In [2]:

```
import tensorflow as tf
from tensorflow.keras.applications import EfficientNetB0
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, BatchNormalization
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.losses import CategoricalCrossentropy
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau

# Directories where your data is stored
train_dir = r'C:\Users\Abhishek\Desktop\New folder (2)\Paddy\train'
validation_dir = r'C:\Users\Abhishek\Desktop\New folder (2)\Paddy\valid'
test_dir = r'C:\Users\Abhishek\Desktop\New folder (2)\Paddy\test'

# Define constants
IMG_SIZE = 224
BATCH_SIZE = 8  # Reduced batch size
NUM_CLASSES = 3  # Number of classes in your dataset
EPOCHS = 100

# Generate batches of tensor image data with real-time data augmentation
datagen = ImageDataGenerator(
    rescale=1./255,
    horizontal_flip=True,
    vertical_flip=True)

train_generator = datagen.flow_from_directory(
    train_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical')

validation_generator = datagen.flow_from_directory(
    validation_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical')

test_generator = datagen.flow_from_directory(
    test_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical')

# Load base model
base_model = EfficientNetB0(weights='imagenet', include_top=False, input_shape=(IMG_

# Add a new top layer
x = base_model.output
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = Dense(512, activation='relu')(x)  # Reduced the number of neurons
x = Dropout(0.2)(x)  # Add dropout layer to reduce overfitting
x = BatchNormalization()(x)
predictions = Dense(NUM_CLASSES, activation='softmax')(x)

# This is the model we will train
model = tf.keras.models.Model(inputs=base_model.input, outputs=predictions)

# Freeze the base model
for layer in base_model.layers:
    layer.trainable = False
```

```
# Compile the model
model.compile(optimizer=Adam(lr=0.001), loss=CategoricalCrossentropy(), metrics=['ac

# Define callbacks
early_stopping = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights
reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.2, patience=5, min_lr=0.0

# Train the model
history = model.fit(
    train_generator,
    epochs=EPOCHS,
    validation_data=validation_generator,
    callbacks=[early_stopping, reduce_lr])

# Unfreeze the layers of the base model and fine-tune the entire model
for layer in base_model.layers:
    layer.trainable = True

# Recompile the model
model.compile(optimizer=Adam(lr=0.00001), loss=CategoricalCrossentropy(), metrics=['

# Continue training the model
history_fine_tuning = model.fit(
    train_generator,
    epochs=EPOCHS,
    validation_data=validation_generator,
    callbacks=[early_stopping, reduce_lr])

# Evaluate the model on the test data after fine-tuning
# Evaluate the model on the test data after fine-tuning
score = model.evaluate(test_generator)
print(f'Test loss: {score[0]} / Test accuracy: {score[1]}')
```

```
Found 911 images belonging to 3 classes.
Found 219 images belonging to 3 classes.
Found 164 images belonging to 3 classes.
WARNING:absl:`lr` is deprecated, please use `learning_rate` instead, or use the lega
cy optimizer, e.g.,tf.keras.optimizers.legacy.Adam.
Epoch 1/100
114/114 [==============================] - 269s 2s/step - loss: 1.3602 - accuracy:
0.3095 - val_loss: 1.2024 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 2/100
114/114 [==============================] - 213s 2s/step - loss: 1.2375 - accuracy:
0.3370 - val_loss: 1.0996 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 3/100
114/114 [==============================] - 229s 2s/step - loss: 1.2020 - accuracy:
0.3480 - val_loss: 1.1347 - val_accuracy: 0.3470 - lr: 0.0010
Epoch 4/100
