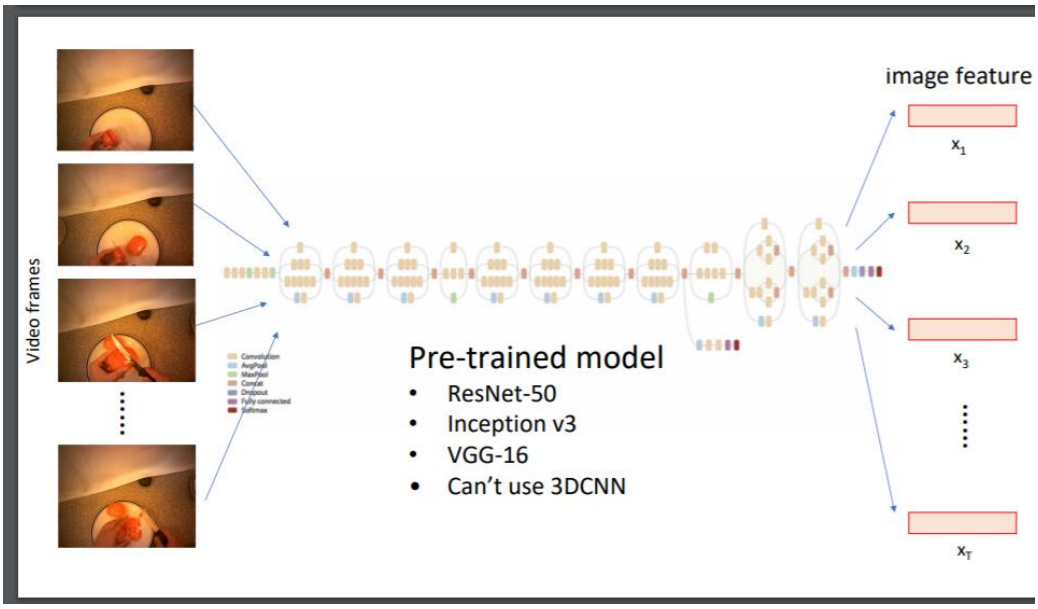
Name: 楊耀程 Dep.:電機四 Student ID:B03901161

[Problem1]

1. (5%) Describe your strategies of extracting CNN-based video features, training the model and other implementation details.

   整體架構 (同作業說明，pretrain model 為 inception v3)
   Inception v3 有 finetune mixed_7b 之後的層，finetune inception 的 learning rate 為 5e-5，後面 fc 的 learning rate 為 1e-3。

   Input 為影片 downsample 到 16 個 frame，此 16 個 frame 經過 inception v3 後取得的 feature，做 **average pooling**，原本也有考慮做 concatenate，但是 16 個 frame 串接實在太多了，會造成運算上的負擔。



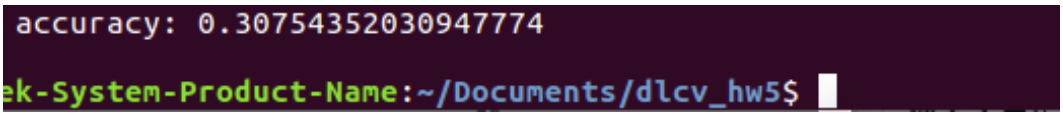
FC 架構(除了 pretrain model 之外)

```
==========================
actRecogFC_classifier(
  (avgpool1d_2): AvgPool1d(kernel_size=(16,), stride=(16,), padding=(0,), ceil_mode=False, count_include_pad=True)
  (fc1): Sequential(
    (0): Linear(in_features=2048, out_features=4096)
    (1): ReLU()
  )
  (fc2): Sequential(
    (0): Linear(in_features=4096, out_features=4096)
    (1): ReLU()
  )
  (fc3): Sequential(
    (0): Linear(in_features=4096, out_features=11)
  )
)
```

