In [14]:
```
# Vikas Singh Narwariya
# Roll no.- 231030065
# pandas: Library for handling and analyzing data in tables.
# numpy: Library for numerical operations and working with arrays.
# sklearn (train_test_split): Used to split data into training and testing sets.
# sklearn (StandardScaler, LabelEncoder): StandardScaler normalizes data; LabelE
# sklearn (classification_report, accuracy_score): Used to evaluate the performa
# keras (Sequential): Sequential allows you to build models layer by layer.
# keras (Dense, Dropout, BatchNormalization, LeakyReLU):
# Dense: fully connected layer, Dropout: prevents overfitting, BatchNormalizatio
# keras (ModelCheckpoint): Saves the best model weights during training if accur
# re: Library for working with regular expressions. It is a tool that helps you
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import classification_report, accuracy_score
from keras.models import Sequential
from keras.layers import Dense, Dropout, BatchNormalization, LeakyReLU
from keras.callbacks import ModelCheckpoint
import re

# Helper function to enable natural sorting
# The function breaks a string into digits and non-digit parts
# Take an example, for example, "image10" becomes ['image', 10] for proper sorti
def natural_sort_key(text):
    return [int(chunk) if chunk.isdigit() else chunk for chunk in re.split(r'(\d

# Load dataset and prepare for training
# pd.read_csv() is used to read the CSV file containing features and target labe
# This dataset contains features extracted from images for classification (ravel
df = pd.read_csv('images_for_training.csv')

# Seperate features and target labels
# I use .iloc to slice the DataFrame:
# features: All columns except the last one because in last column target labels
# target: The last column (the classification label)
features = df.iloc[:, :-1].values  # First 34 columns (features)
target = df.iloc[:, -1].values    # Last column (target labels)

# Encode target labels to binary (0 or 1)
# LabelEncoder converts categorical labels like 'Raveling' and 'Non_raveling' in
encoder = LabelEncoder()
encoded_target = encoder.fit_transform(target)  # Encodes labels into 0 (Non_rav

# Split the dataset into training and testing subsets
# train_test_split divides the dataset into two parts:
# Training data (80%) for training the model
# Testing data (20%) for evaluating the model performance
X_train, X_test, y_train, y_test = train_test_split(features, encoded_target, te

# Standardize features to make them have mean = 0 and standard deviation = 1
# This improves the training speed and accuracy of the neural network
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)  # Fit the scaler to the training
X_test_scaled = scaler.transform(X_test)  # Use the same scaler on test data to

# Initialize the Sequential model
```

```
# Sequential is a linear stack of layers for building a neural network model
model = Sequential()

# Add input layer and first hidden layer
# Input layer: This is the first layer where the model takes in the data. In thi
# Dense layer: Every neuron in this layer is connected to all the neurons in the
# LeakyReLU activation: This activation function is similar to ReLU but allows s
# BatchNormalization: This normalizes the inputs to the layer, which helps the n
# Dropout: During training, 30% of the neurons are randomly "dropped" to prevent
# In case of overfitting model become too specific to the training data and not

model.add(Dense(128, input_shape=(X_train_scaled.shape[1],)))  # 34 input featur
model.add(LeakyReLU(alpha=0.2))  # Activation function
model.add(BatchNormalization())  # Normalizes layer output for stable training
model.add(Dropout(0.3))  # Dropout 30% of neurons to prevent overfitting

# Add second hidden layer
# Dense layer with 64 neurons, followed by LeakyReLU, BatchNormalization, and Dr
model.add(Dense(64))  # Fully connected layer with 64 neurons
model.add(LeakyReLU(alpha=0.2))  # Activation function
model.add(BatchNormalization())  # Normalization to speed up training
model.add(Dropout(0.3))  # Dropout 30% of neurons to avoid overfitting

# Add third hidden layer
# Dense layer with 32 neurons, LeakyReLU for activation, BatchNormalization, and
model.add(Dense(32))  # Fully connected layer with 32 neurons
model.add(LeakyReLU(alpha=0.2))  # Leaky ReLU activation function