114/114 [==============================] - 219s 2s/step - loss: 1.2005 - accuracy:
0.3326 - val_loss: 1.1348 - val_accuracy: 0.3470 - lr: 0.0010
Epoch 5/100
114/114 [==============================] - 215s 2s/step - loss: 1.1720 - accuracy:
0.3128 - val_loss: 1.1162 - val_accuracy: 0.3470 - lr: 0.0010
Epoch 6/100
114/114 [==============================] - 216s 2s/step - loss: 1.1523 - accuracy:
0.3249 - val_loss: 1.1018 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 7/100
114/114 [==============================] - 231s 2s/step - loss: 1.1393 - accuracy:
0.3074 - val_loss: 1.0996 - val_accuracy: 0.3014 - lr: 0.0010
Epoch 8/100
114/114 [==============================] - 292s 3s/step - loss: 1.1338 - accuracy:
0.3370 - val_loss: 1.0995 - val_accuracy: 0.3014 - lr: 2.0000e-04
Epoch 9/100
114/114 [==============================] - 252s 2s/step - loss: 1.1341 - accuracy:
0.3337 - val_loss: 1.1001 - val_accuracy: 0.3014 - lr: 2.0000e-04
Epoch 10/100
```

```
114/114 [==============================] - 216s 2s/step - loss: 1.1370 - accuracy:
0.3063 - val_loss: 1.0983 - val_accuracy: 0.3516 - lr: 2.0000e-04
Epoch 11/100
114/114 [==============================] - 235s 2s/step - loss: 1.1259 - accuracy:
0.3425 - val_loss: 1.1016 - val_accuracy: 0.3014 - lr: 2.0000e-04
Epoch 12/100
114/114 [==============================] - 189s 2s/step - loss: 1.1268 - accuracy:
0.3249 - val_loss: 1.0985 - val_accuracy: 0.3744 - lr: 2.0000e-04
Epoch 13/100
114/114 [==============================] - 208s 2s/step - loss: 1.1254 - accuracy:
0.3425 - val_loss: 1.0983 - val_accuracy: 0.4384 - lr: 2.0000e-04
Epoch 14/100
114/114 [==============================] - 210s 2s/step - loss: 1.1150 - accuracy:
0.3600 - val_loss: 1.0998 - val_accuracy: 0.3014 - lr: 2.0000e-04
Epoch 15/100
114/114 [==============================] - 190s 2s/step - loss: 1.1199 - accuracy:
0.3458 - val_loss: 1.0976 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 16/100
114/114 [==============================] - 179s 2s/step - loss: 1.1220 - accuracy:
0.3238 - val_loss: 1.0991 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 17/100
114/114 [==============================] - 186s 2s/step - loss: 1.1168 - accuracy:
0.3381 - val_loss: 1.1009 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 18/100
114/114 [==============================] - 190s 2s/step - loss: 1.1135 - accuracy:
0.3304 - val_loss: 1.0990 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 19/100
114/114 [==============================] - 148s 1s/step - loss: 1.1167 - accuracy:
0.3315 - val_loss: 1.0985 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 20/100
114/114 [==============================] - 157s 1s/step - loss: 1.1295 - accuracy:
0.2887 - val_loss: 1.1019 - val_accuracy: 0.3014 - lr: 2.0000e-04
Epoch 21/100
114/114 [==============================] - 146s 1s/step - loss: 1.1168 - accuracy:
0.3183 - val_loss: 1.0995 - val_accuracy: 0.3014 - lr: 4.0000e-05
Epoch 22/100
114/114 [==============================] - 191s 2s/step - loss: 1.1210 - accuracy:
0.3249 - val_loss: 1.0985 - val_accuracy: 0.3516 - lr: 4.0000e-05
Epoch 23/100
114/114 [==============================] - 186s 2s/step - loss: 1.1170 - accuracy:
0.3128 - val_loss: 1.0986 - val_accuracy: 0.3425 - lr: 4.0000e-05
Epoch 24/100
114/114 [==============================] - 191s 2s/step - loss: 1.1025 - accuracy:
0.3469 - val_loss: 1.0990 - val_accuracy: 0.3014 - lr: 4.0000e-05
Epoch 25/100
114/114 [==============================] - 193s 2s/step - loss: 1.1110 - accuracy:
0.3348 - val_loss: 1.0994 - val_accuracy: 0.3014 - lr: 4.0000e-05
WARNING:absl:`lr` is deprecated, please use `learning_rate` instead, or use the lega
cy optimizer, e.g.,tf.keras.optimizers.legacy.Adam.
