Deep Learning for Computer Vision Homework 2

R10522606 曾柏翔

Promble 1

1. Show your model architecture.

Generator

Layer (type)	Output Shape	Param #
ConvTranspose2d-1	[-1, 512, 4, 4]	1,048,576
BatchNorm2d-2	[-1, 512, 4, 4]	1,024
ReLU-3	[-1, 512, 4, 4]	0
ConvTranspose2d-4	[-1, 256, 8, 8]	2,097,152
BatchNorm2d-5	[-1, 256, 8, 8]	512
ReLU-6	[-1, 256, 8, 8]	0
ConvTranspose2d-7	[-1, 128, 16, 16]	524,288
BatchNorm2d-8	[-1, 128, 16, 16]	256
ReLU-9	[-1, 128, 16, 16]	0
ConvTranspose2d-10	[-1, 64, 32, 32]	131,072
BatchNorm2d-11	[-1, 64, 32, 32]	128
ReLU-12	[-1, 64, 32, 32]	0
ConvTranspose2d-13	[-1, 3, 64, 64]	3,072
Tanh-14	[-1, 3, 64, 64]	0

Discriminator

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 32, 32]	3,072
BatchNorm2d-2	[-1, 64, 32, 32]	128
LeakyReLU-3	[-1, 64, 32, 32]	0
Conv2d-4	[-1, 128, 16, 16]	131,072
BatchNorm2d-5	[-1, 128, 16, 16]	256
LeakyReLU-6	[-1, 128, 16, 16]	0
Conv2d-7	[-1, 256, 8, 8]	524,288
BatchNorm2d-8	[-1, 256, 8, 8]	512
LeakyReLU-9	[-1, 256, 8, 8]	0
Conv2d-10	[-1, 512, 4, 4]	2,097,152
BatchNorm2d-11	[-1, 512, 4, 4]	1,024
LeakyReLU-12	[-1, 512, 4, 4]	0
Conv2d-13	[-1, 1, 1, 1]	8,192
Flatten-14	[-1, 1]	0
Sigmoid-15	[-1, 1]	0

2. Save the 1000 generated images in the assigned folder path for evaluation, and show the first 32 images.

3. FID and IS to evaluate your generated images.

	Baseline
FID	29.08
IS	2.04

4. Discuss what you've observed and learned from implementing GAN.

在這題中，對於對抗網路有進一步的認識，第一次測試 FID 只有 49，但經過修改 model 與 train 的部分架構後，得到了較好的分數，但仍然沒有辦法通過 Strong Baseline。

因此對於 model 再次進行了層數的調整，並使 Train images 進行隨機灰階、縮放、翻轉等等增強，但最終發現只有對於水平翻轉有增加精確度的趨勢。推測是因為 Data 是人臉，所以某些增強，反而會導致更難辨識，而人臉的對稱性對於水平反轉是不會有辨識上的障礙，所以能達到增強的效果。

Promble 2

1. Show your model architecture.

```
Generator
```

Layer (type)	Output Shape	Param #
Embedding	[-1, 128, 1, 1]	
ConvTranspose2d-1	[-1, 512, 4, 4]	1,048,576
BatchNorm2d-2	[-1, 512, 4, 4]	1,024
ReLU-3	[-1, 512, 4, 4]	0
ConvTranspose2d-4	[-1, 256, 4, 4]	131,072
BatchNorm2d-5	[-1, 256, 4, 4]	512
ReLU-6	[-1, 256, 4, 4]	0
ConvTranspose2d-7	[-1, 128, 8, 8]	524,288
BatchNorm2d-8	[-1, 128, 8, 8]	256
ReLU-9	[-1, 128, 8, 8]	0
ConvTranspose2d-10	[-1, 64, 14, 14]	32,768
BatchNorm2d-11	[-1, 64, 14, 14]	128
ReLU-12	[-1, 64, 14, 14]	0
ConvTranspose2d-13	[-1, 3, 28, 28]	3,072
Tanh-14	[-1, 3, 28, 28]	0

```
Generator
```