model.add(BatchNormalization())  # Normalize the outputs of the layer
model.add(Dropout(0.2))  # Dropout 20% of neurons to avoid overfitting

# Add output layer for binary classification
# Output layer (Dense(1)): This layer has 1 neuron because it's a binary classif
# Sigmoid activation: This activation function converts the output into a probab
model.add(Dense(1, activation='sigmoid'))  # Output layer with 1 neuron (for bin

# Loss ('binary_crossentropy'): This function measures the difference between th
# Optimizer ('adam'): Adam is an algorithm that adjusts the learning rate as tra
# Metrics ('accuracy'): Accuracy measures how often the model's predictions matc
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy']

# ModelCheckpoint: This tool saves the model's weights to know what it has learn
# Best accuracy: It only saves the weights if the model reaches the highest accu
checkpoint = ModelCheckpoint('best_model_weights.keras', monitor='accuracy', sav

# Train the model
# fit() trains the model on the training data
# epochs=150: The model will go through the entire training data 150 times
# batch_size=16: Processes 16 samples at a time before updating the weights
# callbacks=[checkpoint]: Saves the best model weights during training
model.fit(X_train_scaled, y_train, epochs=150, batch_size=16, verbose=1, callbac

# Load the best model weights for evaluation
# After training, the best-performing model weights are loaded
model.load_weights('best_model_weights.keras')

# Make predictions on the test set: Use the trained model to predict outcomes fo
# model.predict() outputs probabilities: The model gives a probability score bet
# Use > 0.5 to convert probabilities to binary labels (0 or 1): If the probabili
predictions = (model.predict(X_test_scaled) > 0.5).astype("int32")
```

```
# Display evaluation metrics
# classification_report() prints precision, recall, and F1-score for both classe
# accuracy_score() calculates the accuracy of the model on the test data
print(classification_report(y_test, predictions))  # Print detailed classificati
print("Accuracy:", accuracy_score(y_test, predictions))  # Print the overall acc

# Load external test dataset
# This dataset contains unseen test images, and we use it to make final predicti
external_test_df = pd.read_csv('images_for_testing.csv')

# Filenames are strings and perform natural sorting
# Sorting the external test dataset by filenames in a human-friendly order (e.g.
external_test_df['filename'] = external_test_df['filename'].astype(str)  # Ensur
sorted_test_df = external_test_df.sort_values(by='filename', key=lambda col: col

# Extract and scale the test features (assuming features start from the second c
# We are excluding the filename column and scaling the features using the same s
external_test_features = sorted_test_df.iloc[:, 1:].values  # Extract features f
external_test_scaled = scaler.transform(external_test_features)  # Scale the tes

# Predict on the external test set
# model.predict() gives probabilities, which we convert to binary labels (0 or 1
external_test_predictions = (model.predict(external_test_scaled) > 0.5).astype("

# Prepare DataFrame for output
# Create a DataFrame with filenames and predicted classes for the external test
output_df = pd.DataFrame({
    'filename': sorted_test_df['filename'].values,  # Use the sorted filenames
    'predicted_class': external_test_predictions.flatten()  # Flatten prediction
})

# Map the binary labels back to their original class names
# 1 -> 'Raveling' and 0 -> 'Non_raveling'
output_df['predicted_class'] = output_df['predicted_class'].map({1: 'Raveling',

# Save predictions to a CSV file
# The final predictions are saved to 'final_output_predictions.csv'
output_file_path = 'final_output_predictions_vs_2.csv'
output_df.to_csv(output_file_path, index=False)  # Save the DataFrame to a CSV f

print(f"Predictions saved to {output_file_path}")  # Inform the user that the pr
```

```
D:\Anaconda\Lib\site-packages\keras\src\layers\core\dense.py:87: UserWarning: Do
not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential
models, prefer using an `Input(shape)` object as the first layer in the model ins
tead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
D:\Anaconda\Lib\site-packages\keras\src\layers\activations\leaky_relu.py:41: User
Warning: Argument `alpha` is deprecated. Use `negative_slope` instead.
