```
from keras.models import Sequential
from keras.layers import Dense, Conv2D, MaxPooling2D, Flatten


model = Sequential()


model.add(Conv2D(64, (3, 3), activation='relu', input_shape=(64, 64, 3)))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(3, activation='softmax'))
```

```
/usr/local/lib/python3.11/dist-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pass an `input_shape`/`
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 64)	1,792
max_pooling2d (MaxPooling2D)	(None, 31, 31, 64)	0
flatten (Flatten)	(None, 61504)	0
dense (Dense)	(None, 128)	7,872,640
dense_1 (Dense)	(None, 3)	387

Total params: 7,874,819 (30.04 MB)
Trainable params: 7,874,819 (30.04 MB)
Non-trainable params: 0 (0.00 B)

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale =1./255,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   rotation_range=0.2,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   fill_mode='nearest',
                                   vertical_flip=True,
                                   horizontal_flip=True)
test_datagen = ImageDataGenerator(rescale=1./255)


Train_path='/content/drive/MyDrive/bts/Train'
Test_path='/content/drive/MyDrive/bts/Test'
Train_generator = train_datagen.flow_from_directory(Train_path,
                                                    target_size=(64, 64),
                                                    batch_size=32,
                                                    class_mode='categorical')
Test_generator = test_datagen.flow_from_directory(Test_path,
                                                  target_size=(64, 64),
                                                  batch_size=32,
                                                  class_mode='categorical')
```

```
Found 18 images belonging to 3 classes.
Found 12 images belonging to 3 classes.
```

```
Train_generator.class_indices
```

```
{'b': 0, 's': 1, 't': 2}
```

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


model.fit(Train_generator, epochs=100, validation_data=Test_generator)
```

```
/usr/local/lib/python3.11/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:121: UserWarning: Your `PyDataset`
  self._warn_if_super_not_called()
Epoch 1/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 7s 7s/step - accuracy: 0.3333 - loss: 1.1118 - val_accuracy: 0.3333 - val_loss: 6.5063
Epoch 2/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 384ms/step - accuracy: 0.3333 - loss: 5.5655 - val_accuracy: 0.4167 - val_loss: 9.2236
Epoch 3/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 377ms/step - accuracy: 0.4444 - loss: 7.6421 - val_accuracy: 0.3333 - val_loss: 13.2671
Epoch 4/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 392ms/step - accuracy: 0.3333 - loss: 10.3691 - val_accuracy: 0.3333 - val_loss: 8.4702
```

```
Epoch 5/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 374ms/step - accuracy: 0.3333 - loss: 6.0781 - val_accuracy: 0.4167 - val_loss: 1.5378
Epoch 6/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 346ms/step - accuracy: 0.7222 - loss: 0.8244 - val_accuracy: 0.4167 - val_loss: 3.2190
Epoch 7/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 360ms/step - accuracy: 0.5000 - loss: 3.0532 - val_accuracy: 0.4167 - val_loss: 4.0604
Epoch 8/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 357ms/step - accuracy: 0.5000 - loss: 3.7818 - val_accuracy: 0.3333 - val_loss: 3.4235
Epoch 9/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 357ms/step - accuracy: 0.6111 - loss: 3.2072 - val_accuracy: 0.4167 - val_loss: 2.8534
Epoch 10/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 375ms/step - accuracy: 0.5556 - loss: 2.6900 - val_accuracy: 0.2500 - val_loss: 2.9420
Epoch 11/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 362ms/step - accuracy: 0.4444 - loss: 2.1928 - val_accuracy: 0.3333 - val_loss: 3.0120
Epoch 12/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 359ms/step - accuracy: 0.3889 - loss: 1.8650 - val_accuracy: 0.3333 - val_loss: 2.5733
Epoch 13/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 379ms/step - accuracy: 0.4444 - loss: 1.5251 - val_accuracy: 0.3333 - val_loss: 1.9594
Epoch 14/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 365ms/step - accuracy: 0.4444 - loss: 1.1735 - val_accuracy: 0.4167 - val_loss: 1.4693
Epoch 15/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 382ms/step - accuracy: 0.5556 - loss: 0.8152 - val_accuracy: 0.4167 - val_loss: 1.3546
Epoch 16/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 369ms/step - accuracy: 0.7222 - loss: 0.6893 - val_accuracy: 0.4167 - val_loss: 1.5401
Epoch 17/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 365ms/step - accuracy: 0.5556 - loss: 0.7257 - val_accuracy: 0.4167 - val_loss: 1.6096
Epoch 18/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 384ms/step - accuracy: 0.6667 - loss: 0.7414 - val_accuracy: 0.4167 - val_loss: 1.5569
Epoch 19/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 353ms/step - accuracy: 0.6111 - loss: 0.7396 - val_accuracy: 0.5000 - val_loss: 1.4425
Epoch 20/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 372ms/step - accuracy: 0.6111 - loss: 0.8029 - val_accuracy: 0.5833 - val_loss: 1.3312
Epoch 21/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 394ms/step - accuracy: 0.6667 - loss: 0.7303 - val_accuracy: 0.5833 - val_loss: 1.2434
Epoch 22/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 372ms/step - accuracy: 0.7222 - loss: 0.7081 - val_accuracy: 0.5833 - val_loss: 1.1657
Epoch 23/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 643ms/step - accuracy: 0.7222 - loss: 0.6708 - val_accuracy: 0.5833 - val_loss: 1.1175
Epoch 24/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 361ms/step - accuracy: 0.6667 - loss: 0.6990 - val_accuracy: 0.5833 - val_loss: 1.1028
Epoch 25/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 378ms/step - accuracy: 0.8333 - loss: 0.5955 - val_accuracy: 0.5833 - val_loss: 1.1236
Epoch 26/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 368ms/step - accuracy: 0.8333 - loss: 0.6107 - val_accuracy: 0.6667 - val_loss: 1.1728
Epoch 27/100
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 356ms/step - accuracy: 0.7778 - loss: 0.6384 - val_accuracy: 0.5833 - val_loss: 1.2323
Epoch 28/100
```

```
model.save('b-s-t-classifier.h5')
```

```
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This file format is c
```

Start coding or generate with AI.