Epoch 1/100
114/114 [==============================] - 494s 4s/step - loss: 0.5394 - accuracy:
0.8024 - val_loss: 1.5883 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 2/100
114/114 [==============================] - 395s 3s/step - loss: 0.3363 - accuracy:
0.8793 - val_loss: 2.4781 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 3/100
114/114 [==============================] - 399s 3s/step - loss: 0.3102 - accuracy:
0.8957 - val_loss: 1.7917 - val_accuracy: 0.3014 - lr: 0.0010
Epoch 4/100
114/114 [==============================] - 401s 4s/step - loss: 0.2050 - accuracy:
0.9341 - val_loss: 1.9226 - val_accuracy: 0.3516 - lr: 0.0010
Epoch 5/100
114/114 [==============================] - 418s 4s/step - loss: 0.1893 - accuracy:
0.9319 - val_loss: 1.7522 - val_accuracy: 0.2922 - lr: 0.0010
Epoch 6/100
114/114 [==============================] - 445s 4s/step - loss: 0.1805 - accuracy:
0.9429 - val_loss: 2.8488 - val_accuracy: 0.3744 - lr: 0.0010
Epoch 7/100
114/114 [==============================] - 438s 4s/step - loss: 0.1027 - accuracy:
```

```
0.9638 - val_loss: 2.5925 - val_accuracy: 0.4521 - lr: 2.0000e-04
Epoch 8/100
114/114 [==============================] - 364s 3s/step - loss: 0.0640 - accuracy:
0.9769 - val_loss: 0.9765 - val_accuracy: 0.6256 - lr: 2.0000e-04
Epoch 9/100
114/114 [==============================] - 359s 3s/step - loss: 0.0295 - accuracy:
0.9956 - val_loss: 1.2251 - val_accuracy: 0.6393 - lr: 2.0000e-04
Epoch 10/100
114/114 [==============================] - 459s 4s/step - loss: 0.0303 - accuracy:
0.9934 - val_loss: 0.7888 - val_accuracy: 0.5845 - lr: 2.0000e-04
Epoch 11/100
114/114 [==============================] - 248s 2s/step - loss: 0.0413 - accuracy:
0.9901 - val_loss: 1.6642 - val_accuracy: 0.3927 - lr: 2.0000e-04
Epoch 12/100
114/114 [==============================] - 237s 2s/step - loss: 0.0436 - accuracy:
0.9868 - val_loss: 0.3365 - val_accuracy: 0.8721 - lr: 2.0000e-04
Epoch 13/100
114/114 [==============================] - 241s 2s/step - loss: 0.0199 - accuracy:
0.9945 - val_loss: 4.9019 - val_accuracy: 0.2968 - lr: 2.0000e-04
Epoch 14/100
114/114 [==============================] - 303s 3s/step - loss: 0.0187 - accuracy:
0.9967 - val_loss: 13.2789 - val_accuracy: 0.3470 - lr: 2.0000e-04
Epoch 15/100
114/114 [==============================] - 310s 3s/step - loss: 0.0152 - accuracy:
0.9967 - val_loss: 1.5674 - val_accuracy: 0.6301 - lr: 2.0000e-04
Epoch 16/100
114/114 [==============================] - 312s 3s/step - loss: 0.0188 - accuracy:
0.9956 - val_loss: 0.8165 - val_accuracy: 0.8082 - lr: 2.0000e-04
Epoch 17/100
114/114 [==============================] - 372s 3s/step - loss: 0.0175 - accuracy:
0.9956 - val_loss: 1.7096 - val_accuracy: 0.5662 - lr: 2.0000e-04
Epoch 18/100
114/114 [==============================] - 391s 3s/step - loss: 0.0225 - accuracy:
0.9956 - val_loss: 0.2273 - val_accuracy: 0.9132 - lr: 4.0000e-05
Epoch 19/100
114/114 [==============================] - 393s 3s/step - loss: 0.0080 - accuracy:
0.9989 - val_loss: 0.0294 - val_accuracy: 1.0000 - lr: 4.0000e-05
Epoch 20/100
114/114 [==============================] - 266s 2s/step - loss: 0.0132 - accuracy:
0.9967 - val_loss: 0.0347 - val_accuracy: 0.9954 - lr: 4.0000e-05
Epoch 21/100
114/114 [==============================] - 242s 2s/step - loss: 0.0194 - accuracy:
0.9912 - val_loss: 0.0858 - val_accuracy: 0.9635 - lr: 4.0000e-05
Epoch 22/100
114/114 [==============================] - 245s 2s/step - loss: 0.0140 - accuracy:
0.9945 - val_loss: 0.1972 - val_accuracy: 0.9178 - lr: 4.0000e-05
Epoch 23/100
114/114 [==============================] - 239s 2s/step - loss: 0.0166 - accuracy:
0.9978 - val_loss: 0.8265 - val_accuracy: 0.7671 - lr: 4.0000e-05
Epoch 24/100
114/114 [==============================] - 162s 1s/step - loss: 0.0050 - accuracy:
1.0000 - val_loss: 0.0832 - val_accuracy: 0.9817 - lr: 4.0000e-05
Epoch 25/100
114/114 [==============================] - 163s 1s/step - loss: 0.0141 - accuracy:
0.9967 - val_loss: 0.0413 - val_accuracy: 0.9954 - lr: 1.0000e-05
Epoch 26/100
114/114 [==============================] - 162s 1s/step - loss: 0.0113 - accuracy:
0.9978 - val_loss: 0.0340 - val_accuracy: 0.9909 - lr: 1.0000e-05
Epoch 27/100
