In [ ]:

```python
import pandas as pd
import numpy as np
import os
import shutil
import random
from tqdm import tqdm
from pathlib import Path
import cv2 as cv
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras import layers, optimizers, losses, metrics, callbacks
from tensorflow.keras import Sequential, Model, Input
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from sklearn.model_selection import train_test_split
import warnings
warnings.filterwarnings("ignore")
random.seed(45)

print(tf.__version__)
```

```
2.19.0
```

In [ ]:

```python
import glob
from PIL import  Image, ImageFile
from joblib  import Parallel, delayed
ImageFile.LOAD_TRUNCATED_IMAGES = True
```

LOAD IMAGE DATA AND UNDERSTANDING SOME PROPERTIES OF IMAGE

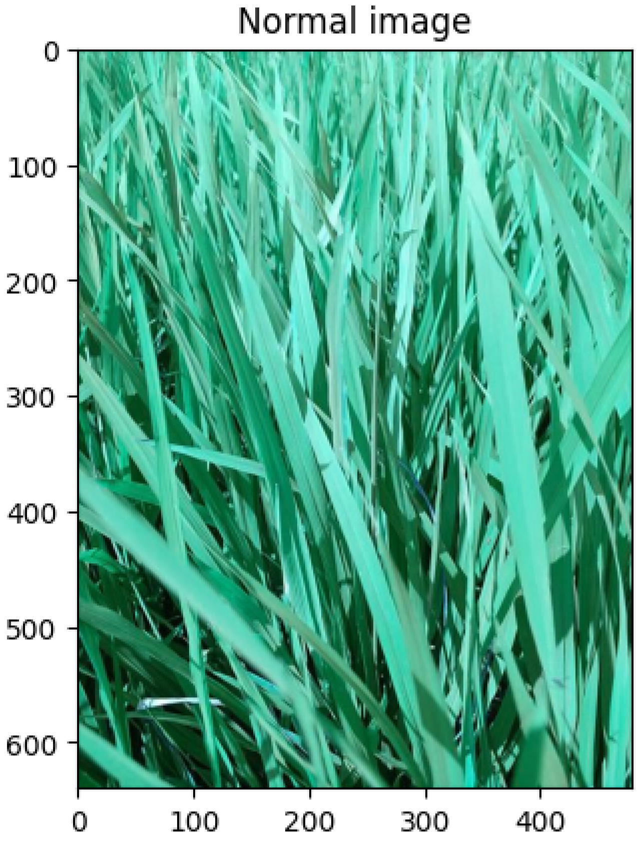
In [3]:

```python
def image_properties(path):
    for img in random.sample(os.listdir(path),1):
        print('Image name =',img)
        image = cv.imread(os.path.join(path, img),cv.IMREAD_COLOR)
        break

    return image
```

In [4]:

```python
HOME_PATH = os.getcwd() + "/"

path = HOME_PATH + 'train_images/normal'

image = image_properties(path)
plt.imshow(image)
plt.title('Normal image')
print(f"The dimensions are {image.shape[0]} pixels height and {image.shape[1]} pixe
print(f"The maximum pixel value is {image.max():.2f}")
print(f"The minimum pixel value is {image.min():.2f}")
print(f"The mean value of the pixels is {image.mean():.2f}")
print(f"The standard deviation is {image.std():.2f}")
```

```
Image name = 103292.jpg
The dimensions are 640 pixels height and 480 pixels width
The maximum pixel value is 255.00
The minimum pixel value is 0.00
The mean value of the pixels is 120.33
The standard deviation is 69.71
```



Loading a dataset

In [5]:

```
batch_size = 64
img_height = 256
img_width =  256

data_dir = HOME_PATH + 'train_images'
```

Image Data-Generator

Data Normalization And Data Augmentation

In [6]:

```
img_datagen = tf.keras.preprocessing.image.ImageDataGenerator(
                            rescale=1.0/255.0,
                            validation_split=0.3,
                            rotation_range=5,
                            shear_range=0.3,
                            zoom_range=0.3,
                            width_shift_range=0.05,
                            height_shift_range=0.05,
                            horizontal_flip=True,
                            vertical_flip=True
                )
```

Training Dataset (70%)

In [7]:

```
train_gen = img_datagen.flow_from_directory(
                            data_dir,
                            subset="training",
                            seed=42,
                            target_size=(img_height, img_width),
                            batch_size=batch_size,
                            class_mode="categorical",
                            color_mode='rgb'
            )
```

```
Found 7288 images belonging to 10 classes.
```

Testing Dataset (30%)

In [8]:

```
valid_gen = img_datagen.flow_from_directory(
                            data_dir,
                            subset="validation",
                            seed=42,
                            target_size=(img_height, img_width),
                            batch_size=batch_size,
                            class_mode="categorical"
            )
```

```
Found 3119 images belonging to 10 classes.
```

Class Lables

In [9]:

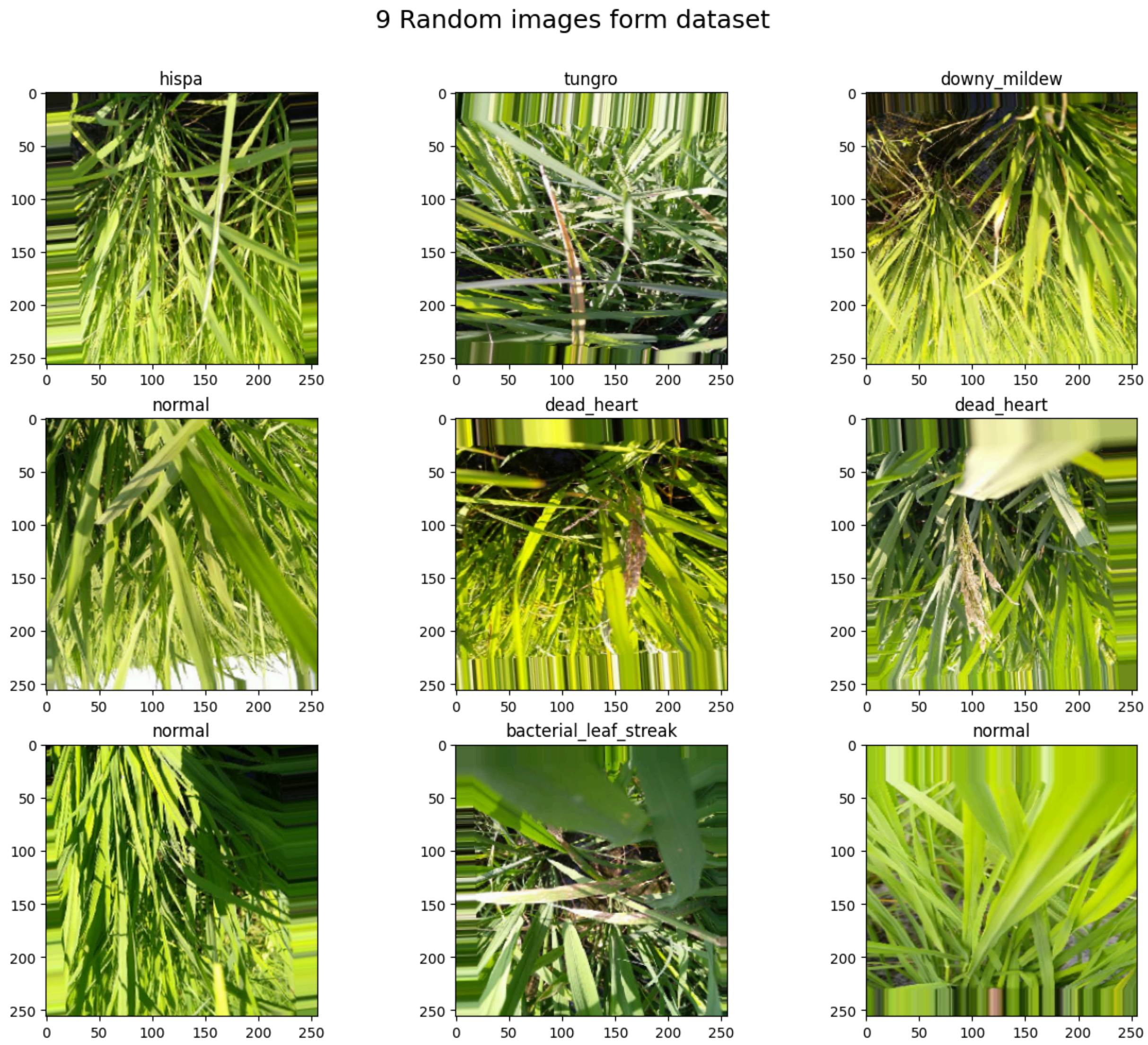
```
print('Total No Of Classes in the datasetL:',len(train_gen.class_indices))
print('Class Names:',train_gen.class_indices)
```

```
Total No Of Classes in the datasetL: 10
Class Names: {'bacterial_leaf_blight': 0, 'bacterial_leaf_streak': 1, 'bacterial_pan
icle_blight': 2, 'blast': 3, 'brown_spot': 4, 'dead_heart': 5, 'downy_mildew': 6, 'h
ispa': 7, 'normal': 8, 'tungro': 9}
```

