Modelo T sem data augmentation otimizer Adam

Neste modelo optamos por escolher o ResNet50 em vez do VGG16 utilizado nas aulas, visto que Escolhemos o ResNet50 por sua arquitetura avançada com blocos residuais, que permitem redes mais profundas e eficientes, resultando em melhor desempenho e capacidade de generalização em comparação ao VGG16, e assim conseguiriamos testar um novo modelo. Além disso, O ResNet50 parece ser mais suportado na comunidade de deep learning.

```
from tensorflow.keras.metrics import Metric
from tensorflow.keras import backend as K
import json
import os
import matplotlib.pyplot as plt
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import GlobalAveragePooling2D, Dropout,
Dense, BatchNormalization
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping,
ReduceLROnPlateau, CSVLogger
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.regularizers import l2
from tensorflow.keras.mixed_precision import set_global_policy
```

Usar float16 pode acelerar significativamente o treinamento do modelo e reduz também a memória necessária para o treino, sem perder grande precisão.

```
# MIX precision training -- facilita no treino!
set_global_policy('mixed_float16')

os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'

# CONSTANTES
BATCH_SIZE = 64
IMG_SIZE = 150
NUM_CLASSES = 10  # nº classes para identificar
NUM_EPOCHS = 60
LEARNING_RATE = 0.0001
DENSE_LAYERS = [1024, 512, 256, 128]
```

```
INFO:tensorflow:Mixed precision compatibility check (mixed_float16):
OK
Your GPU will likely run quickly with dtype policy mixed_float16 as it
has compute capability of at least 7.0. Your GPU: NVIDIA GeForce RTX
4070, compute capability 8.9
```

Dataset de treino, validação e teste

```
# Folders do dataset
train_dirs = ['./dataset/train/train1', './dataset/train/train2',
              './dataset/train/train3', './dataset/train/train5']
validation_dir = './dataset/validation'
test_dir = './dataset/test'
```

Data generators

```
train_datagen = ImageDataGenerator(rescale=1./255)
validation_datagen = ImageDataGenerator(rescale=1./255)
test_datagen = ImageDataGenerator(rescale=1./255)

# training generators
train_generators = [train_datagen.flow_from_directory(
    train_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical') for train_dir in train_dirs]

# Necessário para juntar os trainning generators and repeat
def combined_generator(generators):
    while True:
        for generator in generators:
            for batch in generator:
                yield batch


train_generator = combined_generator(train_generators)

# Validation e test generators
validation_generator = validation_datagen.flow_from_directory(
    validation_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical')

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

# Load the pre-trained ResNet50 model without the top layer and adjust
input shape
base_model = ResNet50(weights='imagenet', include_top=False,
                      input_shape=(IMG_SIZE, IMG_SIZE, 3))
```

```
Found 10000 images belonging to 10 classes.
Found 10000 images belonging to 10 classes.
Found 10000 images belonging to 10 classes.
Found 10000 images belonging to 10 classes.
Found 10000 images belonging to 10 classes.
Found 10000 images belonging to 10 classes.
```

UnFreeze layers

Para melhorar o modelo, foram descongeladas as últimas 50 camadas da rede base, permitindo que seus pesos sejam treinados. Isso ajuda o modelo a aprender melhor os padrões do novo conjunto de dados. Descongelar muitas camadas pode causar problemas, por isso mantivemos o número baixo. Com isso, equilibramos entre ajustar o modelo e evitar problemas de estabilidade. Essa técnica melhora o desempenho sem complicações.

(no modelo t com data augmentation testamos com 100 layers!)

```
# Descongelar camadas (nao meter valores demasiado altos)
for layer in base_model.layers[-50:]:
    layer.trainable = True
```

