Import Req Lib

In [1]:
```
%matplotlib inline


import shutil
import random
import numpy as np
from warnings import filterwarnings
filterwarnings('ignore')

from tensorflow.keras import layers,regularizers, optimizers
from tensorflow.keras import models
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import LeakyReLU,Dense, Activation, Flatten, Dropout, BatchNormalization,Conv2D, MaxPooling2D
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import tensorflow as tf

import os
import time
import csv

from matplotlib import figure
```

Define 3 worker

In [2]:
```
# Set the number of threads
number_of_worker = 3
os.environ['OMP_NUM_THREADS'] = '3'  # OpenMP threads for parallelism
os.environ['TF_NUM_INTEROP_THREADS'] = '3'  # Threads for inter-operation parallelism
os.environ['TF_NUM_INTRAOP_THREADS'] = '3'  # Threads for intra-operation parallelism

# Confirm TensorFlow is using the specified number of threads
tf.config.threading.set_inter_op_parallelism_threads(number_of_worker)
tf.config.threading.set_intra_op_parallelism_threads(number_of_worker)
```

Train Val data Split

In [3]:
```
source_dir = r"C:\Users\nikhi\OneDrive\Desktop\Final Project\DATA\Convert_Audio_File_to_jpg_file"
target_dir = r'genres_train_val_split_data'
split_ratio = 0.8

def Train_Test_Split(source_dir,target_dir,split_ratio):
    # Define source and target directories
    train_dir = os.path.join(target_dir, 'train')
    val_dir = os.path.join(target_dir, 'val')

    # Create target directories if they don't exist
    os.makedirs(train_dir, exist_ok=True)
    os.makedirs(val_dir, exist_ok=True)

    # Get the list of class directories
    classes = [d for d in os.listdir(source_dir) if os.path.isdir(os.path.join(source_dir, d))]

    for class_name in classes:
        # Create class directories in train and val folders
        os.makedirs(os.path.join(train_dir, class_name), exist_ok=True)
        os.makedirs(os.path.join(val_dir, class_name), exist_ok=True)

        # Get list of images in the class directory
        class_dir = os.path.join(source_dir, class_name)
        images = [f for f in os.listdir(class_dir) if os.path.isfile(os.path.join(class_dir, f))]

        # Shuffle the images
        random.shuffle(images)

        # Compute the split point
        split_point = int(len(images) * split_ratio)

        # Split the images into training and validation sets
        train_images = images[:split_point]
        val_images = images[split_point:]
```

```python
        # Move the images to the respective directories
        for img in train_images:
            shutil.copy(os.path.join(class_dir, img), os.path.join(train_dir, class_name, img))

        for img in val_images:
            shutil.copy(os.path.join(class_dir, img), os.path.join(val_dir, class_name, img))

    print("Data split completed successfully!")
```

In [4]:
```python
Train_Test_Split(source_dir,target_dir,split_ratio)
```

```
Data split completed successfully!
```

Load the Data

In [5]:
```python
WIDTH = 64
HEIGHT = 64
BATCH_SIZE = 32
TRAIN_DIR=r'genres_train_val_split_data/train'
val_dir = r'genres_train_val_split_data/val'

# data prep
train_datagen = ImageDataGenerator(
    rescale=1./255.,validation_split=0.25)


train_generator = train_datagen.flow_from_directory(
    TRAIN_DIR,
    target_size=(HEIGHT, WIDTH),
        batch_size=BATCH_SIZE,
        class_mode='categorical')

validation_gen = train_datagen.flow_from_directory(
    val_dir,target_size = (HEIGHT,WIDTH),
    batch_size = BATCH_SIZE,
    class_mode = 'categorical'
 )
```

```
Found 800 images belonging to 10 classes.
Found 200 images belonging to 10 classes.
```