Displaying a 9 Random images form dataset

In [10]:

```
fig, axs = plt.subplots(nrows=3, ncols=3, figsize=(15, 12))
plt.subplots_adjust(hspace=0.2)
fig.suptitle("9 Random images form dataset", fontsize=18, y=0.95)

for i in range(1,10):
    plt.subplot(3,3,i)
    img, label = train_gen.__next__()
    plt.title(list(train_gen.class_indices.keys())[np.argmax(label)])
    plt.imshow(img[0])
```



Vision Transformer (ViT)

REF:

`https://keras.io/examples/vision/image_classification_with_visic`

In [11]:

```
# Loads an image and convert them to numpy array
def load_images(paths):
  data = []
  labels = []
  i = 0
  for label, path in tqdm(enumerate(paths)):
    for img_path in os.listdir(path):
      image = np.array(Image.open(os.path.join(path,img_path)).convert('RGB').resiz
      data.append(image)
      labels.append(label)

  return np.array(data), np.asarray(labels)
```

In [12]:

```
images, labels = load_images(glob.glob(HOME_PATH + 'train_images/*'))
```

```
10it [00:52,  5.23s/it]
```

Train Test Split (70:30)

In [13]:

```
X_train, X_test, y_train, y_test = train_test_split(images, labels.reshape(-1,1), t
```

In [14]:

```
num_classes = 10
input_shape = (256, 256, 3)
```

Configure the hyperparameters

In [15]:

```
learning_rate = 0.001
weight_decay = 0.0001
batch_size = 32
num_epochs = 100
image_size = 72
patch_size = 6
num_patches = (image_size // patch_size) ** 2
projection_dim = 64
num_heads = 4
transformer_units = [
    projection_dim * 2,
    projection_dim,
]  # Size of the transformer layers
transformer_layers = 8
mlp_head_units = [2048, 1024]  # Size of the dense layers of the final classifier
```

Data Augmentation

In [16]:

```
data_augmentation = keras.Sequential(
    [
        layers.Normalization(),
        layers.Resizing(image_size, image_size),
        layers.RandomFlip("horizontal"),
        layers.RandomRotation(factor=0.02),
        layers.RandomZoom(
            height_factor=0.2, width_factor=0.2
        ),
    ],
    name="data_augmentation",
)
# Compute the mean and the variance of the training data for normalization.
data_augmentation.layers[0].adapt(X_train)
```

Multilayer perceptron (MLP)

In [17]:

```
def mlp(x, hidden_units, dropout_rate):
    for units in hidden_units:
        x = layers.Dense(units, activation=tf.nn.gelu)(x)
        x = layers.Dropout(dropout_rate)(x)
    return x
```

Implementing patch creation as a layer

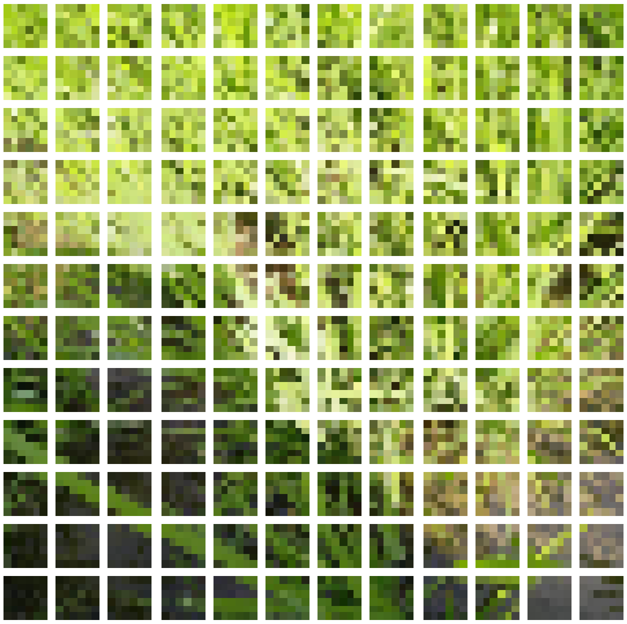
In [18]:

```
class Patches(layers.Layer):
    def __init__(self, patch_size):
        super(Patches, self).__init__()
        self.patch_size = patch_size

    def call(self, images):
        batch_size = tf.shape(images)[0]
        patches = tf.image.extract_patches(
            images=images,
            sizes=[1, self.patch_size, self.patch_size, 1],
            strides=[1, self.patch_size, self.patch_size, 1],
            rates=[1, 1, 1, 1],
            padding="VALID",
        )
        patch_dims = patches.shape[-1]
        patches = tf.reshape(patches, [batch_size, -1, patch_dims])
        return patches
```

In [19]:

```
plt.figure(figsize=(4, 4))
image = X_train[np.random.choice(range(X_train.shape[0]))]
plt.imshow(image.astype("uint8"))
plt.axis("off")

resized_image = tf.image.resize(
    tf.convert_to_tensor([image]), size=(image_size, image_size)
)
patches = Patches(patch_size)(resized_image)
print(f"Image size: {image_size} X {image_size}")
print(f"Patch size: {patch_size} X {patch_size}")
print(f"Patches per image: {patches.shape[1]}")
print(f"Elements per patch: {patches.shape[-1]}")

n = int(np.sqrt(patches.shape[1]))
plt.figure(figsize=(4, 4))
for i, patch in enumerate(patches[0]):
    ax = plt.subplot(n, n, i + 1)
    patch_img = tf.reshape(patch, (patch_size, patch_size, 3))
    plt.imshow(patch_img.numpy().astype("uint8"))
    plt.axis("off")
```

```
Image size: 72 X 72
Patch size: 6 X 6
Patches per image: 144
Elements per patch: 108
```



Implementing the patch encoding layer

In [20]:

```python
class PatchEncoder(layers.Layer):
    def __init__(self, num_patches, projection_dim):
        super(PatchEncoder, self).__init__()
        self.num_patches = num_patches
        self.projection = layers.Dense(units=projection_dim)
        self.position_embedding = layers.Embedding(
            input_dim=num_patches, output_dim=projection_dim
        )

    def call(self, patch):
        positions = tf.range(start=0, limit=self.num_patches, delta=1)
        encoded = self.projection(patch) + self.position_embedding(positions)
        return encoded
```