  warnings.warn(
```

```
Epoch 1/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.7277 - loss: 0.5233
Epoch 1: accuracy improved from -inf to 0.75357, saving model to best_model_weigh
ts.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 11s 11ms/step - accuracy: 0.7299 - loss: 0.5211
Epoch 2/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8002 - loss: 0.4220
Epoch 2: accuracy improved from 0.75357 to 0.78393, saving model to best_model_we
ights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.7982 - loss: 0.4246
Epoch 3/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8032 - loss: 0.4200
Epoch 3: accuracy improved from 0.78393 to 0.81071, saving model to best_model_we
ights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.8040 - loss: 0.4182
Epoch 4/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8324 - loss: 0.3759
Epoch 4: accuracy improved from 0.81071 to 0.81964, saving model to best_model_we
ights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8317 - loss: 0.3768
Epoch 5/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8740 - loss: 0.3678
Epoch 5: accuracy improved from 0.81964 to 0.83214, saving model to best_model_we
ights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8640 - loss: 0.3711
Epoch 6/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8112 - loss: 0.3966
Epoch 6: accuracy did not improve from 0.83214
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8126 - loss: 0.3975
Epoch 7/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8202 - loss: 0.3849
Epoch 7: accuracy did not improve from 0.83214
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8210 - loss: 0.3846
Epoch 8/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8504 - loss: 0.3514
Epoch 8: accuracy did not improve from 0.83214
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8474 - loss: 0.3551
Epoch 9/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8713 - loss: 0.3116
Epoch 9: accuracy improved from 0.83214 to 0.86071, saving model to best_model_we
ights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.8696 - loss: 0.3156
Epoch 10/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8388 - loss: 0.3759
Epoch 10: accuracy did not improve from 0.86071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8383 - loss: 0.3783
Epoch 11/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8519 - loss: 0.3527
Epoch 11: accuracy did not improve from 0.86071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8516 - loss: 0.3507
Epoch 12/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8469 - loss: 0.3684
Epoch 12: accuracy did not improve from 0.86071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8483 - loss: 0.3655
Epoch 13/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8563 - loss: 0.3216
Epoch 13: accuracy did not improve from 0.86071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8568 - loss: 0.3218
Epoch 14/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8700 - loss: 0.3218
```

```
Epoch 14: accuracy improved from 0.86071 to 0.86607, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8702 - loss: 0.3190
Epoch 15/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8356 - loss: 0.3894
Epoch 15: accuracy did not improve from 0.86607
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8389 - loss: 0.3842
Epoch 16/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8482 - loss: 0.3408
Epoch 16: accuracy improved from 0.86607 to 0.86964, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8554 - loss: 0.3329
Epoch 17/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8867 - loss: 0.2778
Epoch 17: accuracy improved from 0.86964 to 0.87321, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8829 - loss: 0.2895
Epoch 18/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8573 - loss: 0.3532
Epoch 18: accuracy did not improve from 0.87321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8604 - loss: 0.3481
Epoch 19/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8663 - loss: 0.3114
Epoch 19: accuracy did not improve from 0.87321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8660 - loss: 0.3114
Epoch 20/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8527 - loss: 0.3300
Epoch 20: accuracy improved from 0.87321 to 0.87500, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8563 - loss: 0.3299
Epoch 21/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8939 - loss: 0.2679
Epoch 21: accuracy did not improve from 0.87500
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8911 - loss: 0.2725
Epoch 22/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8799 - loss: 0.2690
Epoch 22: accuracy improved from 0.87500 to 0.88571, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8809 - loss: 0.2692
Epoch 23/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8780 - loss: 0.2928
Epoch 23: accuracy did not improve from 0.88571
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8790 - loss: 0.2945
Epoch 24/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8753 - loss: 0.3199
Epoch 24: accuracy did not improve from 0.88571
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8753 - loss: 0.3187
Epoch 25/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8623 - loss: 0.3042
Epoch 25: accuracy did not improve from 0.88571
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8631 - loss: 0.3024
Epoch 26/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8845 - loss: 0.3034
Epoch 26: accuracy improved from 0.88571 to 0.89643, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8876 - loss: 0.2944
Epoch 27/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9114 - loss: 0.2518
Epoch 27: accuracy improved from 0.89643 to 0.90714, saving model to best_model_w
eights.keras
```

```
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9107 - loss: 0.2537
Epoch 28/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8458 - loss: 0.3215
Epoch 28: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8488 - loss: 0.3188
Epoch 29/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8914 - loss: 0.2429
Epoch 29: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8877 - loss: 0.2562
Epoch 30/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8297 - loss: 0.3536
Epoch 30: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8377 - loss: 0.3474
Epoch 31/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8899 - loss: 0.3223
Epoch 31: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8905 - loss: 0.3165
Epoch 32/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8869 - loss: 0.2523
Epoch 32: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8845 - loss: 0.2534
Epoch 33/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9009 - loss: 0.2494
Epoch 33: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9008 - loss: 0.2500
Epoch 34/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8863 - loss: 0.2760
Epoch 34: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8858 - loss: 0.2761
Epoch 35/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8936 - loss: 0.2573
Epoch 35: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8935 - loss: 0.2578
Epoch 36/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8537 - loss: 0.3486
Epoch 36: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8543 - loss: 0.3473
Epoch 37/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8923 - loss: 0.2911
Epoch 37: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8919 - loss: 0.2845
Epoch 38/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9276 - loss: 0.2022
Epoch 38: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9264 - loss: 0.2050
Epoch 39/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9114 - loss: 0.2141
Epoch 39: accuracy did not improve from 0.90714
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9095 - loss: 0.2139
Epoch 40/150
23/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9134 - loss: 0.2240
Epoch 40: accuracy improved from 0.90714 to 0.91071, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9151 - loss: 0.2219
Epoch 41/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8816 - loss: 0.2854
Epoch 41: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8828 - loss: 0.2867
Epoch 42/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8660 - loss: 0.3589
```

```
Epoch 42: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8709 - loss: 0.3421
Epoch 43/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.8921 - loss: 0.2469
Epoch 43: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8924 - loss: 0.2490
Epoch 44/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.8802 - loss: 0.2683
Epoch 44: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.8811 - loss: 0.2751
Epoch 45/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9261 - loss: 0.2260
Epoch 45: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9241 - loss: 0.2265
Epoch 46/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8931 - loss: 0.2674
Epoch 46: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8938 - loss: 0.2671
Epoch 47/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8977 - loss: 0.2643
Epoch 47: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8984 - loss: 0.2630
Epoch 48/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8927 - loss: 0.2646
Epoch 48: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8926 - loss: 0.2654
Epoch 49/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8604 - loss: 0.3112
Epoch 49: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8614 - loss: 0.3128
Epoch 50/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8747 - loss: 0.2569
Epoch 50: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8760 - loss: 0.2559
Epoch 51/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9114 - loss: 0.2445
Epoch 51: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9086 - loss: 0.2459
Epoch 52/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9119 - loss: 0.2183
Epoch 52: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9118 - loss: 0.2186
Epoch 53/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8898 - loss: 0.2382
Epoch 53: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8919 - loss: 0.2374
Epoch 54/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8959 - loss: 0.2973
Epoch 54: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8958 - loss: 0.2969
Epoch 55/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9060 - loss: 0.2358
Epoch 55: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9062 - loss: 0.2352
Epoch 56/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9137 - loss: 0.2064
Epoch 56: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9133 - loss: 0.2081
Epoch 57/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8791 - loss: 0.2656
```

```
Epoch 57: accuracy did not improve from 0.91071
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.8842 - loss: 0.2581
Epoch 58/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9247 - loss: 0.2336
Epoch 58: accuracy improved from 0.91071 to 0.92321, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.9236 - loss: 0.2317
Epoch 59/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9061 - loss: 0.2136