114/114 [==============================] - 163s 1s/step - loss: 0.0054 - accuracy:
1.0000 - val_loss: 0.0386 - val_accuracy: 0.9863 - lr: 1.0000e-05
Epoch 28/100
114/114 [==============================] - 163s 1s/step - loss: 0.0124 - accuracy:
0.9978 - val_loss: 0.0267 - val_accuracy: 0.9909 - lr: 1.0000e-05
Epoch 29/100
114/114 [==============================] - 164s 1s/step - loss: 0.0099 - accuracy:
0.9978 - val_loss: 0.0481 - val_accuracy: 0.9817 - lr: 1.0000e-05
Epoch 30/100
114/114 [==============================] - 163s 1s/step - loss: 0.0049 - accuracy:
```

```
1.0000 - val_loss: 0.0381 - val_accuracy: 0.9863 - lr: 1.0000e-05
Epoch 31/100
114/114 [==============================] - 164s 1s/step - loss: 0.0038 - accuracy:
1.0000 - val_loss: 0.0193 - val_accuracy: 0.9954 - lr: 1.0000e-05
Epoch 32/100
114/114 [==============================] - 166s 1s/step - loss: 0.0117 - accuracy:
0.9945 - val_loss: 0.0574 - val_accuracy: 0.9726 - lr: 1.0000e-05
Epoch 33/100
114/114 [==============================] - 166s 1s/step - loss: 0.0450 - accuracy:
0.9934 - val_loss: 0.0408 - val_accuracy: 0.9817 - lr: 1.0000e-05
Epoch 34/100
114/114 [==============================] - 165s 1s/step - loss: 0.0214 - accuracy:
0.9934 - val_loss: 0.0577 - val_accuracy: 0.9772 - lr: 1.0000e-05
Epoch 35/100
114/114 [==============================] - 165s 1s/step - loss: 0.0145 - accuracy:
0.9945 - val_loss: 0.0213 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 36/100
114/114 [==============================] - 165s 1s/step - loss: 0.0078 - accuracy:
0.9989 - val_loss: 0.0178 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 37/100
114/114 [==============================] - 164s 1s/step - loss: 0.0043 - accuracy:
1.0000 - val_loss: 0.0172 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 38/100
114/114 [==============================] - 164s 1s/step - loss: 0.0071 - accuracy:
0.9978 - val_loss: 0.0217 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 39/100
114/114 [==============================] - 161s 1s/step - loss: 0.0095 - accuracy:
0.9978 - val_loss: 0.0914 - val_accuracy: 0.9680 - lr: 1.0000e-05
Epoch 40/100
114/114 [==============================] - 145s 1s/step - loss: 0.0062 - accuracy:
0.9989 - val_loss: 0.0417 - val_accuracy: 0.9817 - lr: 1.0000e-05
Epoch 41/100
114/114 [==============================] - 132s 1s/step - loss: 0.0047 - accuracy:
1.0000 - val_loss: 0.0144 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 42/100
114/114 [==============================] - 134s 1s/step - loss: 0.0032 - accuracy:
1.0000 - val_loss: 0.0212 - val_accuracy: 1.0000 - lr: 1.0000e-05
Epoch 43/100
114/114 [==============================] - 135s 1s/step - loss: 0.0068 - accuracy:
0.9978 - val_loss: 0.0287 - val_accuracy: 0.9954 - lr: 1.0000e-05
Epoch 44/100
114/114 [==============================] - 137s 1s/step - loss: 0.0076 - accuracy:
0.9989 - val_loss: 0.0537 - val_accuracy: 0.9817 - lr: 1.0000e-05
Epoch 45/100
114/114 [==============================] - 134s 1s/step - loss: 0.0092 - accuracy:
0.9967 - val_loss: 0.0725 - val_accuracy: 0.9817 - lr: 1.0000e-05
Epoch 46/100
114/114 [==============================] - 136s 1s/step - loss: 0.0050 - accuracy:
1.0000 - val_loss: 0.0373 - val_accuracy: 0.9863 - lr: 1.0000e-05
Epoch 47/100
114/114 [==============================] - 138s 1s/step - loss: 0.0162 - accuracy:
0.9956 - val_loss: 0.0264 - val_accuracy: 0.9954 - lr: 1.0000e-05
Epoch 48/100
114/114 [==============================] - 132s 1s/step - loss: 0.0115 - accuracy:
0.9978 - val_loss: 0.0908 - val_accuracy: 0.9680 - lr: 1.0000e-05
Epoch 49/100
114/114 [==============================] - 133s 1s/step - loss: 0.0033 - accuracy:
1.0000 - val_loss: 0.0688 - val_accuracy: 0.9863 - lr: 1.0000e-05
Epoch 50/100
114/114 [==============================] - 133s 1s/step - loss: 0.0126 - accuracy:
0.9989 - val_loss: 0.0907 - val_accuracy: 0.9726 - lr: 1.0000e-05
Epoch 51/100
114/114 [==============================] - 133s 1s/step - loss: 0.0077 - accuracy:
0.9978 - val_loss: 0.0381 - val_accuracy: 0.9954 - lr: 1.0000e-05
21/21 [==============================] - 8s 379ms/step - loss: 0.0769 - accuracy: 0.
9756
Test loss: 0.07693624496459961 / Test accuracy: 0.9756097793579102
```

In [ ]: