Accuracy with original number of epochs (2):

```
  self._warn_if_super_not_called()
Epoch 1/2
489/624 ━━━━━━━━━━━━━━━━━━━━ 39s 295ms/step - accuracy: 0.5176 - loss: 0.7415C:\Users\001053403\AppData\Roaming\Python\P
ython312\site-packages\PIL\TiffImagePlugin.py:950: UserWarning: Truncated File Read
  warnings.warn(str(msg))
624/624 ━━━━━━━━━━━━━━━━━━━━ 0s 291ms/step - accuracy: 0.5233 - loss: 0.7317C:\Users\001053403\AppData\Roaming\Python\Py
thon312\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:121: UserWarning: Your `PyDataset` class sh
ould call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `workers`, `use_multiprocessing`, `max
_queue_size`. Do not pass these arguments to `fit()`, as they will be ignored.
  self._warn_if_super_not_called()
624/624 ━━━━━━━━━━━━━━━━━━━━ 193s 307ms/step - accuracy: 0.5233 - loss: 0.7316 - val_accuracy: 0.5863 - val_loss: 0.6737
Epoch 2/2
  1/624 ━━━━━━━━━━━━━━━━━━━━ 2:10 209ms/step - accuracy: 0.6875 - loss: 0.6488C:\Users\001053403\AppData\Roaming\Python\
Python312\site-packages\keras\src\trainers\epoch_iterator.py:107: UserWarning: Your input ran out of data; interrupting
training. Make sure that your dataset or generator can generate at least `steps_per_epoch * epochs` batches. You may nee
d to use the `.repeat()` function when building your dataset.
  self._interrupted_warning()
624/624 ━━━━━━━━━━━━━━━━━━━━ 9s 15ms/step - accuracy: 0.6875 - loss: 0.6488 - val_accuracy: 0.5811 - val_loss: 0.6751
157/157 ━━━━━━━━━━━━━━━━━━━━ 9s 58ms/step - accuracy: 0.5829 - loss: 0.6755
Validation Loss: 0.6749
Validation Accuracy: 0.5816
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This file
 format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')` or
 `keras.saving.save_model(model, 'my_model.keras')`.
Press any key to continue . . .
```

Accuracy with increased number of epochs (5):

```
Epoch 1/5
114/624 ━━━━━━━━━━━━━━━━━━━━ 2:12 260ms/step - accuracy: 0.4911 - loss: 1.0136C:\Users\001053403\AppData\Roaming\Python\
Python312\site-packages\PIL\TiffImagePlugin.py:950: UserWarning: Truncated File Read
  warnings.warn(str(msg))
624/624 ━━━━━━━━━━━━━━━━━━━━ 0s 267ms/step - accuracy: 0.5328 - loss: 0.7701C:\Users\001053403\AppData\Roaming\Python\Py
thon312\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:121: UserWarning: Your `PyDataset` class sh
ould call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `workers`, `use_multiprocessing`, `max
_queue_size`. Do not pass these arguments to `fit()`, as they will be ignored.
  self._warn_if_super_not_called()
624/624 ━━━━━━━━━━━━━━━━━━━━ 176s 281ms/step - accuracy: 0.5329 - loss: 0.7700 - val_accuracy: 0.6196 - val_loss: 0.6330
Epoch 2/5
  1/624 ━━━━━━━━━━━━━━━━━━━━ 2:06 202ms/step - accuracy: 0.6250 - loss: 0.6944C:\Users\001053403\AppData\Roaming\Python\
Python312\site-packages\keras\src\trainers\epoch_iterator.py:107: UserWarning: Your input ran out of data; interrupting
training. Make sure that your dataset or generator can generate at least `steps_per_epoch * epochs` batches. You may nee
d to use the `.repeat()` function when building your dataset.
  self._interrupted_warning()
624/624 ━━━━━━━━━━━━━━━━━━━━ 9s 14ms/step - accuracy: 0.6250 - loss: 0.6944 - val_accuracy: 0.6270 - val_loss: 0.6364
Epoch 3/5
624/624 ━━━━━━━━━━━━━━━━━━━━ 175s 280ms/step - accuracy: 0.6262 - loss: 0.6477 - val_accuracy: 0.6649 - val_loss: 0.5926
Epoch 4/5
624/624 ━━━━━━━━━━━━━━━━━━━━ 9s 14ms/step - accuracy: 0.7188 - loss: 0.6366 - val_accuracy: 0.6633 - val_loss: 0.5951
Epoch 5/5
624/624 ━━━━━━━━━━━━━━━━━━━━ 174s 280ms/step - accuracy: 0.6793 - loss: 0.5955 - val_accuracy: 0.7584 - val_loss: 0.4986
157/157 ━━━━━━━━━━━━━━━━━━━━ 9s 56ms/step - accuracy: 0.7534 - loss: 0.5013
Validation Loss: 0.4990
Validation Accuracy: 0.7580
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This file
 format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')` or
 `keras.saving.save_model(model, 'my_model.keras')`.
```

The accuracy of the model improved and could most likely be further improved by adding a few more epochs.