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 64, 14, 14]           3,072
       BatchNorm2d-2          [-1, 64, 14, 14]             128
         LeakyReLU-3          [-1, 64, 14, 14]               0
            Conv2d-4          [-1, 128, 7, 7]           131,072
       BatchNorm2d-5          [-1, 128, 7, 7]               256
         LeakyReLU-6          [-1, 128, 7, 7]               0
            Conv2d-7          [-1, 256, 7, 7]           294,912
       BatchNorm2d-8          [-1, 256, 7, 7]               512
         LeakyReLU-9          [-1, 256, 7, 7]               0
           Conv2d-10          [-1, 512, 7, 7]         1,179,648
      BatchNorm2d-11          [-1, 512, 7, 7]             1,024
        LeakyReLU-12          [-1, 512, 7, 7]               0
           Conv2d-13            [-1, 1, 1, 1]            25,088
          Flatten-14                  [-1, 1]               0
          Sigmoid-15                  [-1, 1]               0
           Linear-16                 [-1, 10]           250,890
          Softmax-17                 [-1, 10]               0
----------------------------------------------------------------
```

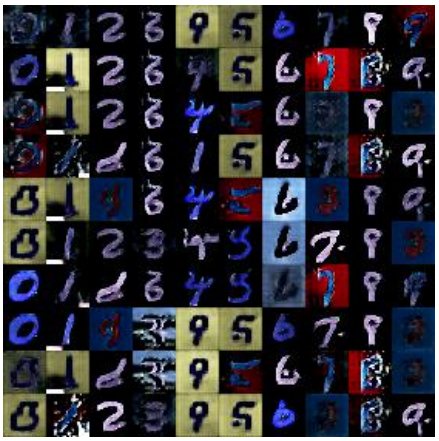
Domain

Class

2. The classification accuracy with a pretrained digit classifier.

Metric	Simple Baseline
Accuracy	84.1%

3. Show 10 images for each digit (0-9).

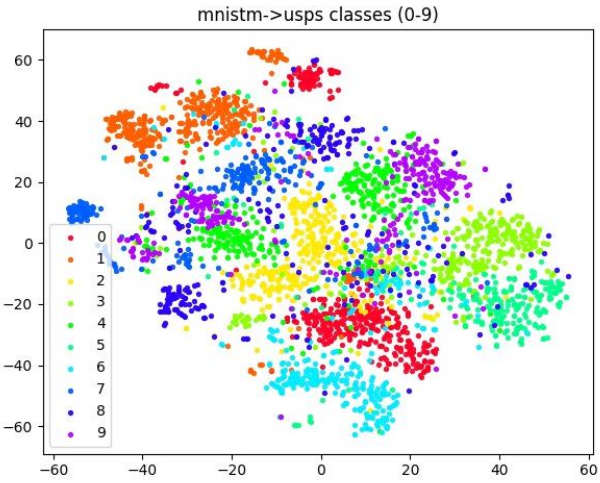


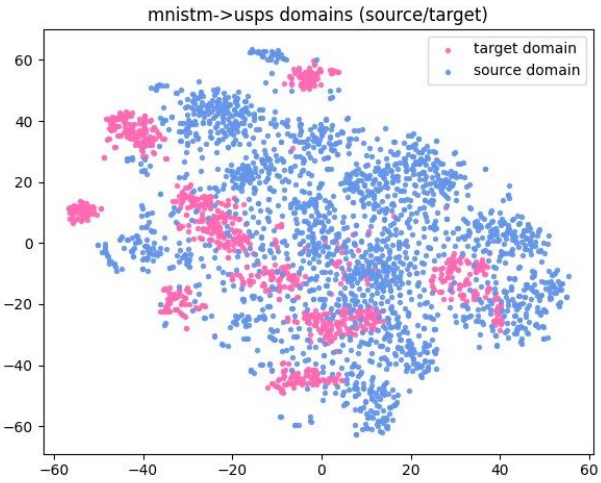
Promble 3

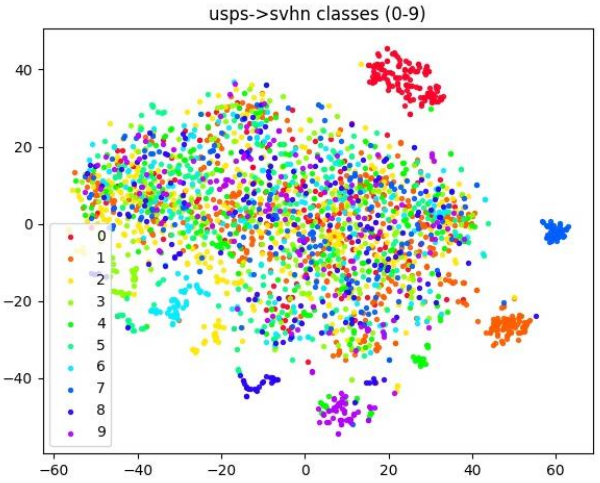
1. Compute the accuracy on target domain.

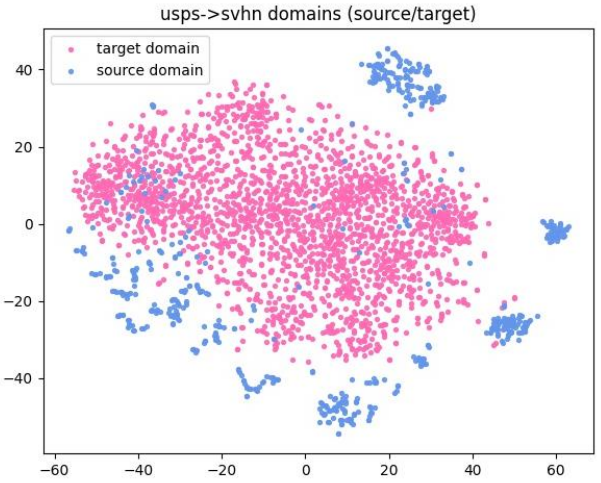
	MNIST-M → USPS	SVHN → MNIST-M	USPS → SVHN
Trained on source	70.0% (1405/2007)	44.1% (4411/10000)	29.1% (7597/26032)