ViT Model

In [21]:

```python
def create_vit_classifier():

    inputs = layers.Input(shape=input_shape)
    # Augment data.
    augmented = data_augmentation(inputs)
    # Create patches.
    patches = Patches(patch_size)(augmented)
    # Encode patches.
    encoded_patches = PatchEncoder(num_patches, projection_dim)(patches)

    # Create multiple layers of the Transformer block.
    for _ in range(transformer_layers):
        # Layer normalization 1.
        x1 = layers.LayerNormalization(epsilon=1e-6)(encoded_patches)
        # Create a multi-head attention layer.
        attention_output = layers.MultiHeadAttention(
            num_heads=num_heads, key_dim=projection_dim, dropout=0.1
        )(x1, x1)
        # Skip connection 1.
        x2 = layers.Add()([attention_output, encoded_patches])
        # Layer normalization 2.
        x3 = layers.LayerNormalization(epsilon=1e-6)(x2)
        # MLP.
        x3 = mlp(x3, hidden_units=transformer_units, dropout_rate=0.1)
        # Skip connection 2.
        encoded_patches = layers.Add()([x3, x2])

    # Create a [batch_size, projection_dim] tensor.
    representation = layers.LayerNormalization(epsilon=1e-6)(encoded_patches)
    representation = layers.Flatten()(representation)
    representation = layers.Dropout(0.5)(representation)
    # Add MLP.
    features = mlp(representation, hidden_units=mlp_head_units, dropout_rate=0.5)
    # Classify outputs.
    logits = layers.Dense(num_classes)(features)
    # Create the Keras model.
    model = keras.Model(inputs=inputs, outputs=logits)
    return model
```

Callback

In [27]:

```python
filepath = HOME_PATH + 'paddy_models/model_vgg_new.keras'

checkpoint = tf.keras.callbacks.ModelCheckpoint(
    filepath = filepath,
    monitor="val_accuracy",
    verbose=1,
    save_best_only=True,
    mode = 'auto'
)
```

In [28]:

```
class TerminateNaN(tf.keras.callbacks.Callback):
  def on_epoch_end(self, epoch, logs={}):
    loss = logs.get('loss')
    if loss is not None:
      if np.isnan(loss) or np.isinf(loss):
        print('Invalid loss and terminated at loss {}'.format(epoch))
        self.model.stop_training = True

terminate_nan = TerminateNaN()
```

In [29]:

```
# Learning rate scheduler
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(
    monitor="val_accuracy",
    factor=0.1,
    patience=10,
    verbose=1,
    mode="auto",
    min_delta=0.001,
    cooldown=3,
    min_lr=0
)
```

In [30]:

```
def train(model):
    optimizer = keras.optimizers.AdamW(
        learning_rate=learning_rate, weight_decay=weight_decay
    )

    model.compile(
        optimizer=optimizer,
        loss=keras.losses.SparseCategoricalCrossentropy(from_logits=True),
        metrics=[
            keras.metrics.SparseCategoricalAccuracy(name="accuracy"),
        ]
    )

    history = model.fit(
        x=X_train,
        y=y_train,
        batch_size=batch_size,
        epochs=num_epochs,
        validation_split=0.3,
        callbacks=[checkpoint, reduce_lr, terminate_nan]
    )

    _, accuracy = model.evaluate(X_test, y_test)
    print(f"Test accuracy: {round(accuracy * 100, 2)}%")

    return history


vit_classifier = create_vit_classifier()
vit_classifier.summary()
```

```
Model: "functional_2"
```

```
┏━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━┓
┃ Layer (type)        ┃ Output Shape      ┃    Param # ┃ Connected to      ┃
┡━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━┩
│ input_layer_2       │ (None, 256, 256,  │          0 │ -                 │
│ (InputLayer)        │ 3)                │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ data_augmentation   │ (None, 72, 72, 3) │          7 │ input_layer_2[0]… │
│ (Sequential)        │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ patches_2 (Patches) │ (None, None, 108) │          0 │ data_augmentatio… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ patch_encoder_1     │ (None, 144, 64)   │     16,192 │ patches_2[0][0]   │
│ (PatchEncoder)      │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ patch_encoder_1[… │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_16 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ patch_encoder_1[… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_16[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_21 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_28          │ (None, 144, 128)  │          0 │ dense_21[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_22 (Dense)    │ (None, 144, 64)   │      8,256 │ dropout_28[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_29          │ (None, 144, 64)   │          0 │ dense_22[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_17 (Add)        │ (None, 144, 64)   │          0 │ dropout_29[0][0], │
│                     │                   │            │ add_16[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_17[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_18 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ add_17[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_18[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_23 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_31          │ (None, 144, 128)  │          0 │ dense_23[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│                     │                   │            │                   │
│ dense_24 (Dense)    │ (None, 144, 64)   │       8,256 │ dropout_31[0][0]  │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dropout_32          │ (None, 144, 64)   │           0 │ dense_24[0][0]    │
│ (Dropout)           │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ add_19 (Add)        │ (None, 144, 64)   │           0 │ dropout_32[0][0], │
│                     │                   │             │ add_18[0][0]      │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │         128 │ add_19[0][0]      │
│ (LayerNormalizatio… │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │      66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │             │ layer_normalizat… │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ add_20 (Add)        │ (None, 144, 64)   │           0 │ multi_head_atten… │
│                     │                   │             │ add_19[0][0]      │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │         128 │ add_20[0][0]      │
│ (LayerNormalizatio… │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dense_25 (Dense)    │ (None, 144, 128)  │       8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dropout_34          │ (None, 144, 128)  │           0 │ dense_25[0][0]    │
│ (Dropout)           │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dense_26 (Dense)    │ (None, 144, 64)   │       8,256 │ dropout_34[0][0]  │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dropout_35          │ (None, 144, 64)   │           0 │ dense_26[0][0]    │
│ (Dropout)           │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ add_21 (Add)        │ (None, 144, 64)   │           0 │ dropout_35[0][0], │
│                     │                   │             │ add_20[0][0]      │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │         128 │ add_21[0][0]      │
│ (LayerNormalizatio… │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │      66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │             │ layer_normalizat… │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ add_22 (Add)        │ (None, 144, 64)   │           0 │ multi_head_atten… │
│                     │                   │             │ add_21[0][0]      │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │         128 │ add_22[0][0]      │
│ (LayerNormalizatio… │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dense_27 (Dense)    │ (None, 144, 128)  │       8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dropout_37          │ (None, 144, 128)  │           0 │ dense_27[0][0]    │
│ (Dropout)           │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dense_28 (Dense)    │ (None, 144, 64)   │       8,256 │ dropout_37[0][0]  │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ dropout_38          │ (None, 144, 64)   │           0 │ dense_28[0][0]    │
│ (Dropout)           │                   │             │                   │
├─────────────────────┼───────────────────┼─────────────┼───────────────────┤
│ add_23 (Add)        │ (None, 144, 64)   │           0 │ dropout_38[0][0], │
│                     │                   │            │ add_22[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_23[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_24 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ add_23[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_24[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_29 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_40          │ (None, 144, 128)  │          0 │ dense_29[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_30 (Dense)    │ (None, 144, 64)   │      8,256 │ dropout_40[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_41          │ (None, 144, 64)   │          0 │ dense_30[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_25 (Add)        │ (None, 144, 64)   │          0 │ dropout_41[0][0], │
│                     │                   │            │ add_24[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_25[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_26 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ add_25[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_26[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_31 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_43          │ (None, 144, 128)  │          0 │ dense_31[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_32 (Dense)    │ (None, 144, 64)   │      8,256 │ dropout_43[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_44          │ (None, 144, 64)   │          0 │ dense_32[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_27 (Add)        │ (None, 144, 64)   │          0 │ dropout_44[0][0], │
│                     │                   │            │ add_26[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_27[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_28 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ add_27[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_28[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_33 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_46          │ (None, 144, 128)  │          0 │ dense_33[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_34 (Dense)    │ (None, 144, 64)   │      8,256 │ dropout_46[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_47          │ (None, 144, 64)   │          0 │ dense_34[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_29 (Add)        │ (None, 144, 64)   │          0 │ dropout_47[0][0], │
│                     │                   │            │ add_28[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_29[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ multi_head_attenti… │ (None, 144, 64)   │     66,368 │ layer_normalizat… │
│ (MultiHeadAttentio… │                   │            │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_30 (Add)        │ (None, 144, 64)   │          0 │ multi_head_atten… │
│                     │                   │            │ add_29[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_30[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_35 (Dense)    │ (None, 144, 128)  │      8,320 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_49          │ (None, 144, 128)  │          0 │ dense_35[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_36 (Dense)    │ (None, 144, 64)   │      8,256 │ dropout_49[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_50          │ (None, 144, 64)   │          0 │ dense_36[0][0]    │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ add_31 (Add)        │ (None, 144, 64)   │          0 │ dropout_50[0][0], │
│                     │                   │            │ add_30[0][0]      │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ layer_normalizatio… │ (None, 144, 64)   │        128 │ add_31[0][0]      │
│ (LayerNormalizatio… │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ flatten_1 (Flatten) │ (None, 9216)      │          0 │ layer_normalizat… │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dropout_51          │ (None, 9216)      │          0 │ flatten_1[0][0]   │
│ (Dropout)           │                   │            │                   │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
│ dense_37 (Dense)    │ (None, 2048)      │ 18,876,416 │ dropout_51[0][0]  │
├─────────────────────┼───────────────────┼────────────┼───────────────────┤
```

```
│ dropout_52           │ (None, 2048)       │            0 │ dense_37[0][0]       │
│ (Dropout)            │                    │              │                      │
├──────────────────────┼────────────────────┼──────────────┼──────────────────────┤
│ dense_38 (Dense)     │ (None, 1024)       │    2,098,176 │ dropout_52[0][0]     │
├──────────────────────┼────────────────────┼──────────────┼──────────────────────┤
│ dropout_53           │ (None, 1024)       │            0 │ dense_38[0][0]       │
│ (Dropout)            │                    │              │                      │
├──────────────────────┼────────────────────┼──────────────┼──────────────────────┤
│ dense_39 (Dense)     │ (None, 10)         │       10,250 │ dropout_53[0][0]     │
└──────────────────────┴────────────────────┴──────────────┴──────────────────────┘
```

```
 Total params: 21,666,769 (82.65 MB)

 Trainable params: 21,666,762 (82.65 MB)

 Non-trainable params: 7 (32.00 B)
```

In [31]:

```
history = train(vit_classifier)
```

```
Epoch 1/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.1647 - loss: 3.9922
Epoch 1: val_accuracy improved from -inf to 0.26673, saving model to c:\Users\Admin
\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 122s 642ms/step - accuracy: 0.1649 - loss: 3.9853 - val
_accuracy: 0.2667 - val_loss: 2.0816 - learning_rate: 0.0010
Epoch 2/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 560ms/step - accuracy: 0.2501 - loss: 2.1435
Epoch 2: val_accuracy improved from 0.26673 to 0.33174, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.2502 - loss: 2.1433 - val_
accuracy: 0.3317 - val_loss: 1.9000 - learning_rate: 0.0010
Epoch 3/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 608ms/step - accuracy: 0.2767 - loss: 2.0216
Epoch 3: val_accuracy improved from 0.33174 to 0.34130, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 103s 676ms/step - accuracy: 0.2767 - loss: 2.0217 - val
_accuracy: 0.3413 - val_loss: 1.9018 - learning_rate: 0.0010
Epoch 4/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.3043 - loss: 1.9498
Epoch 4: val_accuracy improved from 0.34130 to 0.34704, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.3043 - loss: 1.9499 - val_
accuracy: 0.3470 - val_loss: 1.8529 - learning_rate: 0.0010
Epoch 5/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 556ms/step - accuracy: 0.3334 - loss: 1.9343
Epoch 5: val_accuracy improved from 0.34704 to 0.39771, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.3333 - loss: 1.9341 - val_
accuracy: 0.3977 - val_loss: 1.7562 - learning_rate: 0.0010
Epoch 6/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.3544 - loss: 1.8274
Epoch 6: val_accuracy improved from 0.39771 to 0.42017, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.3545 - loss: 1.8273 - val_
accuracy: 0.4202 - val_loss: 1.6521 - learning_rate: 0.0010
Epoch 7/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.3890 - loss: 1.7772
Epoch 7: val_accuracy improved from 0.42017 to 0.44790, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.3890 - loss: 1.7771 - val_
accuracy: 0.4479 - val_loss: 1.6064 - learning_rate: 0.0010
Epoch 8/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.4131 - loss: 1.7098
Epoch 8: val_accuracy improved from 0.44790 to 0.50335, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.4131 - loss: 1.7098 - val_
accuracy: 0.5033 - val_loss: 1.4756 - learning_rate: 0.0010
Epoch 9/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.4257 - loss: 1.6820
Epoch 9: val_accuracy improved from 0.50335 to 0.51673, saving model to c:\Users\Adm
in\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.4258 - loss: 1.6816 - val_
accuracy: 0.5167 - val_loss: 1.4301 - learning_rate: 0.0010
Epoch 10/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.4571 - loss: 1.5784
```

```
Epoch 10: val_accuracy improved from 0.51673 to 0.53107, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.4572 - loss: 1.5782 - val_
accuracy: 0.5311 - val_loss: 1.3678 - learning_rate: 0.0010
Epoch 11/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.4878 - loss: 1.5049
Epoch 11: val_accuracy improved from 0.53107 to 0.58031, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.4878 - loss: 1.5047 - val_
accuracy: 0.5803 - val_loss: 1.2611 - learning_rate: 0.0010
Epoch 12/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.4870 - loss: 1.4878
Epoch 12: val_accuracy improved from 0.58031 to 0.59799, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.4871 - loss: 1.4876 - val_
accuracy: 0.5980 - val_loss: 1.2277 - learning_rate: 0.0010
Epoch 13/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.5385 - loss: 1.3913
Epoch 13: val_accuracy improved from 0.59799 to 0.60421, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.5385 - loss: 1.3911 - val_
accuracy: 0.6042 - val_loss: 1.2028 - learning_rate: 0.0010
Epoch 14/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.5382 - loss: 1.3816
Epoch 14: val_accuracy improved from 0.60421 to 0.61950, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.5382 - loss: 1.3815 - val_
accuracy: 0.6195 - val_loss: 1.1832 - learning_rate: 0.0010
Epoch 15/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.5644 - loss: 1.2833
Epoch 15: val_accuracy improved from 0.61950 to 0.63289, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.5644 - loss: 1.2834 - val_
accuracy: 0.6329 - val_loss: 1.1070 - learning_rate: 0.0010
Epoch 16/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.5784 - loss: 1.2389
Epoch 16: val_accuracy improved from 0.63289 to 0.67065, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.5784 - loss: 1.2388 - val_
accuracy: 0.6707 - val_loss: 0.9989 - learning_rate: 0.0010
Epoch 17/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 550ms/step - accuracy: 0.6073 - loss: 1.1956
Epoch 17: val_accuracy improved from 0.67065 to 0.68212, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 618ms/step - accuracy: 0.6074 - loss: 1.1956 - val_
accuracy: 0.6821 - val_loss: 0.9817 - learning_rate: 0.0010
Epoch 18/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 550ms/step - accuracy: 0.6195 - loss: 1.1429
Epoch 18: val_accuracy did not improve from 0.68212
153/153 ━━━━━━━━━━━━━━━━━━━━ 93s 611ms/step - accuracy: 0.6195 - loss: 1.1431 - val_
accuracy: 0.6625 - val_loss: 1.0301 - learning_rate: 0.0010
Epoch 19/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 552ms/step - accuracy: 0.6357 - loss: 1.1084
Epoch 19: val_accuracy improved from 0.68212 to 0.71558, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 619ms/step - accuracy: 0.6357 - loss: 1.1082 - val_
```

```
accuracy: 0.7156 - val_loss: 0.8844 - learning_rate: 0.0010
Epoch 20/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 551ms/step - accuracy: 0.6640 - loss: 0.9959
Epoch 20: val_accuracy improved from 0.71558 to 0.72467, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 619ms/step - accuracy: 0.6640 - loss: 0.9960 - val_
accuracy: 0.7247 - val_loss: 0.8535 - learning_rate: 0.0010
Epoch 21/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 552ms/step - accuracy: 0.6618 - loss: 1.0185
Epoch 21: val_accuracy improved from 0.72467 to 0.76243, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 619ms/step - accuracy: 0.6618 - loss: 1.0184 - val_
accuracy: 0.7624 - val_loss: 0.7804 - learning_rate: 0.0010
Epoch 22/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 552ms/step - accuracy: 0.6860 - loss: 0.9509
Epoch 22: val_accuracy did not improve from 0.76243
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 612ms/step - accuracy: 0.6860 - loss: 0.9509 - val_
accuracy: 0.7615 - val_loss: 0.7812 - learning_rate: 0.0010
Epoch 23/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 552ms/step - accuracy: 0.6971 - loss: 0.9114
Epoch 23: val_accuracy did not improve from 0.76243
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 612ms/step - accuracy: 0.6971 - loss: 0.9115 - val_
accuracy: 0.7395 - val_loss: 0.8377 - learning_rate: 0.0010
Epoch 24/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 553ms/step - accuracy: 0.7016 - loss: 0.9146
Epoch 24: val_accuracy improved from 0.76243 to 0.79302, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 622ms/step - accuracy: 0.7017 - loss: 0.9143 - val_
accuracy: 0.7930 - val_loss: 0.6699 - learning_rate: 0.0010
Epoch 25/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 552ms/step - accuracy: 0.7408 - loss: 0.8196
Epoch 25: val_accuracy did not improve from 0.79302
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 613ms/step - accuracy: 0.7407 - loss: 0.8198 - val_
accuracy: 0.7868 - val_loss: 0.6967 - learning_rate: 0.0010
Epoch 26/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 554ms/step - accuracy: 0.7341 - loss: 0.7888
Epoch 26: val_accuracy improved from 0.79302 to 0.80497, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 622ms/step - accuracy: 0.7342 - loss: 0.7888 - val_
accuracy: 0.8050 - val_loss: 0.6257 - learning_rate: 0.0010
Epoch 27/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 553ms/step - accuracy: 0.7599 - loss: 0.7497
Epoch 27: val_accuracy did not improve from 0.80497
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 615ms/step - accuracy: 0.7598 - loss: 0.7498 - val_
accuracy: 0.7983 - val_loss: 0.6462 - learning_rate: 0.0010
Epoch 28/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 554ms/step - accuracy: 0.7785 - loss: 0.6954
Epoch 28: val_accuracy improved from 0.80497 to 0.81883, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 622ms/step - accuracy: 0.7785 - loss: 0.6955 - val_
accuracy: 0.8188 - val_loss: 0.5974 - learning_rate: 0.0010
Epoch 29/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 553ms/step - accuracy: 0.7873 - loss: 0.6665
Epoch 29: val_accuracy did not improve from 0.81883
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 615ms/step - accuracy: 0.7873 - loss: 0.6666 - val_
accuracy: 0.8188 - val_loss: 0.5843 - learning_rate: 0.0010
```

```
Epoch 30/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 554ms/step - accuracy: 0.7897 - loss: 0.6526
Epoch 30: val_accuracy improved from 0.81883 to 0.82505, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 623ms/step - accuracy: 0.7897 - loss: 0.6525 - val_
accuracy: 0.8250 - val_loss: 0.5622 - learning_rate: 0.0010
Epoch 31/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 554ms/step - accuracy: 0.7924 - loss: 0.6102
Epoch 31: val_accuracy improved from 0.82505 to 0.84704, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 623ms/step - accuracy: 0.7925 - loss: 0.6102 - val_
accuracy: 0.8470 - val_loss: 0.4839 - learning_rate: 0.0010
Epoch 32/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 554ms/step - accuracy: 0.8216 - loss: 0.5631
Epoch 32: val_accuracy did not improve from 0.84704
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 615ms/step - accuracy: 0.8216 - loss: 0.5633 - val_
accuracy: 0.8451 - val_loss: 0.4960 - learning_rate: 0.0010
Epoch 33/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8115 - loss: 0.5891
Epoch 33: val_accuracy improved from 0.84704 to 0.85038, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.8115 - loss: 0.5891 - val_
accuracy: 0.8504 - val_loss: 0.4976 - learning_rate: 0.0010
Epoch 34/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 555ms/step - accuracy: 0.8295 - loss: 0.5338
Epoch 34: val_accuracy improved from 0.85038 to 0.85516, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.8295 - loss: 0.5339 - val_
accuracy: 0.8552 - val_loss: 0.4705 - learning_rate: 0.0010
Epoch 35/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 555ms/step - accuracy: 0.8396 - loss: 0.5001
Epoch 35: val_accuracy improved from 0.85516 to 0.87906, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.8396 - loss: 0.5002 - val_
accuracy: 0.8791 - val_loss: 0.4209 - learning_rate: 0.0010
Epoch 36/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 555ms/step - accuracy: 0.8392 - loss: 0.5087
Epoch 36: val_accuracy improved from 0.87906 to 0.88910, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.8392 - loss: 0.5087 - val_
accuracy: 0.8891 - val_loss: 0.3902 - learning_rate: 0.0010
Epoch 37/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.8689 - loss: 0.4051
Epoch 37: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 620ms/step - accuracy: 0.8687 - loss: 0.4055 - val_
accuracy: 0.8767 - val_loss: 0.4262 - learning_rate: 0.0010
Epoch 38/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 555ms/step - accuracy: 0.8605 - loss: 0.4274
Epoch 38: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 617ms/step - accuracy: 0.8605 - loss: 0.4276 - val_
accuracy: 0.8824 - val_loss: 0.3937 - learning_rate: 0.0010
Epoch 39/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 556ms/step - accuracy: 0.8700 - loss: 0.4274
Epoch 39: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 618ms/step - accuracy: 0.8700 - loss: 0.4276 - val_
accuracy: 0.8748 - val_loss: 0.4118 - learning_rate: 0.0010
```

```
Epoch 40/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 556ms/step - accuracy: 0.8626 - loss: 0.4486
Epoch 40: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 94s 617ms/step - accuracy: 0.8626 - loss: 0.4485 - val_
accuracy: 0.8853 - val_loss: 0.3712 - learning_rate: 0.0010
Epoch 41/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 556ms/step - accuracy: 0.8583 - loss: 0.4358
Epoch 41: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 618ms/step - accuracy: 0.8583 - loss: 0.4358 - val_
accuracy: 0.8881 - val_loss: 0.3763 - learning_rate: 0.0010
Epoch 42/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 556ms/step - accuracy: 0.8797 - loss: 0.3651
Epoch 42: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 619ms/step - accuracy: 0.8797 - loss: 0.3652 - val_
accuracy: 0.8748 - val_loss: 0.4178 - learning_rate: 0.0010
Epoch 43/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8894 - loss: 0.3435
Epoch 43: val_accuracy did not improve from 0.88910
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 619ms/step - accuracy: 0.8893 - loss: 0.3436 - val_
accuracy: 0.8891 - val_loss: 0.3613 - learning_rate: 0.0010
Epoch 44/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8754 - loss: 0.3861
Epoch 44: val_accuracy improved from 0.88910 to 0.89006, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.8754 - loss: 0.3862 - val_
accuracy: 0.8901 - val_loss: 0.3779 - learning_rate: 0.0010
