```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report, confusion_matrix
import numpy as np
```

```
train_dir = '/content/drive/MyDrive/split_minip/train'
val_dir = '/content/drive/MyDrive/split_minip/val'
test_dir = '/content/drive/MyDrive/split_minip/test'

batch_size = 32
img_size = (224, 224)

train_gen = ImageDataGenerator(rescale=1.0/255)
val_gen = ImageDataGenerator(rescale=1.0/255)
test_gen = ImageDataGenerator(rescale=1.0/255)

train_data = train_gen.flow_from_directory(train_dir, target_size=img_size, batch_size=batch_size, class_mode='categorical')
val_data = val_gen.flow_from_directory(val_dir, target_size=img_size, batch_size=batch_size, class_mode='categorical')
test_data = test_gen.flow_from_directory(test_dir, target_size=img_size, batch_size=batch_size, class_mode='categorical')
```

```
Found 5600 images belonging to 8 classes.
Found 1201 images belonging to 8 classes.
Found 1162 images belonging to 8 classes.
```

```
model = Sequential([
    Conv2D(96, (11, 11), strides=4, activation='relu', input_shape=(224, 224, 3)),
    MaxPooling2D((3, 3), strides=2),

    Conv2D(256, (5, 5), padding='same', activation='relu'),
    MaxPooling2D((3, 3), strides=2),

    Conv2D(384, (3, 3), padding='same', activation='relu'),
    Conv2D(384, (3, 3), padding='same', activation='relu'),
    Conv2D(256, (3, 3), padding='same', activation='relu'),
    MaxPooling2D((3, 3), strides=2),

    Flatten(),
    Dense(4096, activation='relu'),
    Dropout(0.5),
    Dense(4096, activation='relu'),
    Dropout(0.5),
    Dense(train_data.num_classes, activation='softmax')  # Output layer
])
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

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

```
from tensorflow.keras.callbacks import EarlyStopping
early_stopping = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
history = model.fit(
    train_data,
    validation_data=val_data,
    epochs=20,
    callbacks=[early_stopping]
)
```

```
Epoch 1/20
/usr/local/lib/python3.10/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:122: UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs
  self._warn_if_super_not_called()
175/175 ━━━━━━━━━━━━━━━━━━━━ 2123s 12s/step - accuracy: 0.1131 - loss: 2.5150 - val_accuracy: 0.1249 - val_loss: 2.0797
Epoch 2/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 28s 150ms/step - accuracy: 0.1276 - loss: 2.0799 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 3/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 40s 145ms/step - accuracy: 0.1231 - loss: 2.0798 - val_accuracy: 0.1257 - val_loss: 2.0795
Epoch 4/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 42s 152ms/step - accuracy: 0.1344 - loss: 2.0798 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 5/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 27s 150ms/step - accuracy: 0.1385 - loss: 2.0798 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 6/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 27s 151ms/step - accuracy: 0.1146 - loss: 2.0800 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 7/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 40s 148ms/step - accuracy: 0.1126 - loss: 2.0799 - val_accuracy: 0.1249 - val_loss: 2.0794
Epoch 8/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 41s 147ms/step - accuracy: 0.1176 - loss: 2.0797 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 9/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 26s 147ms/step - accuracy: 0.1332 - loss: 2.0796 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 10/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 27s 147ms/step - accuracy: 0.1277 - loss: 2.0797 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 11/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 27s 147ms/step - accuracy: 0.1317 - loss: 2.0796 - val_accuracy: 0.1249 - val_loss: 2.0795
Epoch 12/20
175/175 ━━━━━━━━━━━━━━━━━━━━ 41s 148ms/step - accuracy: 0.1261 - loss: 2.0796 - val_accuracy: 0.1249 - val_loss: 2.0795
```

```
val_loss, val_acc = model.evaluate(val_data)
print(f"Validation Loss: {val_loss}")
print(f'Validation accuracy: {val_acc}')
```

```
38/38 ━━━━━━━━━━━━━━━━━━━━ 4s 113ms/step - accuracy: 0.1187 - loss: 2.0796
Validation Loss: 2.0794482231140137
Validation accuracy: 0.12489592283964157
```

```
test_loss, test_acc = model.evaluate(test_data)
print(f"Test Loss: {test_loss}")
print(f"Test Accuracy: {test_acc}")
```

```
 1/37 ━━━━━━━━━━━━━━━━━━━━ 5s 154ms/step - accuracy: 0.0625 - loss: 2.0806/usr/local/lib/python3.10/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:122:
  self._warn_if_super_not_called()
37/37 ━━━━━━━━━━━━━━━━━━━━ 321s 9s/step - accuracy: 0.1280 - loss: 2.0796
Test Loss: 2.079446315765381
Test Accuracy: 0.12478485703468323
```

```
from sklearn.metrics import classification_report
import numpy as np
from tensorflow.keras.preprocessing.image import ImageDataGenerator

val_pred = model.predict(val_data)
test_pred = model.predict(test_data)

# Convert predictions and true labels to one-dimensional arrays
val_pred_classes = np.argmax(val_pred, axis=1)
test_pred_classes = np.argmax(test_pred, axis=1)
val_true_classes = val_data.classes
test_true_classes = test_data.classes

val_class_names = list(val_data.class_indices.keys())
test_class_names = list(test_data.class_indices.keys())

val_classification_report = classification_report(val_true_classes, val_pred_classes, target_names=val_class_names)
test_classification_report = classification_report(test_true_classes, test_pred_classes, target_names=test_class_names)

print('Validation Classification Report:')
print(val_classification_report)
print('Test Classification Report:')
print(test_classification_report)
```

```
38/38 ━━━━━━━━━━━━━━━━━━━━ 4s 107ms/step
37/37 ━━━━━━━━━━━━━━━━━━━━ 5s 123ms/step
Validation Classification Report:
              precision    recall  f1-score   support

          A+       0.00      0.00      0.00       150
          A-       0.00      0.00      0.00       150
         AB+       0.00      0.00      0.00       150
         AB-       0.12      1.00      0.22       150
          B+       0.00      0.00      0.00       150
          B-       0.00      0.00      0.00       151
          O+       0.00      0.00      0.00       150
          O-       0.00      0.00      0.00       150

    accuracy                           0.12      1201
   macro avg       0.02      0.12      0.03      1201
weighted avg       0.02      0.12      0.03      1201

Test Classification Report:
              precision    recall  f1-score   support

          A+       0.00      0.00      0.00       145
          A-       0.00      0.00      0.00       145
         AB+       0.00      0.00      0.00       145
         AB-       0.12      1.00      0.22       145
          B+       0.00      0.00      0.00       145
          B-       0.00      0.00      0.00       147
          O+       0.00      0.00      0.00       145
          O-       0.00      0.00      0.00       145

    accuracy                           0.12      1162
   macro avg       0.02      0.12      0.03      1162
weighted avg       0.02      0.12      0.03      1162

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1565: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predi
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
```