Adaptation	76.2% (1529/2007)	53.4% (5336/10000)	29.4% (7660/26032)
Trained on target	95.3% (1914/2007)	96.6% (9666/10000)	88.4% (23042/26032)

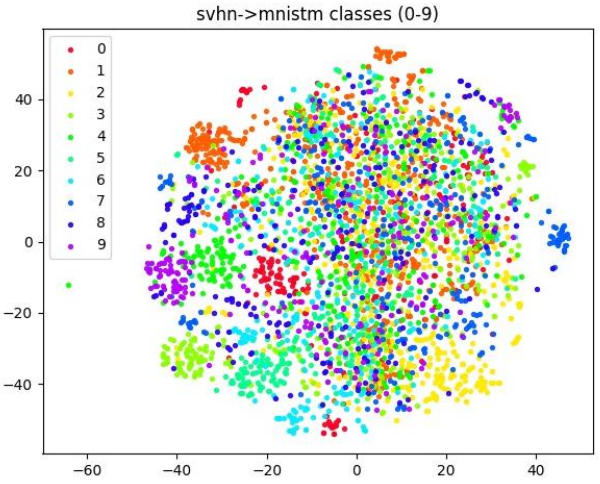
2. Visualize the latent space by mapping the testing images to 2D space with t-SNE and use different colors to indicate data of (a) different digit classes 0-9 and (b) different domains (source/target).

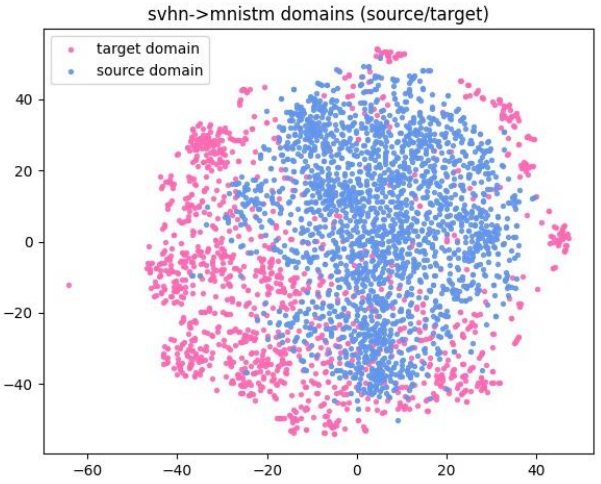




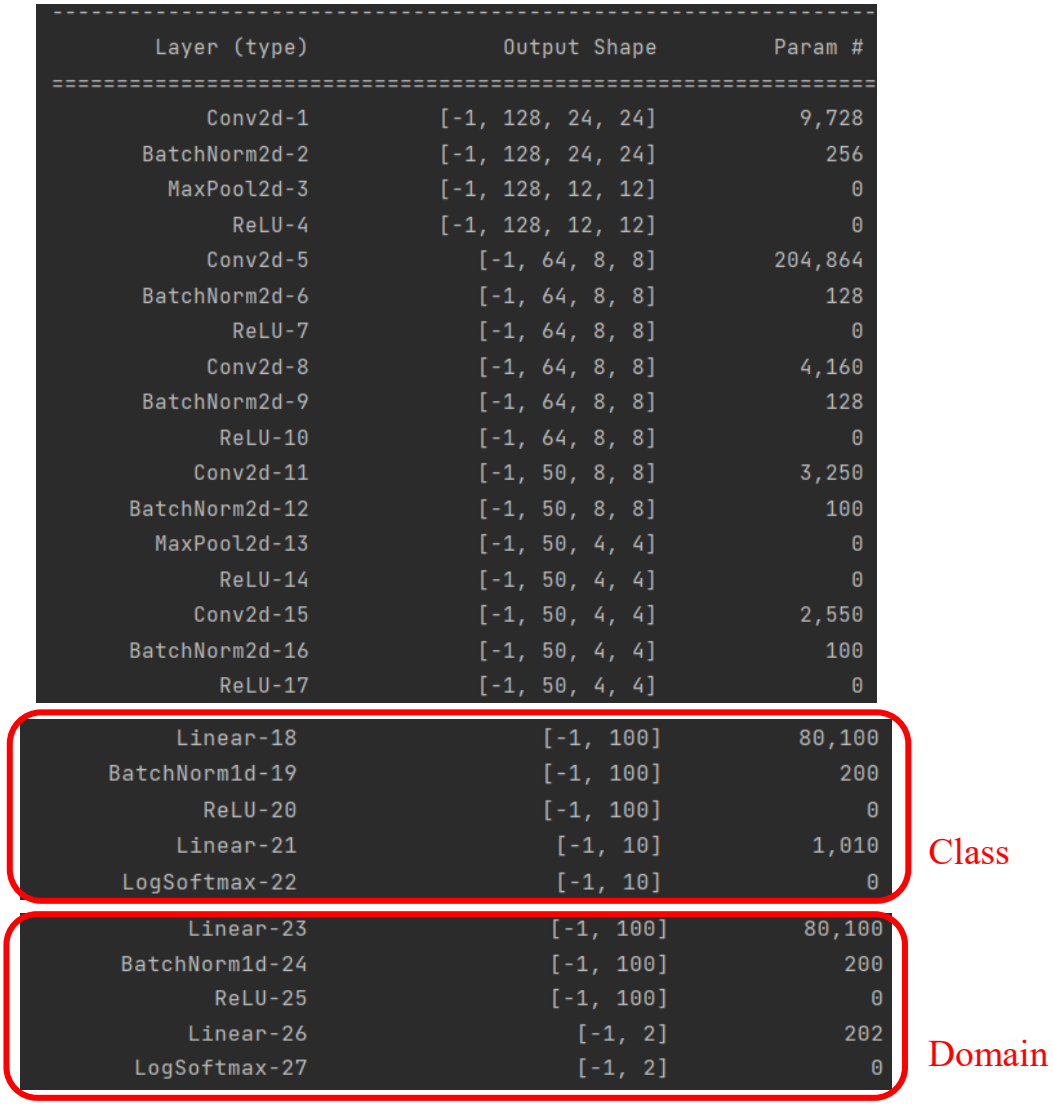








3. Describe the implementation details of your model and discuss what you've observed and learned from implementing DANN.



Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 128, 24, 24]	9,728
BatchNorm2d-2	[-1, 128, 24, 24]	256
MaxPool2d-3	[-1, 128, 12, 12]	0
ReLU-4	[-1, 128, 12, 12]	0
Conv2d-5	[-1, 64, 8, 8]	204,864
BatchNorm2d-6	[-1, 64, 8, 8]	128