Epoch 45/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8816 - loss: 0.3582
Epoch 45: val_accuracy improved from 0.89006 to 0.89293, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 628ms/step - accuracy: 0.8817 - loss: 0.3582 - val_
accuracy: 0.8929 - val_loss: 0.3859 - learning_rate: 0.0010
Epoch 46/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8991 - loss: 0.3057
Epoch 46: val_accuracy improved from 0.89293 to 0.90057, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 628ms/step - accuracy: 0.8991 - loss: 0.3059 - val_
accuracy: 0.9006 - val_loss: 0.3758 - learning_rate: 0.0010
Epoch 47/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.9011 - loss: 0.2994
Epoch 47: val_accuracy did not improve from 0.90057
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 620ms/step - accuracy: 0.9010 - loss: 0.2995 - val_
accuracy: 0.8872 - val_loss: 0.3851 - learning_rate: 0.0010
Epoch 48/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 557ms/step - accuracy: 0.8975 - loss: 0.3118
Epoch 48: val_accuracy improved from 0.90057 to 0.90296, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 628ms/step - accuracy: 0.8975 - loss: 0.3119 - val_
accuracy: 0.9030 - val_loss: 0.3267 - learning_rate: 0.0010
Epoch 49/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.9076 - loss: 0.3046
Epoch 49: val_accuracy did not improve from 0.90296
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 621ms/step - accuracy: 0.9075 - loss: 0.3046 - val_
accuracy: 0.8963 - val_loss: 0.3408 - learning_rate: 0.0010
Epoch 50/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.8965 - loss: 0.3341
```

```
Epoch 50: val_accuracy did not improve from 0.90296
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 621ms/step - accuracy: 0.8966 - loss: 0.3340 - val_
accuracy: 0.9011 - val_loss: 0.3504 - learning_rate: 0.0010
Epoch 51/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.8965 - loss: 0.3368
Epoch 51: val_accuracy did not improve from 0.90296
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 622ms/step - accuracy: 0.8965 - loss: 0.3367 - val_
accuracy: 0.8991 - val_loss: 0.3620 - learning_rate: 0.0010
Epoch 52/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 558ms/step - accuracy: 0.9097 - loss: 0.2777
Epoch 52: val_accuracy improved from 0.90296 to 0.90344, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 629ms/step - accuracy: 0.9096 - loss: 0.2778 - val_
accuracy: 0.9034 - val_loss: 0.3450 - learning_rate: 0.0010
Epoch 53/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9050 - loss: 0.3126
Epoch 53: val_accuracy improved from 0.90344 to 0.90583, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 630ms/step - accuracy: 0.9050 - loss: 0.3126 - val_
accuracy: 0.9058 - val_loss: 0.3324 - learning_rate: 0.0010
Epoch 54/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 560ms/step - accuracy: 0.9153 - loss: 0.2608
Epoch 54: val_accuracy did not improve from 0.90583
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 623ms/step - accuracy: 0.9153 - loss: 0.2609 - val_
accuracy: 0.9039 - val_loss: 0.3469 - learning_rate: 0.0010
Epoch 55/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9126 - loss: 0.2746
Epoch 55: val_accuracy did not improve from 0.90583
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 623ms/step - accuracy: 0.9126 - loss: 0.2746 - val_
accuracy: 0.9030 - val_loss: 0.3496 - learning_rate: 0.0010
Epoch 56/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9104 - loss: 0.2888
Epoch 56: val_accuracy did not improve from 0.90583
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 623ms/step - accuracy: 0.9104 - loss: 0.2887 - val_
accuracy: 0.9034 - val_loss: 0.4116 - learning_rate: 0.0010
Epoch 57/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9192 - loss: 0.2430
Epoch 57: val_accuracy improved from 0.90583 to 0.90631, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 631ms/step - accuracy: 0.9191 - loss: 0.2431 - val_
accuracy: 0.9063 - val_loss: 0.3559 - learning_rate: 0.0010
Epoch 58/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 560ms/step - accuracy: 0.9182 - loss: 0.2632
Epoch 58: val_accuracy improved from 0.90631 to 0.90822, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 631ms/step - accuracy: 0.9182 - loss: 0.2632 - val_
accuracy: 0.9082 - val_loss: 0.3674 - learning_rate: 0.0010
Epoch 59/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9219 - loss: 0.2478
Epoch 59: val_accuracy improved from 0.90822 to 0.91252, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 631ms/step - accuracy: 0.9219 - loss: 0.2479 - val_
accuracy: 0.9125 - val_loss: 0.3532 - learning_rate: 0.0010
Epoch 60/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 560ms/step - accuracy: 0.9249 - loss: 0.2306
Epoch 60: val_accuracy improved from 0.91252 to 0.92113, saving model to c:\Users\Ad
```

```
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 631ms/step - accuracy: 0.9249 - loss: 0.2307 - val_
accuracy: 0.9211 - val_loss: 0.3096 - learning_rate: 0.0010
Epoch 61/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 559ms/step - accuracy: 0.9255 - loss: 0.2327
Epoch 61: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 95s 624ms/step - accuracy: 0.9255 - loss: 0.2328 - val_
accuracy: 0.9101 - val_loss: 0.3489 - learning_rate: 0.0010
Epoch 62/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9118 - loss: 0.2709
Epoch 62: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.9118 - loss: 0.2709 - val_
accuracy: 0.9025 - val_loss: 0.4081 - learning_rate: 0.0010
Epoch 63/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 560ms/step - accuracy: 0.9158 - loss: 0.2511
Epoch 63: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.9158 - loss: 0.2511 - val_
accuracy: 0.9144 - val_loss: 0.3552 - learning_rate: 0.0010
Epoch 64/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9277 - loss: 0.2288
Epoch 64: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9277 - loss: 0.2289 - val_
accuracy: 0.9168 - val_loss: 0.3285 - learning_rate: 0.0010
Epoch 65/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9327 - loss: 0.2075
Epoch 65: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9327 - loss: 0.2076 - val_
accuracy: 0.9125 - val_loss: 0.3608 - learning_rate: 0.0010
Epoch 66/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9211 - loss: 0.2482
Epoch 66: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 625ms/step - accuracy: 0.9211 - loss: 0.2482 - val_
accuracy: 0.9082 - val_loss: 0.3517 - learning_rate: 0.0010
Epoch 67/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9330 - loss: 0.2385
Epoch 67: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9330 - loss: 0.2385 - val_
accuracy: 0.9015 - val_loss: 0.3830 - learning_rate: 0.0010
Epoch 68/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9333 - loss: 0.2283
Epoch 68: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9333 - loss: 0.2283 - val_
accuracy: 0.9106 - val_loss: 0.3762 - learning_rate: 0.0010
Epoch 69/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9178 - loss: 0.2638
Epoch 69: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9179 - loss: 0.2637 - val_
accuracy: 0.9192 - val_loss: 0.3676 - learning_rate: 0.0010
Epoch 70/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9420 - loss: 0.1798
Epoch 70: val_accuracy did not improve from 0.92113

Epoch 70: ReduceLROnPlateau reducing learning rate to 0.00010000000474974513.
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9419 - loss: 0.1800 - val_
accuracy: 0.9054 - val_loss: 0.3696 - learning_rate: 0.0010
Epoch 71/100
```

```
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9373 - loss: 0.1975
Epoch 71: val_accuracy did not improve from 0.92113
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 628ms/step - accuracy: 0.9373 - loss: 0.1974 - val_
accuracy: 0.9183 - val_loss: 0.3147 - learning_rate: 1.0000e-04
Epoch 72/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9566 - loss: 0.1397
Epoch 72: val_accuracy improved from 0.92113 to 0.92686, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 634ms/step - accuracy: 0.9566 - loss: 0.1397 - val_
accuracy: 0.9269 - val_loss: 0.3151 - learning_rate: 1.0000e-04
Epoch 73/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9600 - loss: 0.1290
Epoch 73: val_accuracy did not improve from 0.92686
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9600 - loss: 0.1290 - val_
accuracy: 0.9269 - val_loss: 0.3121 - learning_rate: 1.0000e-04
Epoch 74/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9635 - loss: 0.1170
Epoch 74: val_accuracy improved from 0.92686 to 0.93308, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 636ms/step - accuracy: 0.9635 - loss: 0.1169 - val_
accuracy: 0.9331 - val_loss: 0.3123 - learning_rate: 1.0000e-04
Epoch 75/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 564ms/step - accuracy: 0.9615 - loss: 0.1066
Epoch 75: val_accuracy did not improve from 0.93308
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 629ms/step - accuracy: 0.9615 - loss: 0.1066 - val_
accuracy: 0.9321 - val_loss: 0.3177 - learning_rate: 1.0000e-04
Epoch 76/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9680 - loss: 0.1010
Epoch 76: val_accuracy did not improve from 0.93308
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9679 - loss: 0.1010 - val_
accuracy: 0.9307 - val_loss: 0.3128 - learning_rate: 1.0000e-04
Epoch 77/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9732 - loss: 0.0791
Epoch 77: val_accuracy improved from 0.93308 to 0.93356, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 634ms/step - accuracy: 0.9732 - loss: 0.0791 - val_
accuracy: 0.9336 - val_loss: 0.3135 - learning_rate: 1.0000e-04
Epoch 78/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9677 - loss: 0.0974
Epoch 78: val_accuracy did not improve from 0.93356
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9677 - loss: 0.0974 - val_
accuracy: 0.9316 - val_loss: 0.3194 - learning_rate: 1.0000e-04
Epoch 79/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9690 - loss: 0.0968
Epoch 79: val_accuracy improved from 0.93356 to 0.93451, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 634ms/step - accuracy: 0.9690 - loss: 0.0967 - val_
accuracy: 0.9345 - val_loss: 0.3167 - learning_rate: 1.0000e-04
Epoch 80/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9733 - loss: 0.0891
Epoch 80: val_accuracy improved from 0.93451 to 0.93690, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 634ms/step - accuracy: 0.9733 - loss: 0.0892 - val_
accuracy: 0.9369 - val_loss: 0.3250 - learning_rate: 1.0000e-04
Epoch 81/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9746 - loss: 0.0756
```

```
Epoch 81: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9746 - loss: 0.0756 - val_
accuracy: 0.9364 - val_loss: 0.3149 - learning_rate: 1.0000e-04
Epoch 82/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9735 - loss: 0.0875
Epoch 82: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9735 - loss: 0.0875 - val_
accuracy: 0.9350 - val_loss: 0.3172 - learning_rate: 1.0000e-04
Epoch 83/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9734 - loss: 0.0735
Epoch 83: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9734 - loss: 0.0736 - val_
accuracy: 0.9336 - val_loss: 0.3157 - learning_rate: 1.0000e-04
Epoch 84/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9774 - loss: 0.0642
Epoch 84: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9774 - loss: 0.0642 - val_
accuracy: 0.9345 - val_loss: 0.3348 - learning_rate: 1.0000e-04
Epoch 85/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9793 - loss: 0.0654
Epoch 85: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9793 - loss: 0.0655 - val_
accuracy: 0.9359 - val_loss: 0.3188 - learning_rate: 1.0000e-04
Epoch 86/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9718 - loss: 0.0838
Epoch 86: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9718 - loss: 0.0838 - val_
accuracy: 0.9359 - val_loss: 0.3296 - learning_rate: 1.0000e-04
Epoch 87/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 564ms/step - accuracy: 0.9725 - loss: 0.0756
Epoch 87: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 629ms/step - accuracy: 0.9725 - loss: 0.0756 - val_
accuracy: 0.9355 - val_loss: 0.3255 - learning_rate: 1.0000e-04
Epoch 88/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9731 - loss: 0.0816
Epoch 88: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9731 - loss: 0.0816 - val_
accuracy: 0.9364 - val_loss: 0.3304 - learning_rate: 1.0000e-04
Epoch 89/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9751 - loss: 0.0648
Epoch 89: val_accuracy did not improve from 0.93690
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9751 - loss: 0.0648 - val_
accuracy: 0.9345 - val_loss: 0.3354 - learning_rate: 1.0000e-04
Epoch 90/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9765 - loss: 0.0734
Epoch 90: val_accuracy improved from 0.93690 to 0.93881, saving model to c:\Users\Ad
min\Desktop\COSC2753_A2_MachineLearning/paddy_models/model_vgg_new.keras
153/153 ━━━━━━━━━━━━━━━━━━━━ 97s 633ms/step - accuracy: 0.9765 - loss: 0.0734 - val_
accuracy: 0.9388 - val_loss: 0.3308 - learning_rate: 1.0000e-04
Epoch 91/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9767 - loss: 0.0631
Epoch 91: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9767 - loss: 0.0631 - val_
accuracy: 0.9379 - val_loss: 0.3335 - learning_rate: 1.0000e-04
Epoch 92/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9775 - loss: 0.0726
```

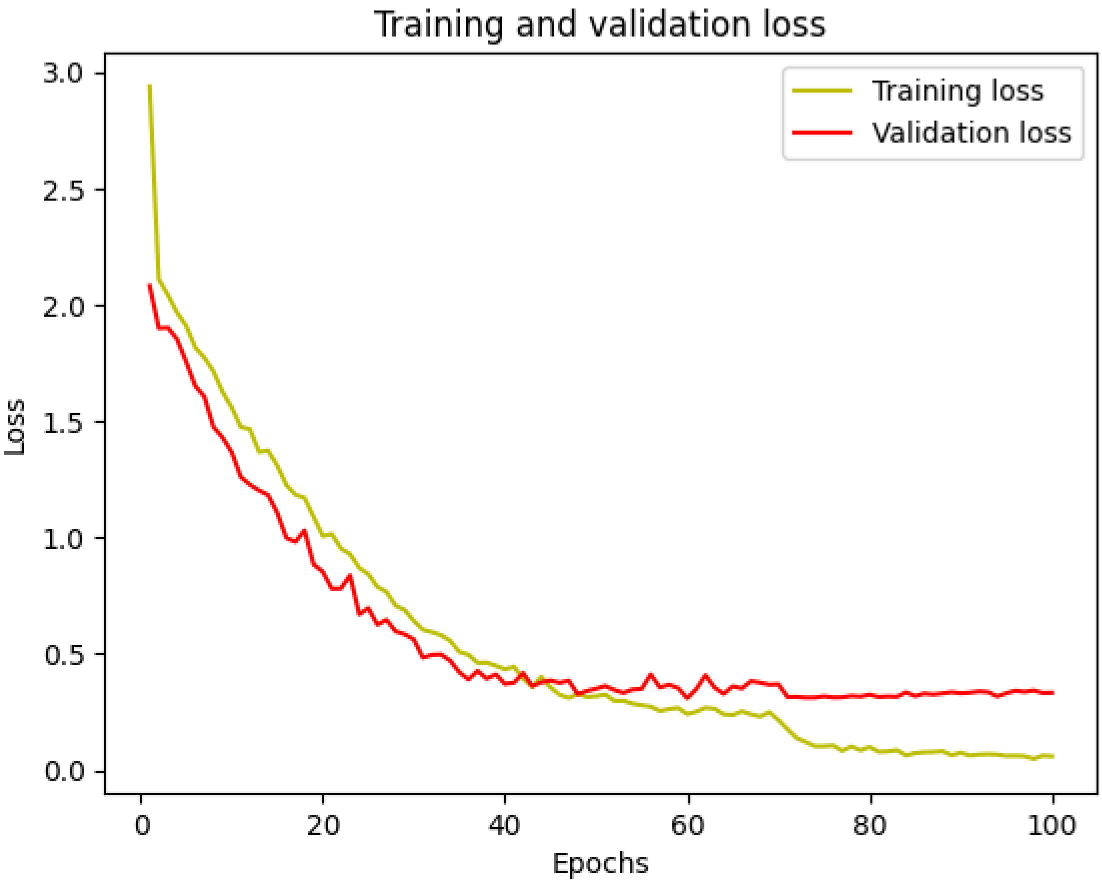
```
Epoch 92: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9775 - loss: 0.0725 - val_
accuracy: 0.9379 - val_loss: 0.3388 - learning_rate: 1.0000e-04
Epoch 93/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 562ms/step - accuracy: 0.9759 - loss: 0.0695
Epoch 93: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9759 - loss: 0.0695 - val_
accuracy: 0.9379 - val_loss: 0.3360 - learning_rate: 1.0000e-04
Epoch 94/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 563ms/step - accuracy: 0.9768 - loss: 0.0732
Epoch 94: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 627ms/step - accuracy: 0.9768 - loss: 0.0732 - val_
accuracy: 0.9388 - val_loss: 0.3174 - learning_rate: 1.0000e-04
Epoch 95/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 561ms/step - accuracy: 0.9808 - loss: 0.0669
Epoch 95: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 96s 626ms/step - accuracy: 0.9808 - loss: 0.0669 - val_
accuracy: 0.9359 - val_loss: 0.3311 - learning_rate: 1.0000e-04
Epoch 96/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 583ms/step - accuracy: 0.9778 - loss: 0.0622
Epoch 96: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 99s 648ms/step - accuracy: 0.9778 - loss: 0.0622 - val_
accuracy: 0.9369 - val_loss: 0.3416 - learning_rate: 1.0000e-04
Epoch 97/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 575ms/step - accuracy: 0.9793 - loss: 0.0703
Epoch 97: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 98s 641ms/step - accuracy: 0.9793 - loss: 0.0703 - val_
accuracy: 0.9355 - val_loss: 0.3371 - learning_rate: 1.0000e-04
Epoch 98/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 576ms/step - accuracy: 0.9823 - loss: 0.0465
Epoch 98: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 98s 642ms/step - accuracy: 0.9823 - loss: 0.0465 - val_
accuracy: 0.9340 - val_loss: 0.3429 - learning_rate: 1.0000e-04
Epoch 99/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 577ms/step - accuracy: 0.9778 - loss: 0.0649
Epoch 99: val_accuracy did not improve from 0.93881
153/153 ━━━━━━━━━━━━━━━━━━━━ 98s 643ms/step - accuracy: 0.9778 - loss: 0.0649 - val_
accuracy: 0.9355 - val_loss: 0.3320 - learning_rate: 1.0000e-04
Epoch 100/100
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 579ms/step - accuracy: 0.9832 - loss: 0.0519
Epoch 100: val_accuracy did not improve from 0.93881

Epoch 100: ReduceLROnPlateau reducing learning rate to 1.0000000474974514e-05.
153/153 ━━━━━━━━━━━━━━━━━━━━ 99s 645ms/step - accuracy: 0.9832 - loss: 0.0520 - val_
accuracy: 0.9374 - val_loss: 0.3330 - learning_rate: 1.0000e-04
108/108 ━━━━━━━━━━━━━━━━━━━━ 17s 161ms/step - accuracy: 0.9255 - loss: 0.3696
Test accuracy: 92.55%
```

