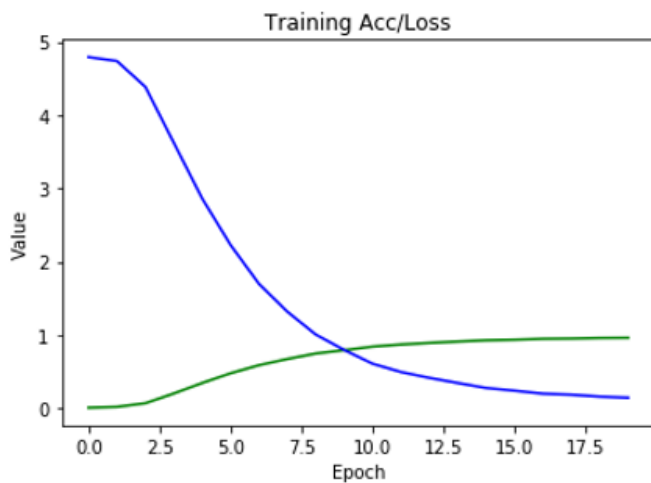


## Dog Breed Identification by Tensorflow

(VGG16, 120 classes, Train:10222, Test:10357)

id	breed
000bec180eb18c7604dcecc8fe0dba07	boston_bull
001513dfcb2ffafc82cccf4d8bbaba97	dingo
001cdf01b096e06d78e9e5112d419397	pekinese
00214f311d5d2247d5dfe4fe24b2303d	bluetick
0021f9ceb3235effd7fcd7f7538ed62	golden_retriever
002211c81b498ef88e1b40b9abf84e1d	bedlington_terrier
00290d3e1fdd27226ba27a8ce248ce85	bedlington_terrier
002a283a315af96eaea0e28e7163b21b	borzoi
003df8b8a8b05244b1d920bb6cf451f9	basenji
0042188c895a2f14ef64a918ed9c7b64	scottish_deerhound
004396df1acd0f1247b740ca2b14616e	shetland_sheepdog
0067dc3eab0b3c3ef0439477624d85d6	walker_hound
00693b8bc2470375cc744a6391d397ec	maltese_dog
006cc3ddb9dc1bd827479569fcdc52dc	bluetick
0075dc49dab4024d12fafa67074d8a81	norfolk_terrier
00792e341f3c6eb33663e415d0715370	african_hunting_dog
007b5a16db9d9ff9d7ad39982703e429	wire-haired_fox_terrier
007b8a07882822475a4ce6581e70b1f8	redbone
007ff9a78eba2aebb558afea3a51c469	lakeland_terrier
008887054b18ba3c7601792b6a453cc3	boxer

### Result (Epoch 20), Training Accuracy & Loss



### Result Scoring & Ranking on Kaggle (Epoch 20), Score: 13.76716, Rank: 1257

Score:

Your most recent submission				
Name	Submitted	Wait time	Execution time	Score
final.csv	3 minutes ago	3 seconds	3 seconds	13.76716
Complete				
<a href="#">Jump to your position on the leaderboard ▼</a>				

# PACMAN

(DFS, BFS, A\* Search)

Result: A star Search

Execute: `python pacman.py -l bigMaze -z .5 -p SearchAgent -a fn=astar,heuristic=manhattanHeuristic`



Result: Finding All the Corners (A star Search, Medium Maze)

Execute: `python pacman.py -l mediumCorners -p AStarCornersAgent -z 0.5`



## n-Queen Problem

(Genetic Algorithm, Hill Climbing)

Result: Hill Climbing (n=50), Average Attack = 2.7

```
lkp104u@csie0[1:58pm]~/AI>time ./a.out
Please enter the algorithm that you want to display (0 for HC, 1 for GA): 0
Please enter the number of queens: 50

Welcome to Hill Climbing ! (Add 50-Queen Problem)
Success Rate = 0.0666667
Average Attack = 2.7
42.580u 0.000s 0:46.38 91.8% 30+167k 0+0io 0pf+0w
```

Result: Genetic Algorithm (n=50, Iteration = 60000, Population Size = 60), Average Attack = 3.33333

```
lkp104u@csie0[9:42pm]~/AI>time ./a.out
Please enter the algorithm that you want to display (0 for HC, 1 for GA): 1
Please enter the number of queens: 50

Welcome to Genetic Algorithm ! (Add 50-Queen Problem)
Iteration = 60000, Population Size = 60

Success Rate = 0
Average Attack = 3.33333
1640.881u 0.007s 27:23.51 99.8% 30+167k 0+0io 0pf+0w
```

## Face Recognition (ML) (VGG 16, 35 Train, 30 Test)

Result (Epoch 30), Test Loss: 0.8576, Test Accuracy: 81.1%

```
1330/1330 [=====] - 30s 23ms/step - loss: 0.8212 - acc: 0.8256
Epoch 16/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.6311 - acc: 0.8624
Epoch 17/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.4584 - acc: 0.9090
Epoch 18/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.3621 - acc: 0.9278
Epoch 19/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.2777 - acc: 0.9489
Epoch 20/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.2163 - acc: 0.9564
Epoch 21/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.1757 - acc: 0.9699
Epoch 22/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.1349 - acc: 0.9752
Epoch 23/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.1128 - acc: 0.9820
Epoch 24/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.0919 - acc: 0.9850
Epoch 25/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.0815 - acc: 0.9865
Epoch 26/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.0702 - acc: 0.9872
Epoch 27/30
1330/1330 [=====] - 30s 23ms/step - loss: 0.0621 - acc: 0.9895
Epoch 28/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.0653 - acc: 0.9857
Epoch 29/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.0511 - acc: 0.9902
Epoch 30/30
1330/1330 [=====] - 31s 23ms/step - loss: 0.0467 - acc: 0.9902
1122/1122 [=====] - 21s 18ms/step

test loss: 0.8576554069791102
test accuracy: 0.8110516934046346
```

## Face Recognition (NIAG) (Open CV with face\_recognition)

Demo Result:

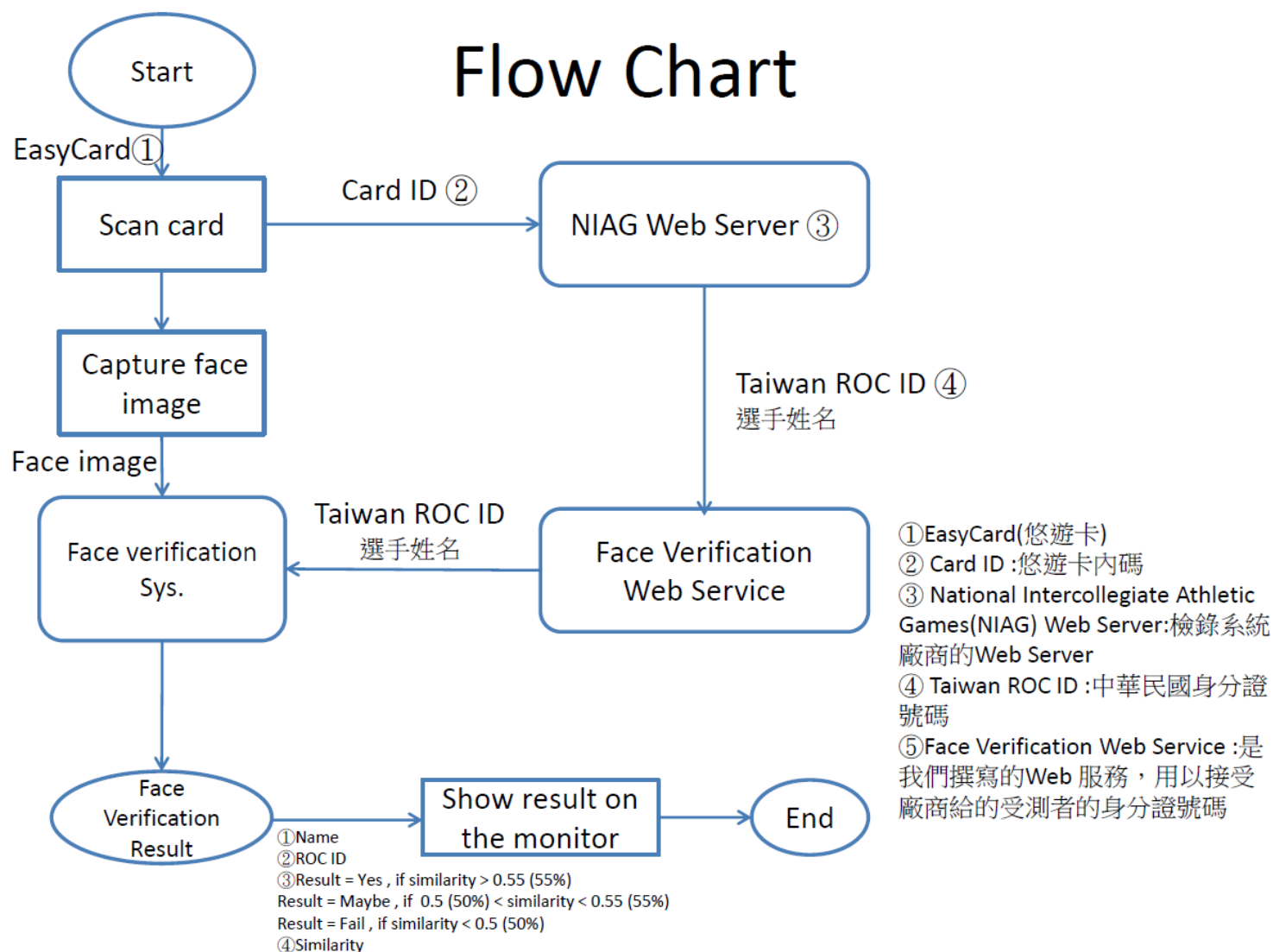


Time Consuming → Overall: 1.54 sec, Prediction: 0.912 sec

```
218.161.99.246 - - [09/May/2019 12:04:39] "POST /company HTTP/1.1" 200 162 1.001638
218.161.99.246 - - [09/May/2019 12:04:39] "POST /current/card/ HTTP/1.1" 200 1410.001153
林楷博
Transmission Success !

Overall Time: 0.912417 sec
1.54088401794
218.161.99.246 - - [09/May/2019 12:04:40] "POST /recognize HTTP/1.1" 200 1206191.541807
```

Flow Chart:



**Independent Study on Computer Science**  
(T-shirt: 1292, Logo: 2558, Shirt: 1349, Suit: 349)

Procedure:

Result (Epoch 400)

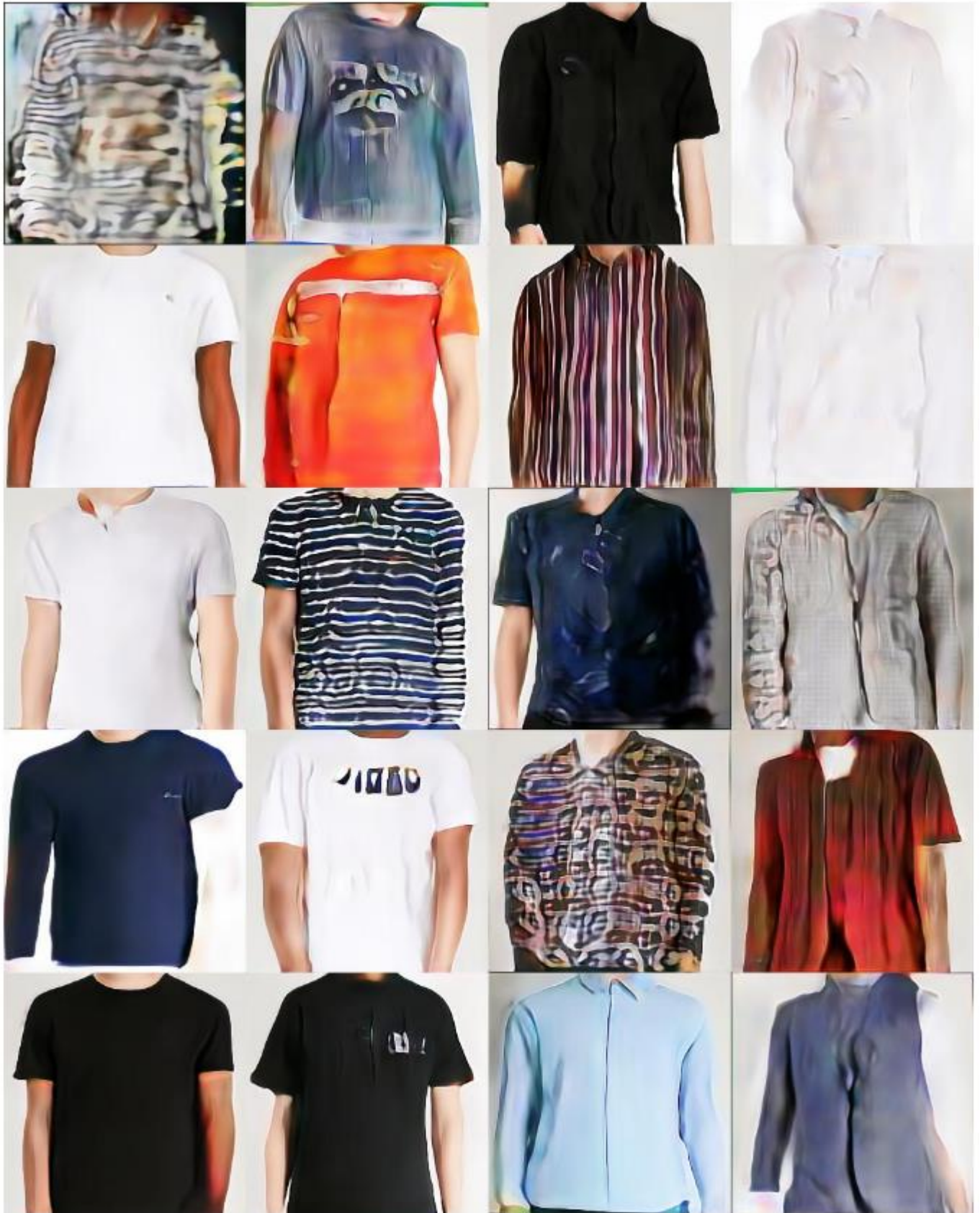


(Epoch: 400, Resolution: 128\*128)



Result (Epoch 2900):

Result (Epoch 2900)



(Epoch: 2900, Resolution: 256\*256)

Result (Epoch 3094):

Result (Epoch 3094)



(Epoch: 3094, Resolution: 256\*256)



## Result in Directory:




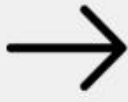



## Scribble to Painting









(Sparse Coding + CNN, Platform: Python 3.6 + Matlab R2019a)

Interface:


AI Painting Transfer



Painting Style: Kandinsky

 <input type="radio"/> Magritte	 <input type="radio"/> Monet	 <input type="radio"/> Renoir	 <input type="radio"/> van Gogh
 <input type="radio"/> Degas	 <input type="radio"/> Escher	 <input checked="" type="radio"/> Kandinsky	 <input type="radio"/> Lifshitz

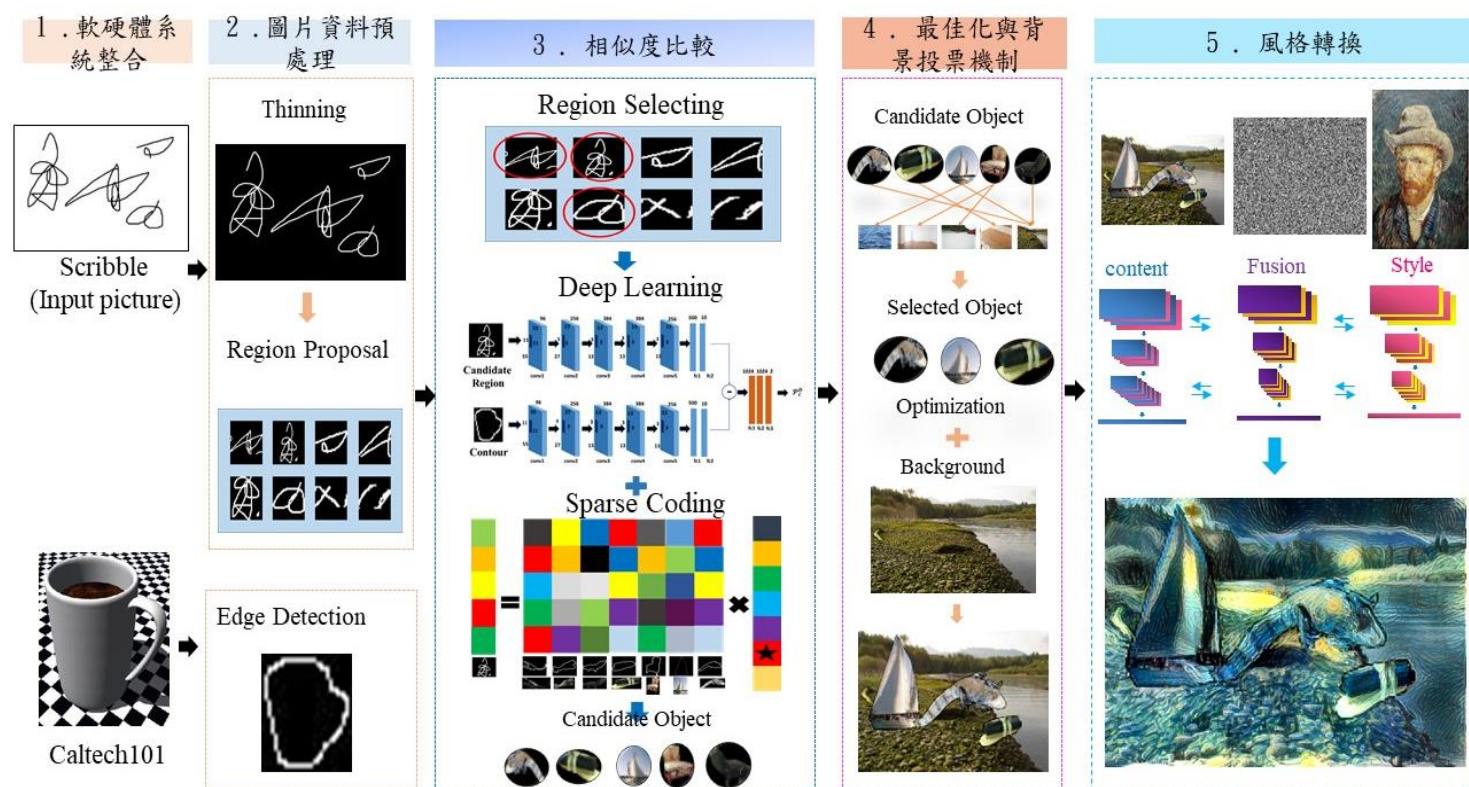
Now: 1.jpg



New style	Select	Start Watching
Open Result	Start Transfer	Stop Watching

☐ other

## Flow Chart:



## Results:

使用者塗鴉



轉換後作品