Métricas de Avaliação para Classificação Binária (Precision, Recall e F1 Score)

```
class Precision(Metric):
    def __init__(self, name='precision', **kwargs):
        super(Precision, self).__init__(name=name, **kwargs)
        self.true_positives = self.add_weight(name='tp',
initializer='zeros')
        self.predicted_positives = self.add_weight(
            name='pp', initializer='zeros')

    def update_state(self, y_true, y_pred, sample_weight=None):
        y_pred = K.round(y_pred)
        y_true = K.cast(y_true, 'float32')
        self.true_positives.assign_add(K.sum(y_true * y_pred))
        self.predicted_positives.assign_add(K.sum(y_pred))

    def result(self):
        return self.true_positives / (self.predicted_positives +
K.epsilon())

    def reset_states(self):
        self.true_positives.assign(0)
```

```
        self.predicted_positives.assign(0)


class Recall(Metric):
    def __init__(self, name='recall', **kwargs):
        super(Recall, self).__init__(name=name, **kwargs)
        self.true_positives = self.add_weight(name='tp',
initializer='zeros')
        self.actual_positives = self.add_weight(name='ap',
initializer='zeros')

    def update_state(self, y_true, y_pred, sample_weight=None):
        y_pred = K.round(y_pred)
        y_true = K.cast(y_true, 'float32')
        self.true_positives.assign_add(K.sum(y_true * y_pred))
        self.actual_positives.assign_add(K.sum(y_true))

    def result(self):
        return self.true_positives / (self.actual_positives +
K.epsilon())

    def reset_states(self):
        self.true_positives.assign(0)
        self.actual_positives.assign(0)


class F1Score(Metric):
    def __init__(self, name='f1_score', **kwargs):
        super(F1Score, self).__init__(name=name, **kwargs)
        self.precision = Precision()
        self.recall = Recall()

    def update_state(self, y_true, y_pred, sample_weight=None):
        self.precision.update_state(y_true, y_pred)
        self.recall.update_state(y_true, y_pred)

    def result(self):
        precision = self.precision.result()
        recall = self.recall.result()
        return 2 * ((precision * recall) / (precision + recall +
K.epsilon()))

    def reset_states(self):
        self.precision.reset_states()
        self.recall.reset_states()
```

Definir as layers do modelo

```
# Definir as layers do modelo com parametros ajustados para reduzir o
overfitting
model = Sequential([
    base_model,
    BatchNormalization(),
    GlobalAveragePooling2D(),
    # Increase model complexity
    Dense(DENSE_LAYERS[0], activation='relu',
kernel_regularizer=l2(0.03)),
    Dropout(0.5),  # High dropout rate for regularization
    BatchNormalization(),
    Dense(DENSE_LAYERS[1], activation='relu',
kernel_regularizer=l2(0.03)),
    Dropout(0.5),
    BatchNormalization(),
    Dense(DENSE_LAYERS[2], activation='relu',
kernel_regularizer=l2(0.03)),
    Dropout(0.5),
    Dense(DENSE_LAYERS[3], activation='relu',
kernel_regularizer=l2(0.03)),
    Dropout(0.5),
    BatchNormalization(),
    Dense(NUM_CLASSES, activation='softmax', dtype='float32')
])

# Compilar o modelo
model.compile(optimizer=Adam(learning_rate=LEARNING_RATE),
              loss='categorical_crossentropy',
              metrics=['accuracy', Precision(), Recall(), F1Score()])

model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 5, 5, 2048)        23587712

 batch_normalization (BatchN  (None, 5, 5, 2048)       8192
 ormalization)

 global_average_pooling2d (G  (None, 2048)             0
 lobalAveragePooling2D)

 dense (Dense)               (None, 1024)              2098176

 dropout (Dropout)           (None, 1024)              0
```

```
 batch_normalization_1 (Batc  (None, 1024)              4096
 hNormalization)

 dense_1 (Dense)              (None, 512)               524800

 dropout_1 (Dropout)          (None, 512)               0

 batch_normalization_2 (Batc  (None, 512)               2048
 hNormalization)

 dense_2 (Dense)              (None, 256)               131328

 dropout_2 (Dropout)          (None, 256)               0

 dense_3 (Dense)              (None, 128)               32896

 dropout_3 (Dropout)          (None, 128)               0

 batch_normalization_3 (Batc  (None, 128)               512
 hNormalization)

 dense_4 (Dense)              (None, 10)                1290

=================================================================
Total params: 26,391,050
Trainable params: 26,330,506
Non-trainable params: 60,544
_________________________________________________________________
```