Model Architecture

In [6]:
```python
model = Sequential()
model.add(Conv2D(32, (3, 3), padding='same',
                 input_shape=(64,64,3)))
model.add(Activation('relu'))
model.add(Conv2D(64, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), padding='same'))
model.add(Activation('relu'))
model.add(Conv2D(64, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.5))
model.add(Conv2D(128, (3, 3), padding='same'))
model.add(Activation('relu'))
model.add(Conv2D(128, (3, 3)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.5))
model.add(Flatten())
model.add(Dense(512))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(10, activation='softmax'))
model.compile(optimizers.RMSprop(learning_rate=0.0005, decay=1e-6),loss="categorical_crossentropy",metrics=["accuracy"])
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 32)	896
activation (Activation)	(None, 64, 64, 32)	0
conv2d_1 (Conv2D)	(None, 62, 62, 64)	18,496
activation_1 (Activation)	(None, 62, 62, 64)	0
max_pooling2d (MaxPooling2D)	(None, 31, 31, 64)	0
dropout (Dropout)	(None, 31, 31, 64)	0
conv2d_2 (Conv2D)	(None, 31, 31, 64)	36,928
activation_2 (Activation)	(None, 31, 31, 64)	0
conv2d_3 (Conv2D)	(None, 29, 29, 64)	36,928
activation_3 (Activation)	(None, 29, 29, 64)	0
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 64)	0
dropout_1 (Dropout)	(None, 14, 14, 64)	0
conv2d_4 (Conv2D)	(None, 14, 14, 128)	73,856
activation_4 (Activation)	(None, 14, 14, 128)	0
conv2d_5 (Conv2D)	(None, 12, 12, 128)	147,584
activation_5 (Activation)	(None, 12, 12, 128)	0
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 128)	0
dropout_2 (Dropout)	(None, 6, 6, 128)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 512)	2,359,808
activation_6 (Activation)	(None, 512)	0
dropout_3 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 10)	5,130

Total params: 2,679,626 (10.22 MB)

Trainable params: 2,679,626 (10.22 MB)

Non-trainable params: 0 (0.00 B)

In [7]:

```
STEP_SIZE_TRAIN=train_generator.n//train_generator.batch_size
# Measure the execution time
start_time = time.time()

model.fit(train_generator,validation_data=validation_gen,epochs=200)

end_time = time.time()
elapsed_time = end_time - start_time
```

```
Epoch 1/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 17s 524ms/step - accuracy: 0.1172 - loss: 2.3200 - val_accuracy: 0.1000 - val_loss: 2.2990
Epoch 2/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 532ms/step - accuracy: 0.1234 - loss: 2.2973 - val_accuracy: 0.2100 - val_loss: 2.1400
Epoch 3/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 446ms/step - accuracy: 0.1766 - loss: 2.2264 - val_accuracy: 0.1950 - val_loss: 2.1511
Epoch 4/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 450ms/step - accuracy: 0.2260 - loss: 2.0911 - val_accuracy: 0.2800 - val_loss: 1.9925
Epoch 5/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.2132 - loss: 2.0567 - val_accuracy: 0.3100 - val_loss: 1.9550
Epoch 6/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 498ms/step - accuracy: 0.2552 - loss: 1.9979 - val_accuracy: 0.2850 - val_loss: 1.9534
Epoch 7/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 501ms/step - accuracy: 0.2653 - loss: 2.0255 - val_accuracy: 0.2500 - val_loss: 1.9111
Epoch 8/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 482ms/step - accuracy: 0.2902 - loss: 1.9696 - val_accuracy: 0.3850 - val_loss: 1.8503
Epoch 9/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 477ms/step - accuracy: 0.3088 - loss: 1.9267 - val_accuracy: 0.3300 - val_loss: 1.8287
Epoch 10/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 508ms/step - accuracy: 0.3537 - loss: 1.8062 - val_accuracy: 0.3650 - val_loss: 1.8139
Epoch 11/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 14s 565ms/step - accuracy: 0.3134 - loss: 1.8423 - val_accuracy: 0.3800 - val_loss: 1.7176
Epoch 12/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 460ms/step - accuracy: 0.3454 - loss: 1.7533 - val_accuracy: 0.3850 - val_loss: 1.6785
Epoch 13/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 460ms/step - accuracy: 0.4122 - loss: 1.7147 - val_accuracy: 0.3450 - val_loss: 1.7537