Average pooling，以及兩層的 fully connected，還有 softmax 層。

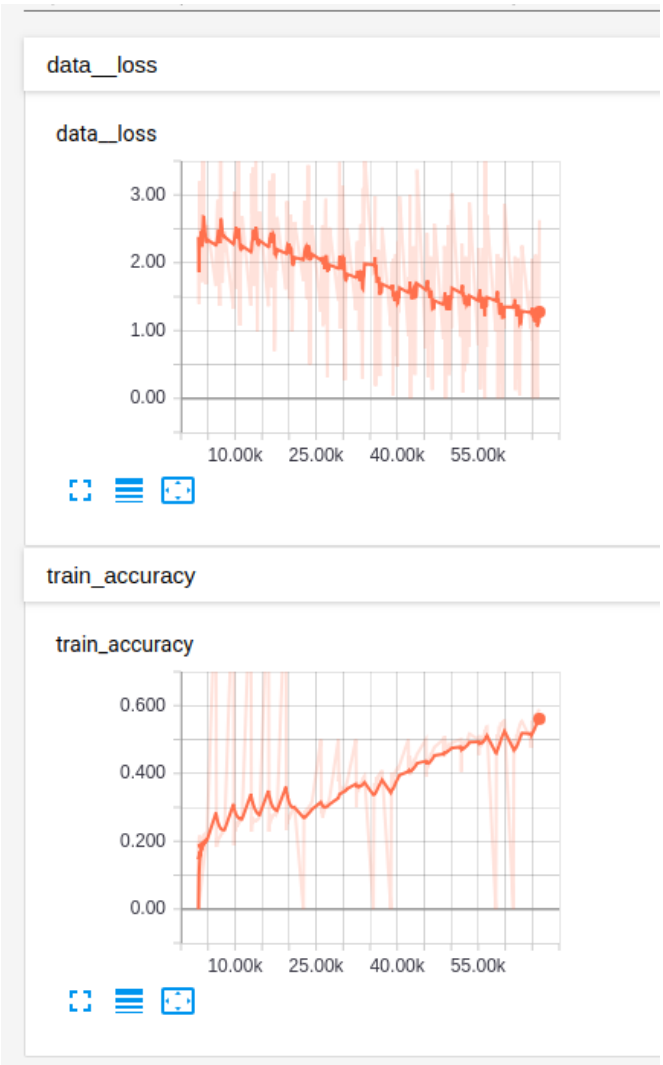
2. (15%) Report your video recognition performance using CNN-based video features and plot the learning curve of your model.

Validation accuracy:



(先前在跑 21 個 class 時，有 train 到 validation 40%準確度，後來改成 11 個 class，時間有限只跑 20 個 epoch，如果時間更長，validation 準確度應該有機會超過 40%一些)

Learning curve:



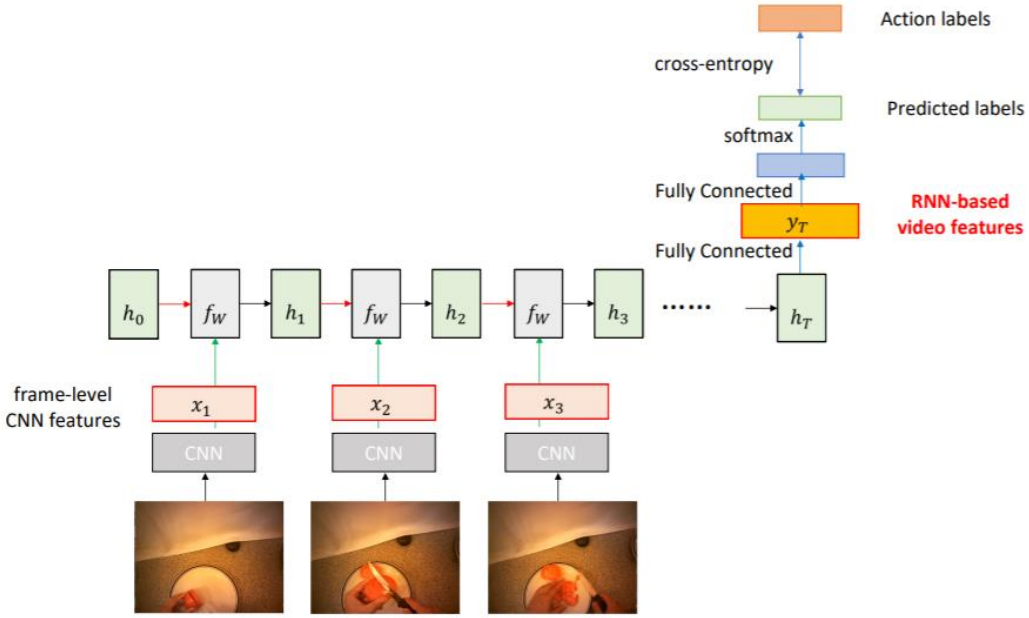
[Problem2]