Definir callbacks

```
# CALLBACKS
os.makedirs('logs', exist_ok=True)
checkpoint =
ModelCheckpoint(f'models/modelo_T_sem_data_augmentation_adam.keras',
                              monitor='val_accuracy', verbose=1,
save_best_only=True, mode='max')
early_stopping = EarlyStopping(
    monitor='val_loss', patience=10, restore_best_weights=True)  #
Increased patience
reduce_lr = ReduceLROnPlateau(
    monitor='val_loss', factor=0.2, patience=4, min_lr=1e-7,
verbose=1)  # More aggressive schedule
csv_logger = CSVLogger(
    f'logs/modelo_T_sem_data_augmentation_adam.csv', separator=',',
append=False)

# calcular passos por epoch
```

```
steps_per_epoch = sum([gen.samples // BATCH_SIZE for gen in
train_generators])
validation_steps = validation_generator.samples // BATCH_SIZE
```

Treino e avaliação do modelo

```
# calcular passos por epoch
# Treinar o modelo - Nao tirar os callbacks
history = model.fit(
    train_generator,
    steps_per_epoch=steps_per_epoch,
    epochs=NUM_EPOCHS,
    validation_data=validation_generator,
    validation_steps=validation_steps,
    callbacks=[checkpoint, early_stopping, reduce_lr, csv_logger]
)

# Avaliar o modelo no test generator
results = model.evaluate(test_generator)
loss, accuracy, precision, recall, f1_score = results[:5]
print(f"Test Loss: {loss}")
print(f"Test Accuracy: {accuracy}")
print(f"Test Precision: {precision}")
print(f"Test Recall: {recall}")
print(f"Test F1 Score: {f1_score}")
```

```
Epoch 1/60
624/624 [==============================] - ETA: 0s - loss: 52.7605 -
accuracy: 0.4382 - precision: 0.6945 - recall: 0.2654 - f1_score:
0.3840

c:\Users\USER\.conda\envs\py310\lib\site-packages\keras\engine\
training.py:2319: UserWarning: Metric Precision implements a
`reset_states()` method; rename it to `reset_state()` (without the
final "s"). The name `reset_states()` has been deprecated to improve
API consistency.
  m.reset_state()
c:\Users\USER\.conda\envs\py310\lib\site-packages\keras\engine\
training.py:2319: UserWarning: Metric Recall implements a
`reset_states()` method; rename it to `reset_state()` (without the
final "s"). The name `reset_states()` has been deprecated to improve
API consistency.
  m.reset_state()
c:\Users\USER\.conda\envs\py310\lib\site-packages\keras\engine\
training.py:2319: UserWarning: Metric F1Score implements a
`reset_states()` method; rename it to `reset_state()` (without the
final "s"). The name `reset_states()` has been deprecated to improve
```

```
API consistency.
  m.reset_state()


Epoch 1: val_accuracy improved from -inf to 0.10407, saving model to
models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 86s 109ms/step - loss:
52.7605 - accuracy: 0.4382 - precision: 0.6945 - recall: 0.2654 -
f1_score: 0.3840 - val_loss: 32.3975 - val_accuracy: 0.1041 -
val_precision: 0.1317 - val_recall: 0.0958 - val_f1_score: 0.1109 -
lr: 1.0000e-04
Epoch 2/60
624/624 [==============================] - ETA: 0s - loss: 16.9773 -
accuracy: 0.9075 - precision: 0.9548 - recall: 0.8296 - f1_score:
0.8878
Epoch 2: val_accuracy improved from 0.10407 to 0.85787, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 62s 100ms/step - loss:
16.9773 - accuracy: 0.9075 - precision: 0.9548 - recall: 0.8296 -
f1_score: 0.8878 - val_loss: 8.1945 - val_accuracy: 0.8579 -
val_precision: 0.9110 - val_recall: 0.8041 - val_f1_score: 0.8542 -
lr: 1.0000e-04
Epoch 3/60
624/624 [==============================] - ETA: 0s - loss: 4.2308 -
accuracy: 0.9617 - precision: 0.9730 - recall: 0.9409 - f1_score:
0.9567
Epoch 3: val_accuracy did not improve from 0.85787
624/624 [==============================] - 62s 100ms/step - loss:
4.2308 - accuracy: 0.9617 - precision: 0.9730 - recall: 0.9409 -
f1_score: 0.9567 - val_loss: 2.3749 - val_accuracy: 0.8475 -
val_precision: 0.8771 - val_recall: 0.8221 - val_f1_score: 0.8487 -
lr: 1.0000e-04
Epoch 4/60
624/624 [==============================] - ETA: 0s - loss: 1.1813 -
accuracy: 0.9715 - precision: 0.9777 - recall: 0.9623 - f1_score:
0.9699
Epoch 4: val_accuracy improved from 0.85787 to 0.87210, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 63s 101ms/step - loss:
1.1813 - accuracy: 0.9715 - precision: 0.9777 - recall: 0.9623 -
f1_score: 0.9699 - val_loss: 1.0297 - val_accuracy: 0.8721 -
val_precision: 0.8898 - val_recall: 0.8586 - val_f1_score: 0.8739 -
lr: 1.0000e-04
Epoch 5/60
624/624 [==============================] - ETA: 0s - loss: 0.5047 -
accuracy: 0.9727 - precision: 0.9775 - recall: 0.9662 - f1_score:
0.9719
Epoch 5: val_accuracy did not improve from 0.87210
624/624 [==============================] - 63s 101ms/step - loss:
0.5047 - accuracy: 0.9727 - precision: 0.9775 - recall: 0.9662 -
```

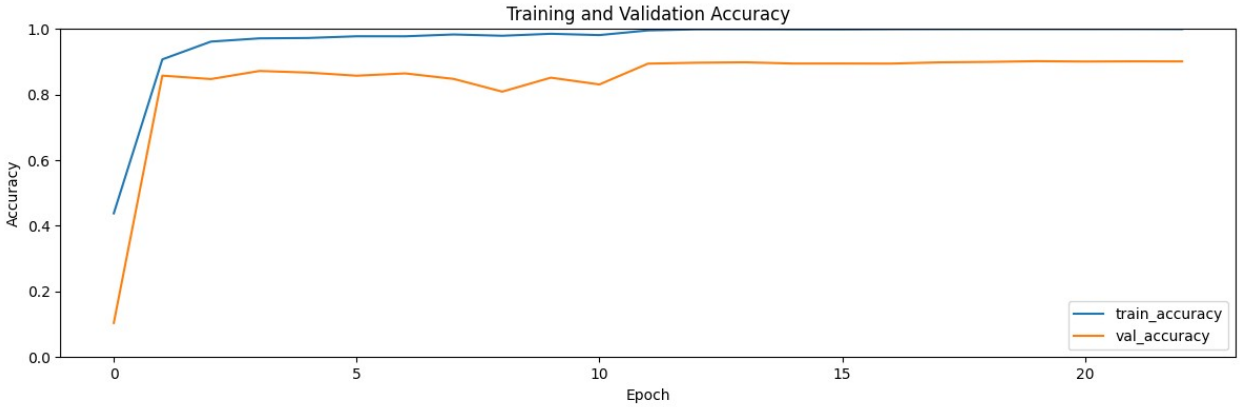
```
f1_score: 0.9719 - val_loss: 0.8315 - val_accuracy: 0.8672 -
val_precision: 0.8811 - val_recall: 0.8578 - val_f1_score: 0.8693 -
lr: 1.0000e-04
Epoch 6/60
624/624 [==============================] - ETA: 0s - loss: 0.3239 -
accuracy: 0.9779 - precision: 0.9816 - recall: 0.9734 - f1_score:
0.9775
Epoch 6: val_accuracy did not improve from 0.87210
624/624 [==============================] - 64s 102ms/step - loss:
0.3239 - accuracy: 0.9779 - precision: 0.9816 - recall: 0.9734 -
f1_score: 0.9775 - val_loss: 0.8259 - val_accuracy: 0.8577 -
val_precision: 0.8710 - val_recall: 0.8499 - val_f1_score: 0.8603 -
lr: 1.0000e-04
Epoch 7/60
624/624 [==============================] - ETA: 0s - loss: 0.2856 -
accuracy: 0.9778 - precision: 0.9814 - recall: 0.9739 - f1_score:
0.9776
Epoch 7: val_accuracy did not improve from 0.87210
624/624 [==============================] - 64s 102ms/step - loss:
0.2856 - accuracy: 0.9778 - precision: 0.9814 - recall: 0.9739 -
f1_score: 0.9776 - val_loss: 0.7655 - val_accuracy: 0.8647 -
val_precision: 0.8745 - val_recall: 0.8569 - val_f1_score: 0.8656 -
lr: 1.0000e-04
Epoch 8/60
624/624 [==============================] - ETA: 0s - loss: 0.2290 -
accuracy: 0.9834 - precision: 0.9859 - recall: 0.9802 - f1_score:
0.9831
Epoch 8: val_accuracy did not improve from 0.87210
624/624 [==============================] - 62s 99ms/step - loss:
0.2290 - accuracy: 0.9834 - precision: 0.9859 - recall: 0.9802 -
f1_score: 0.9831 - val_loss: 0.8915 - val_accuracy: 0.8479 -
val_precision: 0.8604 - val_recall: 0.8411 - val_f1_score: 0.8506 -
lr: 1.0000e-04
Epoch 9/60
624/624 [==============================] - ETA: 0s - loss: 0.2538 -
accuracy: 0.9794 - precision: 0.9826 - recall: 0.9758 - f1_score:
0.9792
Epoch 9: val_accuracy did not improve from 0.87210
624/624 [==============================] - 60s 97ms/step - loss:
0.2538 - accuracy: 0.9794 - precision: 0.9826 - recall: 0.9758 -
f1_score: 0.9792 - val_loss: 1.0544 - val_accuracy: 0.8091 -
val_precision: 0.8219 - val_recall: 0.8010 - val_f1_score: 0.8113 -
lr: 1.0000e-04
Epoch 10/60
624/624 [==============================] - ETA: 0s - loss: 0.2080 -
accuracy: 0.9854 - precision: 0.9875 - recall: 0.9835 - f1_score:
0.9855
Epoch 10: val_accuracy did not improve from 0.87210
624/624 [==============================] - 61s 98ms/step - loss:
```

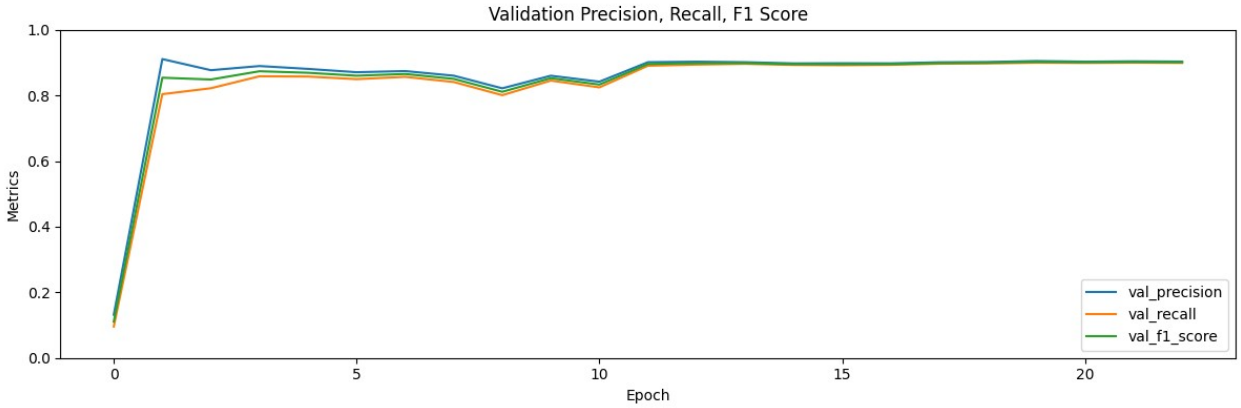
```
0.2080 - accuracy: 0.9854 - precision: 0.9875 - recall: 0.9835 -
f1_score: 0.9855 - val_loss: 0.8823 - val_accuracy: 0.8517 -
val_precision: 0.8603 - val_recall: 0.8455 - val_f1_score: 0.8528 -
lr: 1.0000e-04
Epoch 11/60
624/624 [==============================] - ETA: 0s - loss: 0.2397 -
accuracy: 0.9815 - precision: 0.9846 - recall: 0.9791 - f1_score:
0.9818
Epoch 11: val_accuracy did not improve from 0.87210

Epoch 11: ReduceLROnPlateau reducing learning rate to
1.9999999494757503e-05.
624/624 [==============================] - 66s 105ms/step - loss:
0.2397 - accuracy: 0.9815 - precision: 0.9846 - recall: 0.9791 -
f1_score: 0.9818 - val_loss: 0.9990 - val_accuracy: 0.8310 -
val_precision: 0.8421 - val_recall: 0.8247 - val_f1_score: 0.8333 -
lr: 1.0000e-04
Epoch 12/60
624/624 [==============================] - ETA: 0s - loss: 0.1360 -
accuracy: 0.9956 - precision: 0.9961 - recall: 0.9948 - f1_score:
0.9955
Epoch 12: val_accuracy improved from 0.87210 to 0.89443, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 64s 103ms/step - loss:
0.1360 - accuracy: 0.9956 - precision: 0.9961 - recall: 0.9948 -
f1_score: 0.9955 - val_loss: 0.6250 - val_accuracy: 0.8944 -
val_precision: 0.9015 - val_recall: 0.8906 - val_f1_score: 0.8960 -
lr: 2.0000e-05
Epoch 13/60
624/624 [==============================] - ETA: 0s - loss: 0.0781 -
accuracy: 0.9990 - precision: 0.9991 - recall: 0.9985 - f1_score:
0.9988
Epoch 13: val_accuracy improved from 0.89443 to 0.89724, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 63s 101ms/step - loss:
0.0781 - accuracy: 0.9990 - precision: 0.9991 - recall: 0.9985 -
f1_score: 0.9988 - val_loss: 0.6120 - val_accuracy: 0.8972 -
val_precision: 0.9031 - val_recall: 0.8941 - val_f1_score: 0.8986 -
lr: 2.0000e-05
Epoch 14/60
624/624 [==============================] - ETA: 0s - loss: 0.0587 -
accuracy: 0.9990 - precision: 0.9992 - recall: 0.9984 - f1_score:
0.9988
Epoch 14: val_accuracy improved from 0.89724 to 0.89854, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 61s 98ms/step - loss:
0.0587 - accuracy: 0.9990 - precision: 0.9992 - recall: 0.9984 -
f1_score: 0.9988 - val_loss: 0.6455 - val_accuracy: 0.8985 -
val_precision: 0.9015 - val_recall: 0.8964 - val_f1_score: 0.8990 -
```

```
lr: 2.0000e-05
Epoch 15/60
624/624 [==============================] - ETA: 0s - loss: 0.0527 -
accuracy: 0.9988 - precision: 0.9990 - recall: 0.9985 - f1_score:
0.9988
Epoch 15: val_accuracy did not improve from 0.89854
624/624 [==============================] - 62s 100ms/step - loss:
0.0527 - accuracy: 0.9988 - precision: 0.9990 - recall: 0.9985 -
f1_score: 0.9988 - val_loss: 0.6962 - val_accuracy: 0.8947 -
val_precision: 0.8978 - val_recall: 0.8930 - val_f1_score: 0.8954 -
lr: 2.0000e-05
Epoch 16/60
624/624 [==============================] - ETA: 0s - loss: 0.0533 -
accuracy: 0.9987 - precision: 0.9988 - recall: 0.9983 - f1_score:
0.9986
Epoch 16: val_accuracy did not improve from 0.89854
624/624 [==============================] - 64s 102ms/step - loss:
0.0533 - accuracy: 0.9987 - precision: 0.9988 - recall: 0.9983 -
f1_score: 0.9986 - val_loss: 0.6838 - val_accuracy: 0.8948 -
val_precision: 0.8983 - val_recall: 0.8923 - val_f1_score: 0.8953 -
lr: 2.0000e-05
Epoch 17/60
624/624 [==============================] - ETA: 0s - loss: 0.0428 -
accuracy: 0.9994 - precision: 0.9995 - recall: 0.9992 - f1_score:
0.9994
Epoch 17: val_accuracy did not improve from 0.89854

Epoch 17: ReduceLROnPlateau reducing learning rate to
3.999999898951501e-06.
624/624 [==============================] - 62s 100ms/step - loss:
0.0428 - accuracy: 0.9994 - precision: 0.9995 - recall: 0.9992 -
f1_score: 0.9994 - val_loss: 0.6887 - val_accuracy: 0.8945 -
val_precision: 0.8979 - val_recall: 0.8933 - val_f1_score: 0.8956 -
lr: 2.0000e-05
Epoch 18/60
624/624 [==============================] - ETA: 0s - loss: 0.0372 -
accuracy: 0.9997 - precision: 0.9997 - recall: 0.9996 - f1_score:
0.9997
Epoch 18: val_accuracy did not improve from 0.89854
624/624 [==============================] - 60s 97ms/step - loss:
0.0372 - accuracy: 0.9997 - precision: 0.9997 - recall: 0.9996 -
f1_score: 0.9997 - val_loss: 0.6674 - val_accuracy: 0.8983 -
val_precision: 0.9012 - val_recall: 0.8967 - val_f1_score: 0.8989 -
lr: 4.0000e-06
Epoch 19/60
624/624 [==============================] - ETA: 0s - loss: 0.0340 -
accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 - f1_score:
0.9998
Epoch 19: val_accuracy improved from 0.89854 to 0.89974, saving model
```

```
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 62s 99ms/step - loss:
0.0340 - accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 -
f1_score: 0.9998 - val_loss: 0.6616 - val_accuracy: 0.8997 -
val_precision: 0.9021 - val_recall: 0.8977 - val_f1_score: 0.8999 -
lr: 4.0000e-06
Epoch 20/60
624/624 [==============================] - ETA: 0s - loss: 0.0322 -
accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 - f1_score:
0.9998
Epoch 20: val_accuracy improved from 0.89974 to 0.90194, saving model
to models\modelo_T_sem_data_augmentation_adam.keras
624/624 [==============================] - 65s 103ms/step - loss:
0.0322 - accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 -
f1_score: 0.9998 - val_loss: 0.6464 - val_accuracy: 0.9019 -
val_precision: 0.9050 - val_recall: 0.8999 - val_f1_score: 0.9025 -
lr: 4.0000e-06
Epoch 21/60
624/624 [==============================] - ETA: 0s - loss: 0.0307 -
accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 - f1_score:
0.9998
Epoch 21: val_accuracy did not improve from 0.90194

Epoch 21: ReduceLROnPlateau reducing learning rate to
7.999999979801942e-07.
624/624 [==============================] - 63s 100ms/step - loss:
0.0307 - accuracy: 0.9998 - precision: 0.9999 - recall: 0.9997 -
f1_score: 0.9998 - val_loss: 0.6491 - val_accuracy: 0.9008 -
val_precision: 0.9031 - val_recall: 0.8992 - val_f1_score: 0.9012 -
lr: 4.0000e-06
Epoch 22/60
624/624 [==============================] - ETA: 0s - loss: 0.0295 -
accuracy: 0.9999 - precision: 1.0000 - recall: 0.9998 - f1_score:
0.9999
Epoch 22: val_accuracy did not improve from 0.90194
624/624 [==============================] - 61s 98ms/step - loss:
0.0295 - accuracy: 0.9999 - precision: 1.0000 - recall: 0.9998 -
f1_score: 0.9999 - val_loss: 0.6498 - val_accuracy: 0.9014 -
val_precision: 0.9041 - val_recall: 0.8999 - val_f1_score: 0.9020 -
lr: 8.0000e-07
Epoch 23/60
624/624 [==============================] - ETA: 0s - loss: 0.0294 -
accuracy: 0.9998 - precision: 0.9998 - recall: 0.9997 - f1_score:
0.9997
Epoch 23: val_accuracy did not improve from 0.90194
624/624 [==============================] - 62s 100ms/step - loss:
0.0294 - accuracy: 0.9998 - precision: 0.9998 - recall: 0.9997 -
f1_score: 0.9997 - val_loss: 0.6481 - val_accuracy: 0.9012 -
val_precision: 0.9033 - val_recall: 0.8993 - val_f1_score: 0.9013 -
```

```
lr: 8.0000e-07
157/157 [==============================] - 8s 52ms/step - loss: 0.6787
- accuracy: 0.8908 - precision: 0.8954 - recall: 0.8884 - f1_score:
0.8919
Test Loss: 0.6786661744117737
Test Accuracy: 0.8907999992370605
Test Precision: 0.8953840136528015
Test Recall: 0.8884000182151794
Test F1 Score: 0.8918782472610474

plt.figure(figsize=(12, 8))
plt.subplot(2, 1, 1)
plt.plot(history.history['accuracy'], label='train_accuracy')
plt.plot(history.history['val_accuracy'], label='val_accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.ylim([0, 1])
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(2, 1, 2)
plt.plot(history.history['val_precision'], label='val_precision')
plt.plot(history.history['val_recall'], label='val_recall')
plt.plot(history.history['val_f1_score'], label='val_f1_score')
plt.xlabel('Epoch')
plt.ylabel('Metrics')
plt.ylim([0, 1])
plt.legend(loc='lower right')
plt.title('Validation Precision, Recall, F1 Score')

plt.savefig(f'./plots/modelo_T_sem_data_augmentation_adam.png')
plt.tight_layout()
# plt.show()
```