In [51]:

```python
vit_classifier.save_weights(HOME_PATH + '/paddy_models/vit.weights.h5')
vit_classifier.save(HOME_PATH + '/paddy_models/vit_model.keras')
```

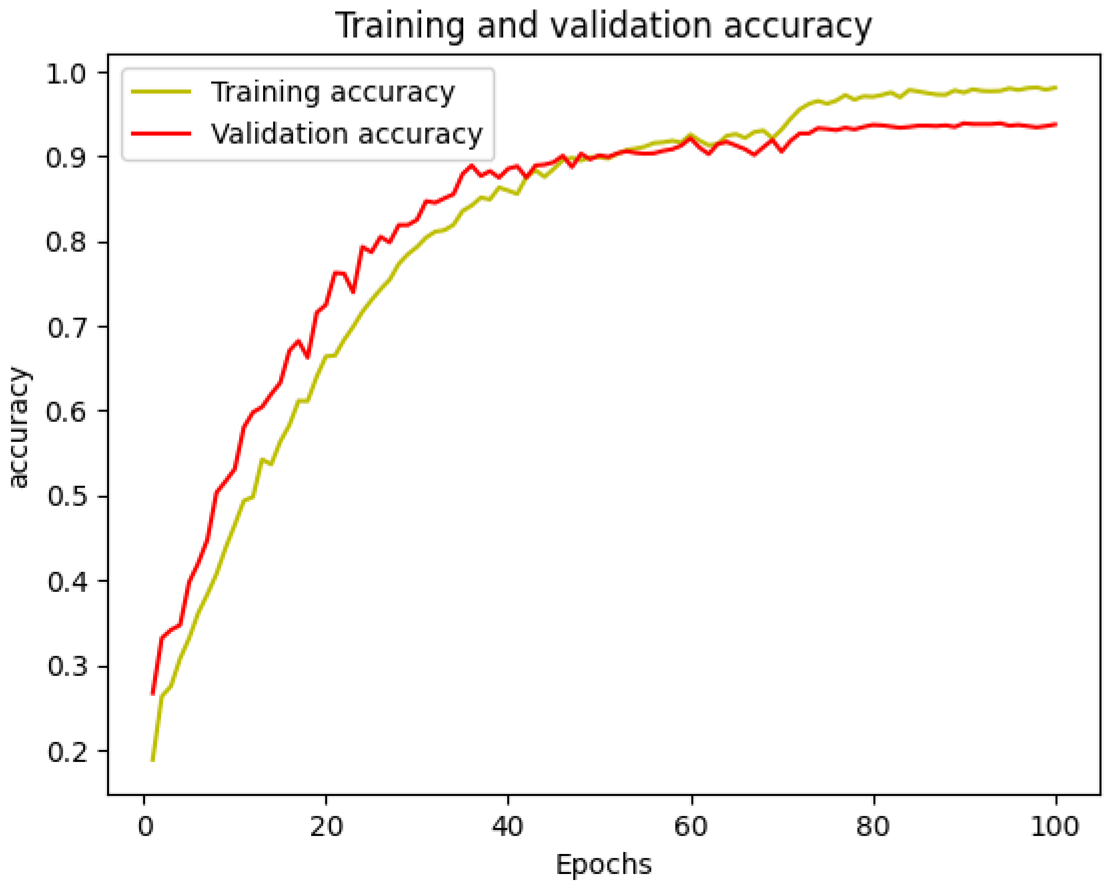
In [43]:

```python
history = history
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(1, len(loss) + 1)
plt.plot(epochs, loss, 'y', label='Training loss')
plt.plot(epochs, val_loss, 'r', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
```



In [44]:

```
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

plt.plot(epochs, acc, 'y', label='Training accuracy')
plt.plot(epochs, val_acc, 'r', label='Validation accuracy')
plt.title('Training and validation accuracy')
plt.xlabel('Epochs')
plt.ylabel('accuracy')
plt.legend()
plt.show()
```



Generate Prediction

In [52]:

```
validation_ds = keras.utils.image_dataset_from_directory(
    directory=HOME_PATH + 'train_images',
    batch_size=16,
    image_size=(256, 256),
    validation_split=0.2,
    subset="validation",
    seed=123
)

test_ds = keras.utils.image_dataset_from_directory(
    directory = HOME_PATH + 'test_images',
    batch_size = 16,
    image_size = (256, 256),
    label_mode = None,
    shuffle=False
)

# Predict the labels of the test set
predictions = vit_classifier.predict(test_ds)

predicted_labels = [labels[prediction.argmax()] for prediction in predictions]

loss, accuracy = vit_classifier.evaluate(validation_ds)
print(f'Validation Loss: {loss:.4f}, Validation Accuracy: {accuracy:.4f}')
```

```
Found 10407 files belonging to 10 classes.
Using 2081 files for validation.
Found 3469 files.
217/217 ━━━━━━━━━━━━━━━━━━━━ 19s 86ms/step
131/131 ━━━━━━━━━━━━━━━━━━━━ 11s 84ms/step - accuracy: 0.9603 - loss: 0.1769
Validation Loss: 0.1875, Validation Accuracy: 0.9611
```

In [53]:

```python
# Create a submission file
submission_df = pd.DataFrame({'image_id': test_ds.file_paths, 'label': predicted_la

submission_df['image_id'] = submission_df['image_id'].apply(lambda x: x.split('/')[

submission_df.to_csv('sample_submission.csv', index=False)
```

In [54]:

```python
# Display unique predicted labels
print(set(predicted_labels))

# Display submission head
print(submission_df.head())
```

```
{np.int64(0)}
                 image_id  label
0  test_images\200001.jpg      0
1  test_images\200002.jpg      0
2  test_images\200003.jpg      0
3  test_images\200004.jpg      0
4  test_images\200005.jpg      0
```