1. (5%) Describe your RNN models and implementation details for action recognition.

整體架構 (同作業說明，pretrain model 為 inception v3)

Model input: 為不同長度的影片作為 input，使用 batchsize = 1，比較容易處理不同長度的問題。此外，為避免記憶體不夠，有設定影片長度上限為 500，如果

長度超過 500 就直接 trim 掉。我直接將不同長度的影片，一一給進 model 作為 input。
(我有嘗試過固定 frame 的做法，同第一小題，通通 down sample 到 16 個 frame，但效果不好，並無法過 baseline，準確度也遠低於使用不同長度的 RNN，推測是影片的長度相差太多，如果用固定 frame 會少掉很多資訊)

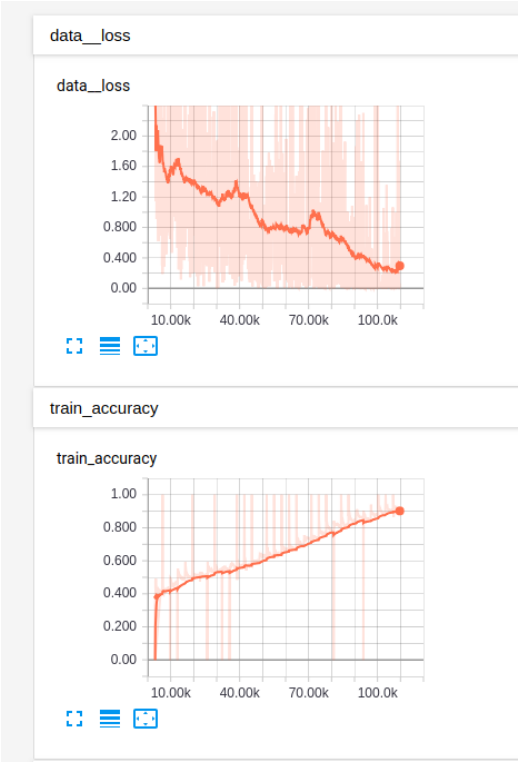


RNN 架構(除了 pretrain model 之外)

```
actRecog_RNN_VarLen_classifier(
  (gru): GRU(2048, 256, batch_first=True, bidirectional=True)
  (fc): Linear(in_features=512, out_features=11)
)
```

為雙向的單層 GRU，hidden unit 為 256。

Learning curve:

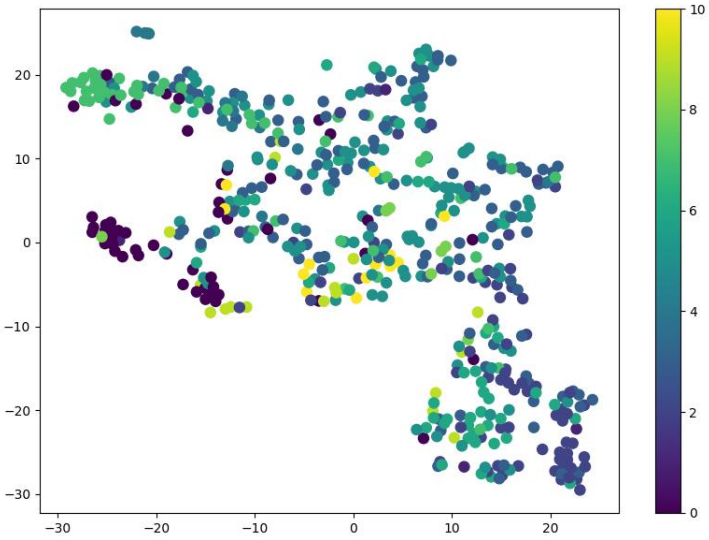


可以看到 loss 穩定下降，training accuracy 也一直升高，一共跑了 35 個 epoch，但是後來選用第 20 個 epoch 的 model，發現 validation 準確率在 20 個 epoch 時趨於穩定，大約都有 50%以上)

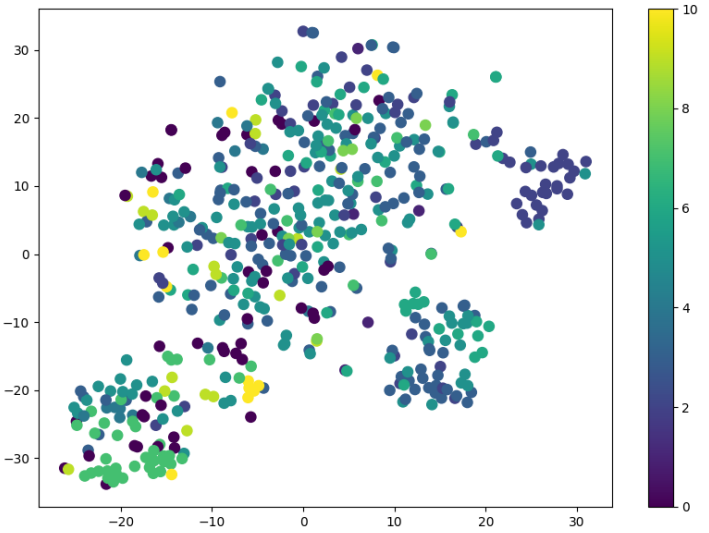
2. (15%) Visualize CNN-based video features and RNN-based video features to 2D space (with tSNE). You need to generate two separate graphs and color them with respect to different action labels. Do you see any improvement for action recognition? Please explain your observation.

TSNE:

CNN:

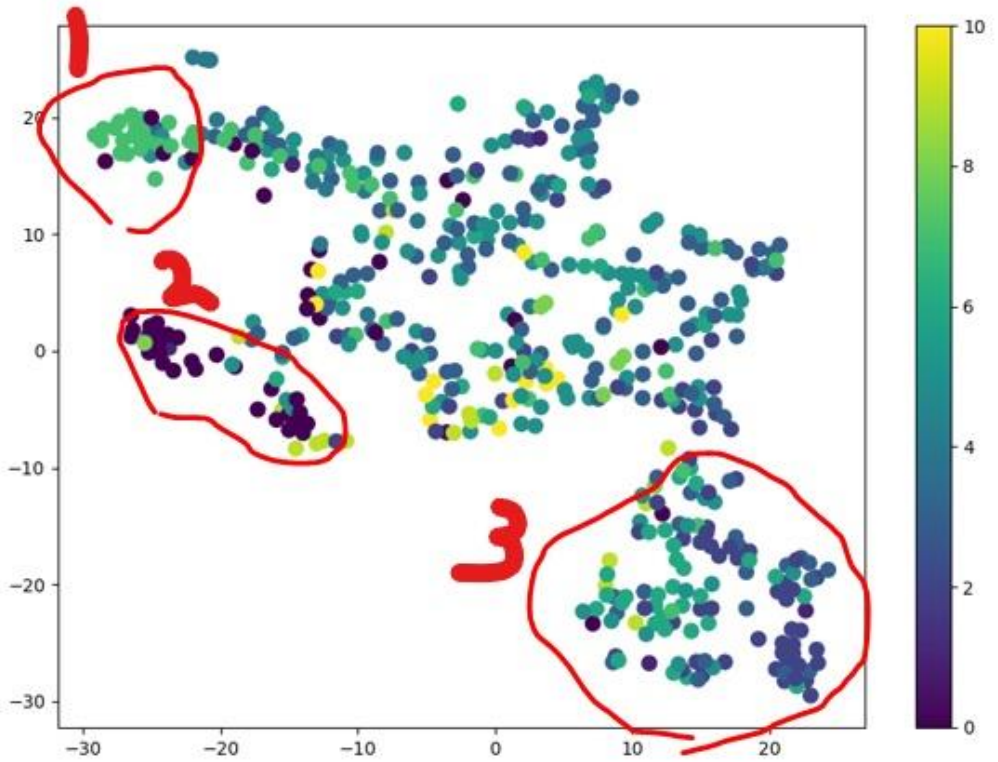

20
10
0
−10
−20
−30
−30
−20
−10
0
10
20
10
8
6
4
2
0

RNN:

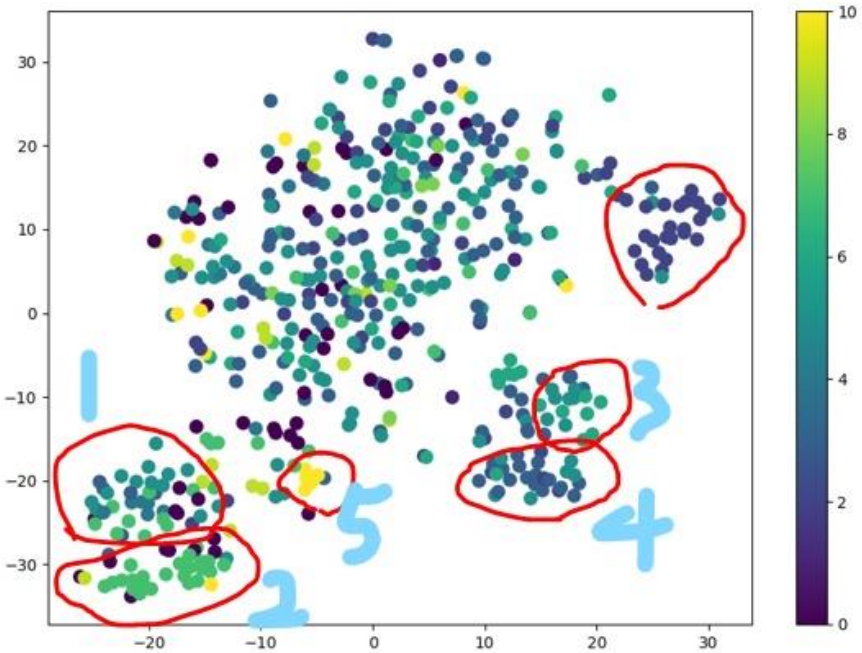

30
20
10
0
−10
−20
−30
−20
−10
0
10
20
30
10
8
6
4
2
0

Explain:

兩個都有看到某些 class 有被成功分開，代表他們取的 feature 都有一些代表性，像 CNN 的這幾種 class 都有被分開，而且圈 1 圈 2 群聚的很好。



RNN 的部分，可以從圈出的部分發現，這幾個群都分的蠻好，而且分群分得更仔細，像是圈 1,圈 2 雖然很接近，但是還是有分開，圈 3,圈 4 很接近，也有分開，而且 cnn 沒分開的黃色 label，在這邊也比較有被分出來(圈 5)



整體而言，我覺得 **TSNE 的部分 RNN 還是比 CNN 好一些，某些群分得更開一些**，實測方面 Validation 準確度也差很多，在 validation set 上 RNN 的準確率達到了 0.512，而 cnn 只有 0.307，是進步非常多的。