Código para Extrair as Features

As features e as labels extraídos do modelo são concatenados e guardados em ficheiros NumPy.

```
# Function to extract features
import os
import numpy as np
from tensorflow.keras.models import Model
# Function to extract features
def extract_features_and_labels(generators, model):
    features = []
    labels = []
    total_samples = sum([gen.samples for gen in generators])
    batches_seen = 0

    for generator in generators:
        for inputs_batch, labels_batch in generator:
            features_batch = model.predict(inputs_batch)
            features.append(features_batch)
            labels.append(labels_batch)
            batches_seen += 1
            if batches_seen * generator.batch_size >= total_samples:
                break
```

```
    return np.concatenate(features), np.concatenate(labels)

# Extract features and labels
train_features, train_labels =
extract_features_and_labels(train_generators, feature_extractor_model)
validation_features, validation_labels =
extract_features_and_labels([validation_generator],
feature_extractor_model)
test_features, test_labels =
extract_features_and_labels([test_generator], feature_extractor_model)

# Ensure directories exist before saving
os.makedirs('features', exist_ok=True)
os.makedirs('labels', exist_ok=True)

# Save features and labels to files
np.save('features/modelo_T_sem_data_augmentation_train_features_adam.n
py', train_features)
np.save('labels/modelo_T_sem_data_augmentation_train_labels_adam.npy',
train_labels)
np.save('features/modelo_T_sem_data_augmentation_validation_features_a
dam.npy', validation_features)
np.save('labels/modelo_T_sem_data_augmentation_validation_labels_adam.
npy', validation_labels)
np.save('features/modelo_T_sem_data_augmentation_test_features_adam.np
y', test_features)
np.save('labels/modelo_T_sem_data_augmentation_test_labels_adam.npy',
test_labels)

print('Feature extraction completed.')
```

```
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 20ms/step
```

```
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 14ms/step
```

```
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
```

```
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 36ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 17ms/step
```

```
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 19ms/step
```

```
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 25ms/step
```

```
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 23ms/step
```

```
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
```

```
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 11ms/step
1/1 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 10ms/step
```

```
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 41ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
```

```
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 33ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 34ms/step
2/2 [==============================] - 0s 22ms/step
```

```
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
```

```
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 27ms/step
```

```
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 33ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 35ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 33ms/step
2/2 [==============================] - 0s 22ms/step
```

```
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 27ms/step
```

```
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 33ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
```

```
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 33ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 26ms/step
1/1 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 22ms/step
```

```
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 31ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 23ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 12ms/step
```

```
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 21ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 24ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 13ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 32ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 26ms/step
2/2 [==============================] - 0s 27ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 20ms/step
2/2 [==============================] - 0s 30ms/step
2/2 [==============================] - 0s 17ms/step
```

```
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 14ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 22ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 25ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 19ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 28ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 12ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 17ms/step
2/2 [==============================] - 0s 18ms/step
2/2 [==============================] - 0s 15ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 10ms/step
2/2 [==============================] - 0s 29ms/step
2/2 [==============================] - 0s 11ms/step
2/2 [==============================] - 0s 16ms/step
2/2 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 22ms/step
Feature extraction completed.
```