: 0.3878 - val\_accuracy: 0.8370  
Epoch 77/100  
63/63 [=====] - 32s 513ms/step - loss: 0.1401 - accuracy: 0.9440 - val\_loss: 0.3743 - val\_accuracy: 0.8780  
Epoch 78/100  
63/63 [=====] - 34s 540ms/step - loss: 0.1336 - accuracy: 0.9525 - val\_loss: 0.3771 - val\_accuracy: 0.8770  
Epoch 79/100  
63/63 [=====] - 32s 500ms/step - loss: 0.1383 - accuracy: 0.9500 - val\_loss: 0.4525 - val\_accuracy: 0.8620  
Epoch 80/100  
63/63 [=====] - 32s 501ms/step - loss: 0.1322 - accuracy: 0.9485 - val\_loss: 0.5906 - val\_accuracy: 0.7820  
Epoch 81/100  
63/63 [=====] - 32s 505ms/step - loss: 0.1278 - accuracy: 0.9505 - val\_loss: 0.6811 - val\_accuracy: 0.8300  
Epoch 82/100  
63/63 [=====] - 31s 494ms/step - loss: 0.1281 - accuracy: 0.9505 - val\_loss: 0.4436 - val\_accuracy: 0.8650  
Epoch 83/100  
63/63 [=====] - 33s 520ms/step - loss: 0.1318 - accuracy: 0.9500 - val\_loss: 0.4134 - val\_accuracy: 0.8750  
Epoch 84/100  
63/63 [=====] - 32s 511ms/step - loss: 0.1148 - accuracy: 0.9520 - val\_loss: 0.3954 - val\_accuracy: 0.8800  
Epoch 85/100  
63/63 [=====] - 32s 514ms/step - loss: 0.1405 - accuracy: 0.9420 - val\_loss: 0.8298 - val\_accuracy: 0.8230  
Epoch 86/100  
63/63 [=====] - 30s 471ms/step - loss: 0.1312 - accuracy: 0.9490 - val\_loss: 0.3536 - val\_accuracy: 0.8820  
Epoch 87/100  
63/63 [=====] - 31s 492ms/step - loss: 0.1372 - accuracy: 0.9465 - val\_loss: 0.3339 - val\_accuracy: 0.8830  
Epoch 88/100  
63/63 [=====] - 33s 520ms/step - loss: 0.1157 - accuracy: 0.9535 - val\_loss: 0.3872 - val\_accuracy: 0.8680  
Epoch 89/100  
63/63 [=====] - 34s 537ms/step - loss: 0.1233 - accuracy: 0.9515 - val\_loss: 0.4416 - val\_accuracy: 0.8780  
Epoch 90/100  
63/63 [=====] - 32s 510ms/step - loss: 0.1163 - accuracy: 0.9515 - val\_loss: 0.4515 - val\_accuracy: 0.8730  
Epoch 91/100  
63/63 [=====] - 31s 490ms/step - loss: 0.1281 - accuracy: 0.9500 - val\_loss: 0.9718 - val\_accuracy: 0.7670  
Epoch 92/100  
63/63 [=====] - 33s 517ms/step - loss: 0.1120 - accuracy: 0.9605 - val\_loss: 0.5011 - val\_accuracy: 0.8460  
Epoch 93/100  
63/63 [=====] - 33s 521ms/step - loss: 0.1338 - accuracy: 0.9465 - val\_loss: 0.4105 - val\_accuracy: 0.8760  
Epoch 94/100  
63/63 [=====] - 32s 506ms/step - loss: 0.1290 - accuracy: 0.9445 - val\_loss: 0.5118 - val\_accuracy: 0.8490  
Epoch 95/100  
63/63 [=====] - 26s 418ms/step - loss: 0.0927 - accuracy: 0.9620 - val\_loss: 0.4702 - val\_accuracy: 0.8660  
Epoch 96/100  
63/63 [=====] - 32s 504ms/step - loss: 0.1117 - accuracy: 0.9610 - val\_loss: 0.5311 - val\_accuracy: 0.8690  
Epoch 97/100  
63/63 [=====] - 31s 498ms/step - loss: 0.0922 - accuracy: 0.9640 - val\_loss: 0.3964 - val\_accuracy: 0.8670  
Epoch 98/100  
63/63 [=====] - 31s 497ms/step - loss: 0.1088 - accuracy: 0.9580 - val\_loss: 0.5409 - val\_accuracy: 0.8420  
Epoch 99/100  
63/63 [=====] - 33s 518ms/step - loss: 0.1074 - accuracy: 0.9605 - val\_loss: 1.3448 - val\_accuracy: 0.7500  
Epoch 100/100  
63/63 [=====] - 33s 523ms/step - loss: 0.1021 - accuracy: 0.9600 - val\_loss: 0.3840 - val\_accuracy: 0.8810