Epoch 59: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9067 - loss: 0.2152
Epoch 60/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9328 - loss: 0.2023
Epoch 60: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9269 - loss: 0.2090
Epoch 61/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9004 - loss: 0.2308
Epoch 61: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9027 - loss: 0.2326
Epoch 62/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9213 - loss: 0.2005
Epoch 62: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9214 - loss: 0.2053
Epoch 63/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9394 - loss: 0.1787
Epoch 63: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9353 - loss: 0.1851
Epoch 64/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9138 - loss: 0.1971
Epoch 64: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9124 - loss: 0.1998
Epoch 65/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9175 - loss: 0.2030
Epoch 65: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9138 - loss: 0.2119
Epoch 66/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8938 - loss: 0.2392
Epoch 66: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8938 - loss: 0.2408
Epoch 67/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8836 - loss: 0.2606
Epoch 67: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8826 - loss: 0.2652
Epoch 68/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8869 - loss: 0.2588
Epoch 68: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8908 - loss: 0.2519
Epoch 69/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8815 - loss: 0.2808
Epoch 69: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8835 - loss: 0.2783
Epoch 70/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8913 - loss: 0.2819
Epoch 70: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8922 - loss: 0.2818
Epoch 71/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9099 - loss: 0.1930
Epoch 71: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9117 - loss: 0.1957
Epoch 72/150
```

```
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9019 - loss: 0.2294
Epoch 72: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9033 - loss: 0.2289
Epoch 73/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9185 - loss: 0.1968
Epoch 73: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9181 - loss: 0.1981
Epoch 74/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9037 - loss: 0.1991
Epoch 74: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9043 - loss: 0.1980
Epoch 75/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9120 - loss: 0.2136
Epoch 75: accuracy did not improve from 0.92321
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9124 - loss: 0.2163
Epoch 76/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9111 - loss: 0.2165
Epoch 76: accuracy improved from 0.92321 to 0.92500, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 11ms/step - accuracy: 0.9136 - loss: 0.2121
Epoch 77/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8843 - loss: 0.2382
Epoch 77: accuracy did not improve from 0.92500
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.8883 - loss: 0.2366
Epoch 78/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9197 - loss: 0.2185
Epoch 78: accuracy did not improve from 0.92500
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9168 - loss: 0.2223
Epoch 79/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9035 - loss: 0.2607
Epoch 79: accuracy did not improve from 0.92500
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9045 - loss: 0.2580
Epoch 80/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9443 - loss: 0.1771
Epoch 80: accuracy improved from 0.92500 to 0.93036, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9421 - loss: 0.1813
Epoch 81/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8986 - loss: 0.2355
Epoch 81: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8991 - loss: 0.2358
Epoch 82/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8831 - loss: 0.2380
Epoch 82: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.8843 - loss: 0.2388
Epoch 83/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9436 - loss: 0.2074
Epoch 83: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9408 - loss: 0.2127
Epoch 84/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9099 - loss: 0.2502
Epoch 84: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9129 - loss: 0.2410
Epoch 85/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9279 - loss: 0.1933
Epoch 85: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9272 - loss: 0.1933
Epoch 86/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9061 - loss: 0.2232
Epoch 86: accuracy did not improve from 0.93036
```

```
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9081 - loss: 0.2205
Epoch 87/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9486 - loss: 0.1661
Epoch 87: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9452 - loss: 0.1708
Epoch 88/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9198 - loss: 0.1856
Epoch 88: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9173 - loss: 0.1901
Epoch 89/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9245 - loss: 0.2079
Epoch 89: accuracy did not improve from 0.93036
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9231 - loss: 0.2086
Epoch 90/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9559 - loss: 0.1670
Epoch 90: accuracy improved from 0.93036 to 0.93929, saving model to best_model_w
eights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 1s 12ms/step - accuracy: 0.9533 - loss: 0.1683
