期末專題計畫書

U11116035 張晉源

1 題目

臉型辨識的影像處理

2 Introduction

2.1 動機

隨著科技的進步，影像辨識已經成為一項有趣且實用的技術。此計劃希望透過專題學習基本的影像處理技術，並嘗試應用於簡單的臉型辨識。

2.2 目的或做完此專題，要解決的問題

此計劃的目的是設計一個簡單的功能，能夠辨識圖片中的臉型。希望透過學習這項技術，了解基本的影像處理過程，並協助我判斷出我平常難以分辨的臉型

3 文獻

1. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

   https://medium.com/@fearless_fusion_snake_755/%E8%AB%96%E6%96%87%E9%96%B1%E8%AE%80-swin-transformer-hierarchical-vision-transformer-using-shifted-windows-e1c822a0811b

2. A Comparison of CNN and Transformer in Continual Learning JINGWEN FU：

   https://kth.diva-portal.org/smash/get/diva2:1820229/FULLTEXT01.pdf

4 我的方法

由於 CNN 方法較為花時間，我發現了基於 CNN 概念而生的 Swin Transformer，由於有 Self-Attention 的機制，相較於 CNN 的局部分析，此方法在靈活性及效率上都比 CNN 更加具有優勢，在影像處理的任務中會有更好的結果

5 實驗結果

1. 正確率（accuracy）：約 70% 以上

Evaluation on test set

```
outputs = trainer.predict(test_ds)
print(outputs.metrics)
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

y_true = outputs.label_ids
y_pred = outputs.predictions.argmax(1)
labels = train_ds.features['label'].names
```

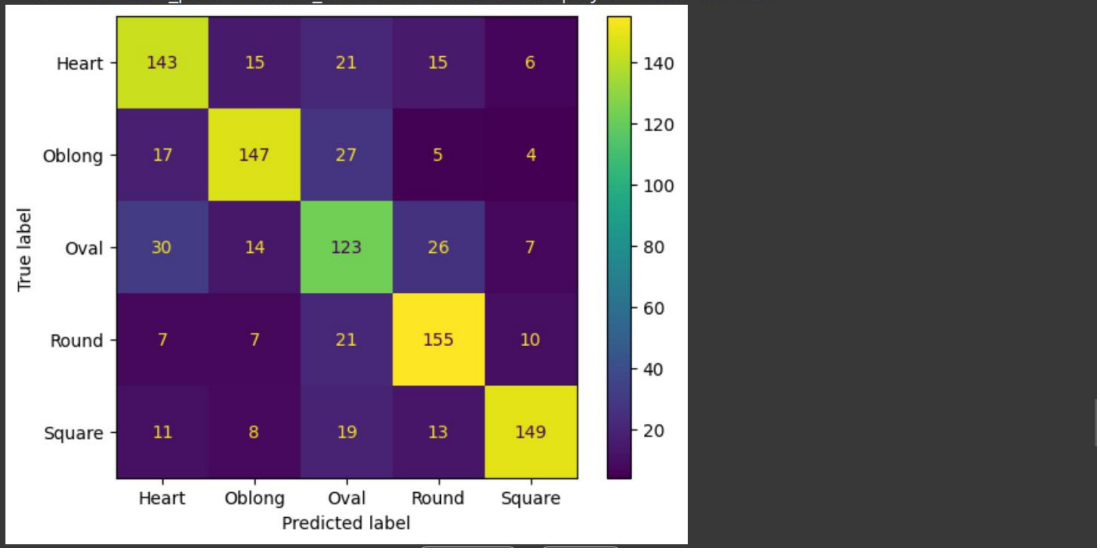
```
{'test_loss': 0.7590962052345276, 'test_accuracy': 0.72, 'test_runtime': 443.5643, 'test_samples_per_second': 2.254, 'test_steps_per_second': 0.072}
```

2. 混淆矩陣（Confusion matrix）：

Confusion matrix

```
# Confusion matrix
# 顯示混淆矩陣
cm = confusion_matrix(y_true, y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=labels)
disp.plot()
```

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x7899e2667490>
```



3. 訓練和驗證準確率（Training and Validation Accuracy）：

```
[ ]  1 # 禁用 Weights & Biases
     2 os.environ["WANDB_DISABLED"] = "true"
     3 os.environ["WANDB_MODE"] = "offline"
     4
     5 # 處理可能出現的圖片載入錯誤
     6 from PIL import ImageFile
     7 ImageFile.LOAD_TRUNCATED_IMAGES = True
     8
     9 # 開始訓練模型
    10 trainer.train()
```

```
wandb: WARNING The `run_name` is currently set to the same value as `TrainingArguments.output_dir`. If th
wandb: Using wandb-core as the SDK backend.  Please refer to https://wandb.me/wandb-core for more informa
Tracking run with wandb version 0.19.1
W&B syncing is set to `offline` in this directory.
Run `wandb online` or set WANDB_MODE=online to enable cloud syncing.
```

[2260/2260 1:02:52, Epoch 20/20]

Epoch	Training Loss	Validation Loss	Accuracy
1	No log	1.435288	0.342500
2	No log	1.232904	0.510000
3	No log	1.135982	0.560000
4	No log	1.046277	0.612500
5	1.277300	1.047763	0.567500
6	1.277300	0.942834	0.630000
7	1.277300	0.966280	0.637500
8	1.277300	0.905068	0.680000
9	0.888500	0.835907	0.710000
10	0.888500	0.824327	0.707500
11	0.888500	0.831144	0.702500
12	0.888500	0.791054	0.712500
13	0.888500	0.767695	0.720000
14	0.687800	0.791414	0.710000
15	0.687800	0.773661	0.717500
16	0.687800	0.754415	0.732500
17	0.687800	0.747798	0.735000
18	0.581200	0.740930	0.732500
19	0.581200	0.735352	0.735000
20	0.581200	0.735347	0.730000

```
TrainOutput(global_step=2260, training_loss=0.8203253416888482, metrics={'train_runtime': 3817.2663,
'train_samples_per_second': 18.862, 'train_steps_per_second': 0.592, 'total_flos': 1.789781226799104e+18,
'train_loss': 0.8203253416888482, 'epoch': 20.0})
```

4. 測試集上分類準確度：

Evaluation on test set

```
1 #  評估模型在測試集上的表現
2 outputs  =  trainer.predict(test_ds)
3 print(outputs.metrics)
4 from  sklearn.metrics  import  confusion_matrix,  ConfusionMatrixDisplay
5
6 #  取得真實標籤和預測標籤
7 y_true  =  outputs.label_ids
8 y_pred  =  outputs.predictions.argmax(1)
9 labels  =  train_ds.features['label'].names
```

```
{'test_loss': 0.7661802768707275, 'test_accuracy': 0.717, 'test_runtime': 480.7879, 'test_samples_per_second': 2.08, 'test_steps_per_second': 0.067}
```

6 參考資料

1. 此課堂之 eeclass 教材

2. 資料集來源：https://www.kaggle.com/datasets/niten19/face-shape-dataset/data