Epoch 14/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 472ms/step - accuracy: 0.3678 - loss: 1.6829 - val_accuracy: 0.3700 - val_loss: 1.7117
Epoch 15/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 495ms/step - accuracy: 0.4258 - loss: 1.6375 - val_accuracy: 0.4200 - val_loss: 1.5848
Epoch 16/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 464ms/step - accuracy: 0.4147 - loss: 1.6018 - val_accuracy: 0.3400 - val_loss: 1.8386
Epoch 17/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 474ms/step - accuracy: 0.4380 - loss: 1.5571 - val_accuracy: 0.4600 - val_loss: 1.4757
Epoch 18/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 469ms/step - accuracy: 0.4516 - loss: 1.4895 - val_accuracy: 0.3950 - val_loss: 1.6467
Epoch 19/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.4831 - loss: 1.3935 - val_accuracy: 0.3550 - val_loss: 1.6572
Epoch 20/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 450ms/step - accuracy: 0.4701 - loss: 1.4577 - val_accuracy: 0.4850 - val_loss: 1.4225
Epoch 21/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.5340 - loss: 1.3667 - val_accuracy: 0.4450 - val_loss: 1.4603
Epoch 22/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.5101 - loss: 1.3524 - val_accuracy: 0.4900 - val_loss: 1.3790
Epoch 23/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 476ms/step - accuracy: 0.5124 - loss: 1.3269 - val_accuracy: 0.4250 - val_loss: 1.4842
Epoch 24/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 454ms/step - accuracy: 0.5223 - loss: 1.3252 - val_accuracy: 0.4550 - val_loss: 1.4401
Epoch 25/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 430ms/step - accuracy: 0.5324 - loss: 1.3002 - val_accuracy: 0.4650 - val_loss: 1.3797
Epoch 26/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 480ms/step - accuracy: 0.5712 - loss: 1.1731 - val_accuracy: 0.4650 - val_loss: 1.4238
Epoch 27/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 442ms/step - accuracy: 0.5879 - loss: 1.2535 - val_accuracy: 0.5150 - val_loss: 1.4551
Epoch 28/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 471ms/step - accuracy: 0.5800 - loss: 1.1953 - val_accuracy: 0.4650 - val_loss: 1.4764
Epoch 29/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 439ms/step - accuracy: 0.6032 - loss: 1.0950 - val_accuracy: 0.4900 - val_loss: 1.3933
Epoch 30/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 432ms/step - accuracy: 0.6123 - loss: 1.0926 - val_accuracy: 0.5650 - val_loss: 1.3480
Epoch 31/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 463ms/step - accuracy: 0.6430 - loss: 0.9993 - val_accuracy: 0.5500 - val_loss: 1.2991
Epoch 32/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 433ms/step - accuracy: 0.6167 - loss: 1.0415 - val_accuracy: 0.5100 - val_loss: 1.4623
Epoch 33/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 447ms/step - accuracy: 0.5989 - loss: 1.0725 - val_accuracy: 0.5750 - val_loss: 1.2746
Epoch 34/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 435ms/step - accuracy: 0.6767 - loss: 0.9582 - val_accuracy: 0.5500 - val_loss: 1.2775
Epoch 35/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 452ms/step - accuracy: 0.6763 - loss: 0.9264 - val_accuracy: 0.5600 - val_loss: 1.3444
Epoch 36/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 438ms/step - accuracy: 0.6904 - loss: 0.8116 - val_accuracy: 0.4150 - val_loss: 1.8726
Epoch 37/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 449ms/step - accuracy: 0.6764 - loss: 0.9323 - val_accuracy: 0.5450 - val_loss: 1.4784
Epoch 38/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 447ms/step - accuracy: 0.7223 - loss: 0.8499 - val_accuracy: 0.5700 - val_loss: 1.3681
Epoch 39/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 450ms/step - accuracy: 0.7044 - loss: 0.8255 - val_accuracy: 0.4900 - val_loss: 1.5316
Epoch 40/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 442ms/step - accuracy: 0.7209 - loss: 0.7767 - val_accuracy: 0.5100 - val_loss: 1.3763
Epoch 41/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 456ms/step - accuracy: 0.7288 - loss: 0.8238 - val_accuracy: 0.5850 - val_loss: 1.2973
```