Epoch 91/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9014 - loss: 0.2010
Epoch 91: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9067 - loss: 0.2046
Epoch 92/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9356 - loss: 0.1857
Epoch 92: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9345 - loss: 0.1872
Epoch 93/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9083 - loss: 0.2617
Epoch 93: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9046 - loss: 0.2631
Epoch 94/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9304 - loss: 0.2131
Epoch 94: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9306 - loss: 0.2110
Epoch 95/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8949 - loss: 0.2376
Epoch 95: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.8955 - loss: 0.2354
Epoch 96/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9063 - loss: 0.2043
Epoch 96: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9067 - loss: 0.2053
Epoch 97/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9055 - loss: 0.2292
Epoch 97: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9066 - loss: 0.2268
Epoch 98/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9147 - loss: 0.1930
Epoch 98: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9149 - loss: 0.1930
Epoch 99/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9255 - loss: 0.2074
Epoch 99: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9250 - loss: 0.2077
Epoch 100/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9160 - loss: 0.2260
Epoch 100: accuracy did not improve from 0.93929
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9157 - loss: 0.2255
Epoch 101/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9456 - loss: 0.1616
```

```
Epoch 101: accuracy improved from 0.93929 to 0.94464, saving model to best_model_
weights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.9453 - loss: 0.1645
Epoch 102/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9153 - loss: 0.2327
Epoch 102: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 1s 15ms/step - accuracy: 0.9154 - loss: 0.2328
Epoch 103/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9385 - loss: 0.1850
Epoch 103: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9368 - loss: 0.1870
Epoch 104/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9229 - loss: 0.2264
Epoch 104: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9221 - loss: 0.2218
Epoch 105/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9313 - loss: 0.1888
Epoch 105: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9331 - loss: 0.1862
Epoch 106/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9393 - loss: 0.1743
Epoch 106: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9366 - loss: 0.1761
Epoch 107/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9378 - loss: 0.1642
Epoch 107: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9366 - loss: 0.1715
Epoch 108/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9274 - loss: 0.1878
Epoch 108: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9258 - loss: 0.1907
Epoch 109/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9203 - loss: 0.1869
Epoch 109: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9202 - loss: 0.1892
Epoch 110/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9031 - loss: 0.2297
Epoch 110: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9046 - loss: 0.2304
Epoch 111/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9304 - loss: 0.1669
Epoch 111: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9278 - loss: 0.1713
Epoch 112/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9558 - loss: 0.1284
Epoch 112: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9526 - loss: 0.1355
Epoch 113/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9061 - loss: 0.2335
Epoch 113: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9065 - loss: 0.2321
Epoch 114/150
31/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9039 - loss: 0.1961
Epoch 114: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9051 - loss: 0.1954
Epoch 115/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9301 - loss: 0.1930
Epoch 115: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9307 - loss: 0.1924
Epoch 116/150
```

```
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9381 - loss: 0.1665
Epoch 116: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9378 - loss: 0.1670
Epoch 117/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9323 - loss: 0.1737
Epoch 117: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9322 - loss: 0.1744
Epoch 118/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9382 - loss: 0.1294
Epoch 118: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9379 - loss: 0.1305
Epoch 119/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9145 - loss: 0.2207
Epoch 119: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9146 - loss: 0.2204
Epoch 120/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9113 - loss: 0.1887
Epoch 120: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9137 - loss: 0.1887
Epoch 121/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9555 - loss: 0.1330
Epoch 121: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9542 - loss: 0.1340
Epoch 122/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9189 - loss: 0.1745
Epoch 122: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9191 - loss: 0.1749
Epoch 123/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9327 - loss: 0.1855