ReLU-7	[-1, 64, 8, 8]	0
Conv2d-8	[-1, 64, 8, 8]	4,160
BatchNorm2d-9	[-1, 64, 8, 8]	128
ReLU-10	[-1, 64, 8, 8]	0
Conv2d-11	[-1, 50, 8, 8]	3,250
BatchNorm2d-12	[-1, 50, 8, 8]	100
MaxPool2d-13	[-1, 50, 4, 4]	0
ReLU-14	[-1, 50, 4, 4]	0
Conv2d-15	[-1, 50, 4, 4]	2,550
BatchNorm2d-16	[-1, 50, 4, 4]	100
ReLU-17	[-1, 50, 4, 4]	0
Linear-18	[-1, 100]	80,100
BatchNorm1d-19	[-1, 100]	200
ReLU-20	[-1, 100]	0
Linear-21	[-1, 10]	1,010
LogSoftmax-22	[-1, 10]	0
Linear-23	[-1, 100]	80,100
BatchNorm1d-24	[-1, 100]	200
ReLU-25	[-1, 100]	0
Linear-26	[-1, 2]	202
LogSoftmax-27	[-1, 2]	0

Class

Domain

透過 feature extraction network 後，再分別接上 Class layer 跟 Domain layer，其中某些層的 channel 是保持不變，單純加深網路來去訓練。

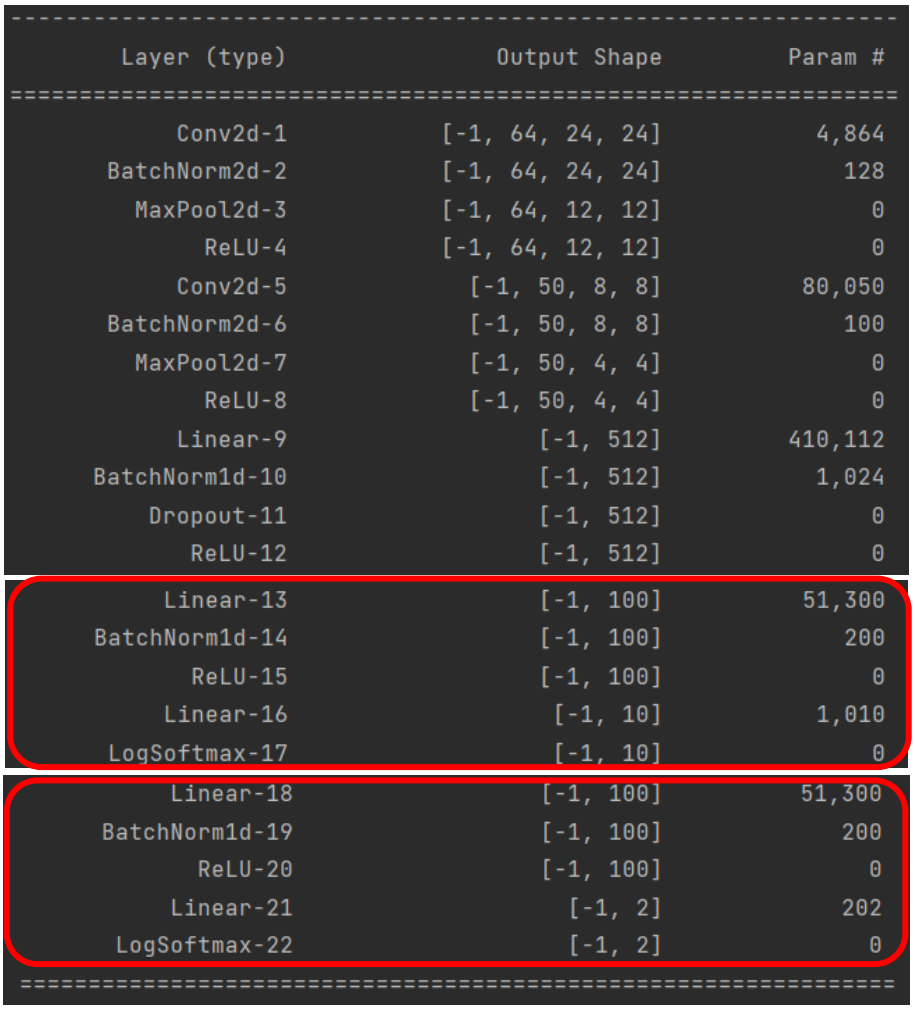
並加上 ColorJitter 增強 source 單調，但 target 花樣較多的 model，以及透過部分裁切來改善 svhn data 一張照片有多個數字的問題。

Bonus

1. Compute the accuracy on target domain.

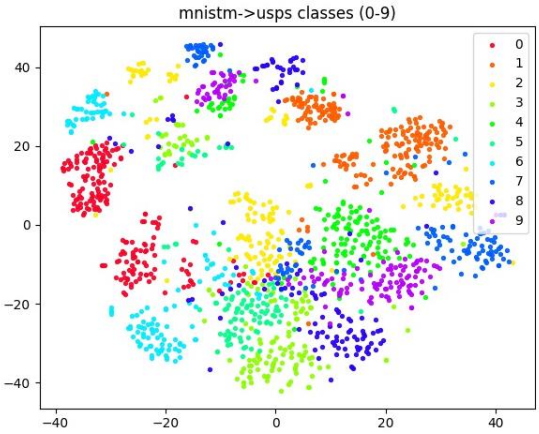
	MNIST-M → USPS	SVHN → MNIST-M	USPS → SVHN
Original model	76.2% (1529/2007)	53.4% (5336/10000)	29.4% (7660/26032)
Improved model	83.2% (1670/2007)	64.1% (6415/10000)	29.6% (7713/26032)

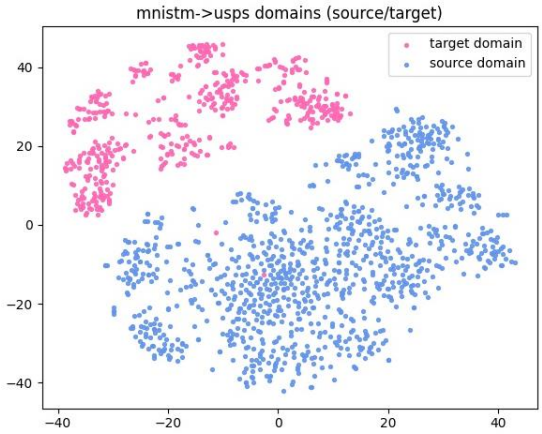
2. Briefly describe implementation details of your model and discuss what you've observed and learned from implementing your improved UDA model.

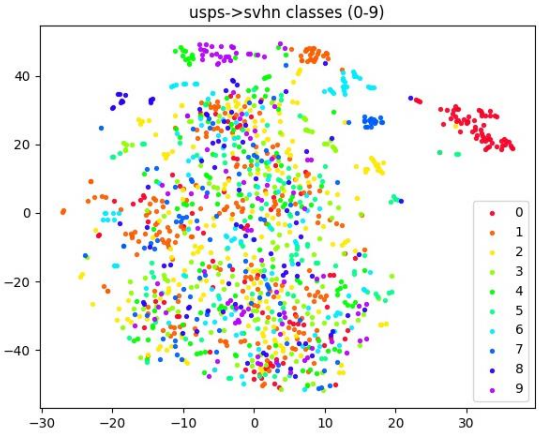


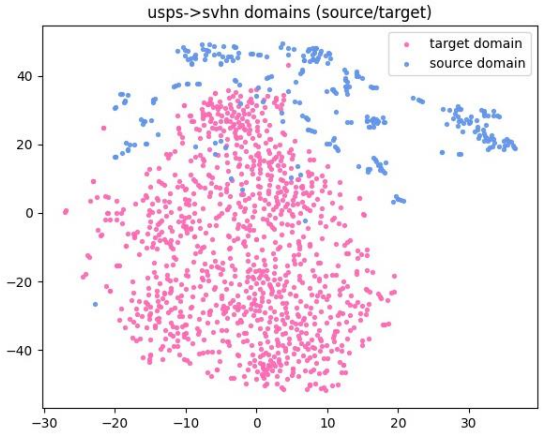
Layer (type)	Output Shape	Param #	
Conv2d-1	[-1, 64, 24, 24]	4,864	
BatchNorm2d-2	[-1, 64, 24, 24]	128	
MaxPool2d-3	[-1, 64, 12, 12]	0	
ReLU-4	[-1, 64, 12, 12]	0	
Conv2d-5	[-1, 50, 8, 8]	80,050	
BatchNorm2d-6	[-1, 50, 8, 8]	100	
MaxPool2d-7	[-1, 50, 4, 4]	0	
ReLU-8	[-1, 50, 4, 4]	0	
Linear-9	[-1, 512]	410,112	
BatchNorm1d-10	[-1, 512]	1,024	
Dropout-11	[-1, 512]	0	
ReLU-12	[-1, 512]	0	
Linear-13	[-1, 100]	51,300	Class
BatchNorm1d-14	[-1, 100]	200	Class
ReLU-15	[-1, 100]	0	Class
Linear-16	[-1, 10]	1,010	Class
LogSoftmax-17	[-1, 10]	0	Class
Linear-18	[-1, 100]	51,300	Domain
BatchNorm1d-19	[-1, 100]	200	Domain
ReLU-20	[-1, 100]	0	Domain
Linear-21	[-1, 2]	202	Domain
LogSoftmax-22	[-1, 2]	0	Domain

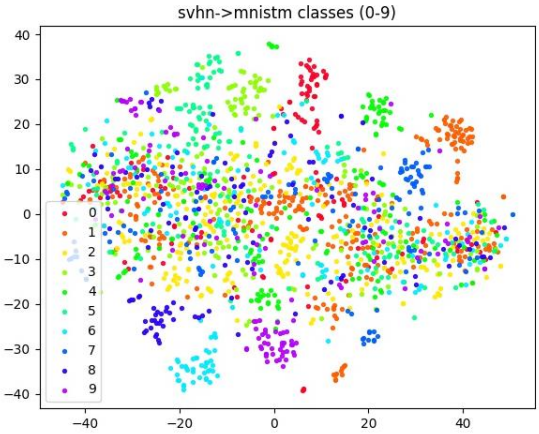
經過測試發現過深的網路並不一定會有較好的結果，因此將 feature extraction network 的層數減少，取而代之加一段 fully connected layer，最終成功得到更好的結果。不過在 USPS → SVHN 上改善的效果並不顯著，我嘗試了多種參數，精確度都只在 25-30 震盪，推斷是因為 usps 照片很單調，並且是黑白的，而拿來 test 的 svhn 較為複雜，以及照片有很多容易混淆的 noise。

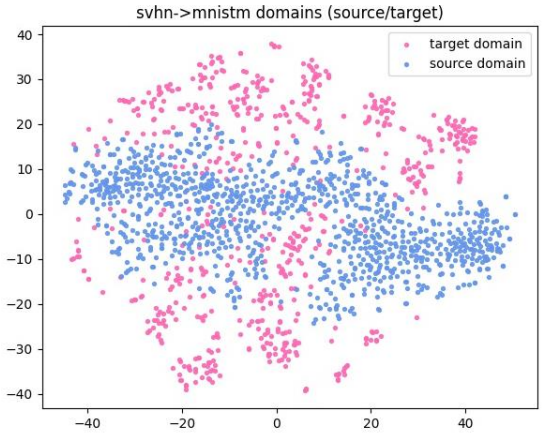












Reference

[1] GAN
https://github.com/kai860115

[2] t-SNE
https://github.com/Leohsieh57

[3] ACGAN
r10522611 周昱辰、https://reurl.cc/73A0Z1

[4] DANN
https://reurl.cc/WXYNaZ