```
Epoch 42/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 438ms/step - accuracy: 0.7672 - loss: 0.7296 - val_accuracy: 0.4550 - val_loss: 1.7125
Epoch 43/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 445ms/step - accuracy: 0.7301 - loss: 0.7281 - val_accuracy: 0.4950 - val_loss: 1.5391
Epoch 44/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 458ms/step - accuracy: 0.7694 - loss: 0.6188 - val_accuracy: 0.5400 - val_loss: 1.4922
Epoch 45/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.7854 - loss: 0.6459 - val_accuracy: 0.5050 - val_loss: 1.6969
Epoch 46/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 502ms/step - accuracy: 0.7587 - loss: 0.7253 - val_accuracy: 0.5800 - val_loss: 1.4690
Epoch 47/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 486ms/step - accuracy: 0.8226 - loss: 0.5347 - val_accuracy: 0.5500 - val_loss: 1.4696
Epoch 48/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 494ms/step - accuracy: 0.8037 - loss: 0.5382 - val_accuracy: 0.5500 - val_loss: 1.4454
Epoch 49/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 464ms/step - accuracy: 0.8286 - loss: 0.4913 - val_accuracy: 0.5650 - val_loss: 1.3963
Epoch 50/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 10s 408ms/step - accuracy: 0.8572 - loss: 0.4218 - val_accuracy: 0.5700 - val_loss: 1.5492
Epoch 51/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 439ms/step - accuracy: 0.8326 - loss: 0.4634 - val_accuracy: 0.5250 - val_loss: 1.8723
Epoch 52/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 462ms/step - accuracy: 0.8203 - loss: 0.5689 - val_accuracy: 0.5500 - val_loss: 1.6347
Epoch 53/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 499ms/step - accuracy: 0.8558 - loss: 0.4413 - val_accuracy: 0.5250 - val_loss: 1.7110
Epoch 54/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 517ms/step - accuracy: 0.8577 - loss: 0.4649 - val_accuracy: 0.5750 - val_loss: 1.5534
Epoch 55/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 510ms/step - accuracy: 0.8622 - loss: 0.3862 - val_accuracy: 0.5800 - val_loss: 1.4429
Epoch 56/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 500ms/step - accuracy: 0.8415 - loss: 0.4130 - val_accuracy: 0.5600 - val_loss: 1.6298
Epoch 57/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 473ms/step - accuracy: 0.8331 - loss: 0.4591 - val_accuracy: 0.5550 - val_loss: 1.7618
Epoch 58/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 512ms/step - accuracy: 0.8447 - loss: 0.4063 - val_accuracy: 0.5600 - val_loss: 1.7022
Epoch 59/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 497ms/step - accuracy: 0.8745 - loss: 0.3720 - val_accuracy: 0.5500 - val_loss: 1.7594
Epoch 60/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 484ms/step - accuracy: 0.8637 - loss: 0.3368 - val_accuracy: 0.5600 - val_loss: 1.4790
Epoch 61/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 458ms/step - accuracy: 0.9153 - loss: 0.2569 - val_accuracy: 0.5450 - val_loss: 1.6113
Epoch 62/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 471ms/step - accuracy: 0.9336 - loss: 0.2270 - val_accuracy: 0.5000 - val_loss: 2.4138
Epoch 63/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 418ms/step - accuracy: 0.9275 - loss: 0.2432 - val_accuracy: 0.5550 - val_loss: 2.1046
Epoch 64/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 426ms/step - accuracy: 0.9020 - loss: 0.2548 - val_accuracy: 0.5650 - val_loss: 1.9601
Epoch 65/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 445ms/step - accuracy: 0.9056 - loss: 0.2923 - val_accuracy: 0.6000 - val_loss: 1.7826
Epoch 66/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 419ms/step - accuracy: 0.9148 - loss: 0.2506 - val_accuracy: 0.5550 - val_loss: 1.8186
Epoch 67/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 441ms/step - accuracy: 0.9171 - loss: 0.2525 - val_accuracy: 0.5900 - val_loss: 1.8137
Epoch 68/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 447ms/step - accuracy: 0.9299 - loss: 0.2053 - val_accuracy: 0.5550 - val_loss: 2.3692
Epoch 69/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 448ms/step - accuracy: 0.9197 - loss: 0.2721 - val_accuracy: 0.5500 - val_loss: 2.1923
Epoch 70/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 459ms/step - accuracy: 0.9512 - loss: 0.1595 - val_accuracy: 0.5800 - val_loss: 2.3808
Epoch 71/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 480ms/step - accuracy: 0.9184 - loss: 0.2301 - val_accuracy: 0.5500 - val_loss: 2.0233
Epoch 72/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 522ms/step - accuracy: 0.9068 - loss: 0.2479 - val_accuracy: 0.5700 - val_loss: 2.1736
Epoch 73/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 479ms/step - accuracy: 0.9345 - loss: 0.1671 - val_accuracy: 0.5800 - val_loss: 1.8012
Epoch 74/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9508 - loss: 0.1823 - val_accuracy: 0.5800 - val_loss: 1.9842
Epoch 75/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 450ms/step - accuracy: 0.9300 - loss: 0.2190 - val_accuracy: 0.5750 - val_loss: 2.5367
Epoch 76/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 464ms/step - accuracy: 0.9385 - loss: 0.1612 - val_accuracy: 0.5600 - val_loss: 2.3644
Epoch 77/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 486ms/step - accuracy: 0.9416 - loss: 0.1744 - val_accuracy: 0.6050 - val_loss: 2.2238
Epoch 78/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9580 - loss: 0.1393 - val_accuracy: 0.5850 - val_loss: 2.2327
Epoch 79/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9435 - loss: 0.1561 - val_accuracy: 0.6100 - val_loss: 2.2643
Epoch 80/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9536 - loss: 0.1949 - val_accuracy: 0.5950 - val_loss: 2.2301
Epoch 81/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9532 - loss: 0.1703 - val_accuracy: 0.5700 - val_loss: 2.2645
Epoch 82/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 459ms/step - accuracy: 0.9617 - loss: 0.0973 - val_accuracy: 0.5700 - val_loss: 2.4140
```

```
Epoch 83/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 450ms/step - accuracy: 0.9347 - loss: 0.1793 - val_accuracy: 0.5800 - val_loss: 2.2786
Epoch 84/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 449ms/step - accuracy: 0.9474 - loss: 0.1033 - val_accuracy: 0.5750 - val_loss: 2.3670
Epoch 85/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9548 - loss: 0.1371 - val_accuracy: 0.5600 - val_loss: 2.4043
Epoch 86/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 455ms/step - accuracy: 0.9691 - loss: 0.0769 - val_accuracy: 0.5900 - val_loss: 2.4306
Epoch 87/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 509ms/step - accuracy: 0.9638 - loss: 0.1350 - val_accuracy: 0.5850 - val_loss: 2.1009
Epoch 88/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 520ms/step - accuracy: 0.9666 - loss: 0.0947 - val_accuracy: 0.5700 - val_loss: 2.6385
Epoch 89/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 463ms/step - accuracy: 0.9648 - loss: 0.1116 - val_accuracy: 0.5850 - val_loss: 2.4659
Epoch 90/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 472ms/step - accuracy: 0.9513 - loss: 0.1856 - val_accuracy: 0.6000 - val_loss: 2.4104
Epoch 91/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 478ms/step - accuracy: 0.9636 - loss: 0.1290 - val_accuracy: 0.6000 - val_loss: 2.7544
Epoch 92/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9492 - loss: 0.1673 - val_accuracy: 0.5850 - val_loss: 2.5110
Epoch 93/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 458ms/step - accuracy: 0.9847 - loss: 0.0773 - val_accuracy: 0.5650 - val_loss: 2.4438
Epoch 94/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 446ms/step - accuracy: 0.9749 - loss: 0.0914 - val_accuracy: 0.5900 - val_loss: 2.8167
Epoch 95/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 463ms/step - accuracy: 0.9696 - loss: 0.0948 - val_accuracy: 0.5450 - val_loss: 2.9110
Epoch 96/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 485ms/step - accuracy: 0.9595 - loss: 0.1383 - val_accuracy: 0.5800 - val_loss: 2.2549
Epoch 97/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 20s 463ms/step - accuracy: 0.9758 - loss: 0.0668 - val_accuracy: 0.5550 - val_loss: 2.1897
Epoch 98/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 458ms/step - accuracy: 0.9547 - loss: 0.1388 - val_accuracy: 0.5750 - val_loss: 2.3984
Epoch 99/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 452ms/step - accuracy: 0.9875 - loss: 0.0453 - val_accuracy: 0.5850 - val_loss: 2.7185
Epoch 100/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 470ms/step - accuracy: 0.9522 - loss: 0.1185 - val_accuracy: 0.5650 - val_loss: 2.8550
Epoch 101/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9707 - loss: 0.0789 - val_accuracy: 0.5600 - val_loss: 2.9188
Epoch 102/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9429 - loss: 0.1708 - val_accuracy: 0.5800 - val_loss: 2.5333
Epoch 103/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 460ms/step - accuracy: 0.9541 - loss: 0.1467 - val_accuracy: 0.5650 - val_loss: 2.8877
Epoch 104/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9750 - loss: 0.0700 - val_accuracy: 0.5700 - val_loss: 2.9475
Epoch 105/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9723 - loss: 0.0933 - val_accuracy: 0.5550 - val_loss: 2.5764
Epoch 106/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.9811 - loss: 0.0727 - val_accuracy: 0.5650 - val_loss: 2.3499
Epoch 107/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9667 - loss: 0.0938 - val_accuracy: 0.5650 - val_loss: 3.0872
Epoch 108/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 436ms/step - accuracy: 0.9589 - loss: 0.1414 - val_accuracy: 0.5950 - val_loss: 3.2135
Epoch 109/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 445ms/step - accuracy: 0.9679 - loss: 0.1062 - val_accuracy: 0.5700 - val_loss: 2.9951
Epoch 110/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.9856 - loss: 0.0488 - val_accuracy: 0.5850 - val_loss: 3.3977
Epoch 111/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 436ms/step - accuracy: 0.9446 - loss: 0.1606 - val_accuracy: 0.5650 - val_loss: 3.1326
Epoch 112/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 476ms/step - accuracy: 0.9832 - loss: 0.0424 - val_accuracy: 0.5650 - val_loss: 2.7708
Epoch 113/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 451ms/step - accuracy: 0.9500 - loss: 0.1129 - val_accuracy: 0.5750 - val_loss: 2.9067
Epoch 114/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9897 - loss: 0.0355 - val_accuracy: 0.5950 - val_loss: 2.9419
Epoch 115/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 446ms/step - accuracy: 0.9769 - loss: 0.0644 - val_accuracy: 0.5600 - val_loss: 3.7960
Epoch 116/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 456ms/step - accuracy: 0.9770 - loss: 0.0871 - val_accuracy: 0.5700 - val_loss: 3.1759
Epoch 117/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 459ms/step - accuracy: 0.9854 - loss: 0.0627 - val_accuracy: 0.5700 - val_loss: 3.0052
Epoch 118/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 459ms/step - accuracy: 0.9806 - loss: 0.0613 - val_accuracy: 0.5750 - val_loss: 3.2695
Epoch 119/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 446ms/step - accuracy: 0.9679 - loss: 0.1012 - val_accuracy: 0.5650 - val_loss: 2.8748
Epoch 120/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 441ms/step - accuracy: 0.9692 - loss: 0.0690 - val_accuracy: 0.6050 - val_loss: 2.5103
Epoch 121/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 440ms/step - accuracy: 0.9825 - loss: 0.0566 - val_accuracy: 0.6000 - val_loss: 2.9525
Epoch 122/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 21s 467ms/step - accuracy: 0.9866 - loss: 0.0416 - val_accuracy: 0.5750 - val_loss: 3.5913
Epoch 123/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 506ms/step - accuracy: 0.9780 - loss: 0.1305 - val_accuracy: 0.5700 - val_loss: 2.7642
```

```
Epoch 124/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 500ms/step - accuracy: 0.9718 - loss: 0.0886 - val_accuracy: 0.5750 - val_loss: 3.1585
Epoch 125/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 458ms/step - accuracy: 0.9804 - loss: 0.0619 - val_accuracy: 0.5650 - val_loss: 2.9625
Epoch 126/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 461ms/step - accuracy: 0.9762 - loss: 0.0649 - val_accuracy: 0.5750 - val_loss: 3.0198
Epoch 127/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 448ms/step - accuracy: 0.9707 - loss: 0.1140 - val_accuracy: 0.5750 - val_loss: 2.7649
Epoch 128/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 425ms/step - accuracy: 0.9731 - loss: 0.0610 - val_accuracy: 0.6200 - val_loss: 2.5508
Epoch 129/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 441ms/step - accuracy: 0.9855 - loss: 0.0462 - val_accuracy: 0.5600 - val_loss: 3.7263
Epoch 130/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 431ms/step - accuracy: 0.9819 - loss: 0.0629 - val_accuracy: 0.5700 - val_loss: 3.2173
Epoch 131/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 431ms/step - accuracy: 0.9690 - loss: 0.0907 - val_accuracy: 0.5850 - val_loss: 3.3084
Epoch 132/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 444ms/step - accuracy: 0.9616 - loss: 0.0969 - val_accuracy: 0.5950 - val_loss: 2.6866
Epoch 133/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 462ms/step - accuracy: 0.9888 - loss: 0.0369 - val_accuracy: 0.5950 - val_loss: 2.8829
Epoch 134/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 447ms/step - accuracy: 0.9853 - loss: 0.0488 - val_accuracy: 0.5750 - val_loss: 2.7760
Epoch 135/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 469ms/step - accuracy: 0.9835 - loss: 0.0482 - val_accuracy: 0.5650 - val_loss: 3.7458
Epoch 136/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 472ms/step - accuracy: 0.9735 - loss: 0.0852 - val_accuracy: 0.5800 - val_loss: 2.7073
Epoch 137/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 484ms/step - accuracy: 0.9857 - loss: 0.0469 - val_accuracy: 0.5500 - val_loss: 3.1712
Epoch 138/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 481ms/step - accuracy: 0.9869 - loss: 0.0471 - val_accuracy: 0.5950 - val_loss: 2.9931
Epoch 139/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 495ms/step - accuracy: 0.9815 - loss: 0.0554 - val_accuracy: 0.5850 - val_loss: 2.8937
Epoch 140/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 484ms/step - accuracy: 0.9905 - loss: 0.0404 - val_accuracy: 0.5650 - val_loss: 3.5348
Epoch 141/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 474ms/step - accuracy: 0.9711 - loss: 0.1107 - val_accuracy: 0.5450 - val_loss: 2.9330
Epoch 142/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 456ms/step - accuracy: 0.9827 - loss: 0.0638 - val_accuracy: 0.5800 - val_loss: 2.7175
Epoch 143/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9693 - loss: 0.1044 - val_accuracy: 0.5900 - val_loss: 2.5806
Epoch 144/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 462ms/step - accuracy: 0.9912 - loss: 0.0224 - val_accuracy: 0.5650 - val_loss: 3.2506
Epoch 145/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 459ms/step - accuracy: 0.9614 - loss: 0.1111 - val_accuracy: 0.5550 - val_loss: 3.4666
Epoch 146/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 456ms/step - accuracy: 0.9848 - loss: 0.0526 - val_accuracy: 0.5650 - val_loss: 3.5513
Epoch 147/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 474ms/step - accuracy: 0.9844 - loss: 0.0788 - val_accuracy: 0.5750 - val_loss: 2.8310
Epoch 148/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 441ms/step - accuracy: 0.9880 - loss: 0.0294 - val_accuracy: 0.6150 - val_loss: 3.0327
Epoch 149/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 463ms/step - accuracy: 0.9798 - loss: 0.0653 - val_accuracy: 0.6100 - val_loss: 2.5929
Epoch 150/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 465ms/step - accuracy: 0.9912 - loss: 0.0427 - val_accuracy: 0.5750 - val_loss: 3.6527
Epoch 151/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 464ms/step - accuracy: 0.9785 - loss: 0.1017 - val_accuracy: 0.5600 - val_loss: 3.2230
Epoch 152/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9716 - loss: 0.0848 - val_accuracy: 0.5850 - val_loss: 3.4565
Epoch 153/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 452ms/step - accuracy: 0.9892 - loss: 0.0360 - val_accuracy: 0.5800 - val_loss: 3.7426
Epoch 154/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 486ms/step - accuracy: 0.9900 - loss: 0.0370 - val_accuracy: 0.5850 - val_loss: 3.6133
Epoch 155/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9752 - loss: 0.0612 - val_accuracy: 0.5850 - val_loss: 3.4285
Epoch 156/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 476ms/step - accuracy: 0.9927 - loss: 0.0228 - val_accuracy: 0.5850 - val_loss: 3.9586
Epoch 157/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 487ms/step - accuracy: 0.9683 - loss: 0.1098 - val_accuracy: 0.5600 - val_loss: 3.3739
Epoch 158/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9795 - loss: 0.0823 - val_accuracy: 0.5700 - val_loss: 3.5236
Epoch 159/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 470ms/step - accuracy: 0.9853 - loss: 0.0560 - val_accuracy: 0.5550 - val_loss: 3.9693
Epoch 160/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 507ms/step - accuracy: 0.9642 - loss: 0.1503 - val_accuracy: 0.5900 - val_loss: 3.8540
Epoch 161/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 473ms/step - accuracy: 0.9885 - loss: 0.0555 - val_accuracy: 0.5750 - val_loss: 3.8009
Epoch 162/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 477ms/step - accuracy: 0.9780 - loss: 0.0851 - val_accuracy: 0.6000 - val_loss: 3.0384
Epoch 163/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 491ms/step - accuracy: 0.9875 - loss: 0.0423 - val_accuracy: 0.6050 - val_loss: 3.1324
Epoch 164/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 472ms/step - accuracy: 0.9842 - loss: 0.0422 - val_accuracy: 0.5850 - val_loss: 3.5753
```

```
Epoch 165/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 509ms/step - accuracy: 0.9946 - loss: 0.0276 - val_accuracy: 0.5800 - val_loss: 4.1496
Epoch 166/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9832 - loss: 0.0464 - val_accuracy: 0.5450 - val_loss: 3.8821
Epoch 167/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 500ms/step - accuracy: 0.9846 - loss: 0.0371 - val_accuracy: 0.5700 - val_loss: 4.1487
Epoch 168/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 498ms/step - accuracy: 0.9689 - loss: 0.1025 - val_accuracy: 0.5750 - val_loss: 3.4159
Epoch 169/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 481ms/step - accuracy: 0.9792 - loss: 0.0696 - val_accuracy: 0.5800 - val_loss: 3.8318
Epoch 170/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 474ms/step - accuracy: 0.9868 - loss: 0.0269 - val_accuracy: 0.5700 - val_loss: 3.6744
Epoch 171/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 20s 456ms/step - accuracy: 0.9849 - loss: 0.0440 - val_accuracy: 0.5550 - val_loss: 3.8352
Epoch 172/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 463ms/step - accuracy: 0.9948 - loss: 0.0249 - val_accuracy: 0.5500 - val_loss: 4.1566
Epoch 173/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 13s 528ms/step - accuracy: 0.9784 - loss: 0.0598 - val_accuracy: 0.5800 - val_loss: 4.4257
Epoch 174/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 20s 489ms/step - accuracy: 0.9742 - loss: 0.0952 - val_accuracy: 0.5450 - val_loss: 4.0259
Epoch 175/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 472ms/step - accuracy: 0.9933 - loss: 0.0179 - val_accuracy: 0.5750 - val_loss: 3.7908
Epoch 176/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9774 - loss: 0.1114 - val_accuracy: 0.5750 - val_loss: 3.6599
Epoch 177/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 466ms/step - accuracy: 0.9853 - loss: 0.0346 - val_accuracy: 0.5600 - val_loss: 4.2077
Epoch 178/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 475ms/step - accuracy: 0.9845 - loss: 0.0378 - val_accuracy: 0.5700 - val_loss: 4.1975
Epoch 179/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 477ms/step - accuracy: 0.9792 - loss: 0.0861 - val_accuracy: 0.5750 - val_loss: 4.6767
Epoch 180/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 481ms/step - accuracy: 0.9858 - loss: 0.0325 - val_accuracy: 0.5700 - val_loss: 4.3123
Epoch 181/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 484ms/step - accuracy: 0.9795 - loss: 0.0905 - val_accuracy: 0.5850 - val_loss: 4.3368
Epoch 182/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 446ms/step - accuracy: 0.9910 - loss: 0.0388 - val_accuracy: 0.6150 - val_loss: 4.0462
Epoch 183/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 452ms/step - accuracy: 0.9808 - loss: 0.0731 - val_accuracy: 0.6000 - val_loss: 3.6067
Epoch 184/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 465ms/step - accuracy: 0.9906 - loss: 0.0207 - val_accuracy: 0.5900 - val_loss: 4.4340
Epoch 185/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 453ms/step - accuracy: 0.9866 - loss: 0.0524 - val_accuracy: 0.5950 - val_loss: 4.0456
Epoch 186/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 435ms/step - accuracy: 0.9843 - loss: 0.0646 - val_accuracy: 0.6100 - val_loss: 3.4074
Epoch 187/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 434ms/step - accuracy: 0.9754 - loss: 0.0781 - val_accuracy: 0.5750 - val_loss: 3.6424
Epoch 188/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 435ms/step - accuracy: 0.9921 - loss: 0.0286 - val_accuracy: 0.5900 - val_loss: 3.4255
Epoch 189/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 458ms/step - accuracy: 0.9945 - loss: 0.0156 - val_accuracy: 0.6150 - val_loss: 3.5043
Epoch 190/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 457ms/step - accuracy: 0.9765 - loss: 0.0669 - val_accuracy: 0.5900 - val_loss: 3.0471
Epoch 191/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 441ms/step - accuracy: 0.9822 - loss: 0.0497 - val_accuracy: 0.5900 - val_loss: 3.5785
Epoch 192/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 432ms/step - accuracy: 0.9866 - loss: 0.0376 - val_accuracy: 0.6000 - val_loss: 3.5430
Epoch 193/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 447ms/step - accuracy: 0.9905 - loss: 0.0331 - val_accuracy: 0.5700 - val_loss: 3.9939
Epoch 194/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 21s 459ms/step - accuracy: 0.9802 - loss: 0.0612 - val_accuracy: 0.5750 - val_loss: 4.2690
Epoch 195/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 458ms/step - accuracy: 0.9828 - loss: 0.0629 - val_accuracy: 0.5700 - val_loss: 3.7037
Epoch 196/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 457ms/step - accuracy: 0.9917 - loss: 0.0292 - val_accuracy: 0.6000 - val_loss: 3.6349
Epoch 197/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 11s 457ms/step - accuracy: 0.9881 - loss: 0.0378 - val_accuracy: 0.5800 - val_loss: 3.7407
Epoch 198/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 460ms/step - accuracy: 0.9869 - loss: 0.0346 - val_accuracy: 0.5800 - val_loss: 3.9399
Epoch 199/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 461ms/step - accuracy: 0.9959 - loss: 0.0230 - val_accuracy: 0.5700 - val_loss: 4.6127
Epoch 200/200
25/25 ━━━━━━━━━━━━━━━━━━━━ 12s 467ms/step - accuracy: 0.9954 - loss: 0.0106 - val_accuracy: 0.5800 - val_loss: 4.2340
```

In [8]:
```python
print(f"Execution time: {elapsed_time:.2f} seconds")
```

```
Execution time: 2385.14 seconds
```

In [9]:
```python
def append_core_data(score_path, num_cores, elapsed_time):
    # Check if the file already exists
    file_exists = os.path.exists(score_path)

    # Open the file in append mode
    with open(score_path, mode='a', newline='') as file:
```

```python
        writer = csv.writer(file)

        # If the file is new, write the header
        if not file_exists:
            writer.writerow(["Number of Cores", "Elapsed Time"])

        # Write the new data
        writer.writerow([num_cores, elapsed_time])
```

In [10]:
```python
score_path = r"C:\Users\nikhi\OneDrive\Desktop\Final Project\DEEP LEARNING WITH HPSC\core_data.txt"
append_core_data(score_path, number_of_worker, elapsed_time)
```