Epoch 123: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9332 - loss: 0.1840
Epoch 124/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9307 - loss: 0.1799
Epoch 124: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9302 - loss: 0.1803
Epoch 125/150
27/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9177 - loss: 0.1690
Epoch 125: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9150 - loss: 0.1820
Epoch 126/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9311 - loss: 0.1857
Epoch 126: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9296 - loss: 0.1894
Epoch 127/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9298 - loss: 0.1807
Epoch 127: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9292 - loss: 0.1820
Epoch 128/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9254 - loss: 0.2055
Epoch 128: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9258 - loss: 0.2036
Epoch 129/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step - accuracy: 0.9501 - loss: 0.1556
Epoch 129: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 1s 17ms/step - accuracy: 0.9412 - loss: 0.1664
Epoch 130/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9380 - loss: 0.1778
Epoch 130: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9364 - loss: 0.1844
Epoch 131/150
```

```
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9373 - loss: 0.2058
Epoch 131: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9371 - loss: 0.2059
Epoch 132/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9508 - loss: 0.1640
Epoch 132: accuracy did not improve from 0.94464
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9499 - loss: 0.1657
Epoch 133/150
23/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9353 - loss: 0.2148
Epoch 133: accuracy improved from 0.94464 to 0.95000, saving model to best_model_
weights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9412 - loss: 0.1987
Epoch 134/150
34/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9582 - loss: 0.1611
Epoch 134: accuracy improved from 0.95000 to 0.95536, saving model to best_model_
weights.keras
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 10ms/step - accuracy: 0.9580 - loss: 0.1601
Epoch 135/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9459 - loss: 0.1603
Epoch 135: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9461 - loss: 0.1599
Epoch 136/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9485 - loss: 0.1396
Epoch 136: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9469 - loss: 0.1431
Epoch 137/150
29/35 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9032 - loss: 0.2333
Epoch 137: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9078 - loss: 0.2242
Epoch 138/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9278 - loss: 0.1886
Epoch 138: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9279 - loss: 0.1864
Epoch 139/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9385 - loss: 0.1483
Epoch 139: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - accuracy: 0.9375 - loss: 0.1512
Epoch 140/150
30/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9133 - loss: 0.1932
Epoch 140: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9159 - loss: 0.1900
Epoch 141/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9373 - loss: 0.1649
Epoch 141: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9367 - loss: 0.1653
Epoch 142/150
23/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9229 - loss: 0.1969
Epoch 142: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9234 - loss: 0.1973
Epoch 143/150
25/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9191 - loss: 0.1501
Epoch 143: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9177 - loss: 0.1615
Epoch 144/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9265 - loss: 0.2180
Epoch 144: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9260 - loss: 0.2172
Epoch 145/150
26/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9228 - loss: 0.1978
Epoch 145: accuracy did not improve from 0.95536
```

```
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9209 - loss: 0.2022
Epoch 146/150
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9559 - loss: 0.1356
Epoch 146: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - accuracy: 0.9555 - loss: 0.1362
Epoch 147/150
24/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9378 - loss: 0.1639
Epoch 147: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9352 - loss: 0.1646
Epoch 148/150
32/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9415 - loss: 0.1363
Epoch 148: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9404 - loss: 0.1384
Epoch 149/150
28/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9425 - loss: 0.1320
Epoch 149: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - accuracy: 0.9443 - loss: 0.1307
Epoch 150/150
33/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9464 - loss: 0.1466
Epoch 150: accuracy did not improve from 0.95536
35/35 ━━━━━━━━━━━━━━━━━━━━ 0s 5ms/step - accuracy: 0.9464 - loss: 0.1472
5/5 ━━━━━━━━━━━━━━━━━━━━ 1s 137ms/step
              precision    recall  f1-score   support

           0       0.94      0.97      0.95        61
           1       0.97      0.95      0.96        79

    accuracy                           0.96       140
   macro avg       0.96      0.96      0.96       140
weighted avg       0.96      0.96      0.96       140

Accuracy: 0.9571428571428572
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 4ms/step
Predictions saved to final_output_predictions_vs_2.csv
```

In [ ]: