

## ADLxMLDS HW4

R06922055 資工所 碩一 吳均庭

### Model Description

Ref : <https://arxiv.org/pdf/1708.05509.pdf>

#### Generator:

網路架構參考SRResnet之架構, Input (100(z) + 23(tag)) dim 向量, 先經過fc放大為  $64 * 12 * 12$  tensor, 經過BN與relu, 再通過16個channel = 64 的Resblock, Resblock內為兩層conv layer(k3n64s1) 通過 BN與relu, 最後經過3層k3n256s2的conv, 每層經都經過pixelSuffler來進行upscaling在通過BN與relu, 最後經過k9n3s1的conv, 通過tanh作為output。

#### Objective :

minimize:

$$\mathcal{L}(G) = \lambda_{adv} \mathcal{L}_{adv}(G) + \mathcal{L}_{cls}(G)$$

$$\mathcal{L}_{adv}(G) = \mathbb{E}_{x \sim P_{noise}, c \sim P_{cond}} [\log(D(G(z, c)))]$$

$$\mathcal{L}_{cls}(G) = \mathbb{E}_{x \sim P_{noise}, c \sim P_{cond}} [\log(P_D[c|G(z, c)])]$$

#### Discriminator:

Input 為  $96 * 96 * 3$  image, 經過 k4n32s2 conv、兩個Resblock(k3n32s1 conv \* 2)、k4n64s3 conv、4個Resblock(k3n64s1 conv \* 2)、k4n128s2 conv、4個Resblock(k3n128s1 conv \* 2)、k3n256s2 conv、4個Resblock(k3n256s1 conv \* 2)、k3n512s2 conv、4個Resblock(k3n512s1 conv \* 2)、k3n1024s2 conv, 最後經過fc output scalar, 同時也拉出一個fc branch 輸出23維vector 來做 multilabel classification。

#### Objective :

minimize:

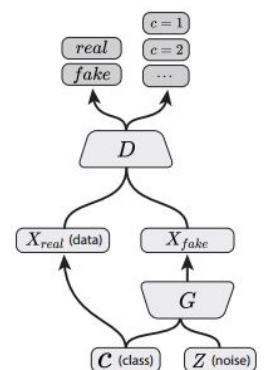
$$\mathcal{L}(D) = \mathcal{L}_{cls}(D) + \lambda_{adv} \mathcal{L}_{adv}(D)$$

$$\mathcal{L}_{adv}(D) = -\mathbb{E}_{x \sim P_{data}} [\log D(x)] - \mathbb{E}_{x \sim P_{noise}, c \sim P_{cond}} [\log(1 - D(G(z, c)))]$$

$$\mathcal{L}_{cls}(D) = \mathbb{E}_{x \sim P_{data}} [\log P_D[label_x|x]]$$

#### ACGAN:

這次使用ACGAN 架構並在Discriminator 最後一層加上一個fc 來做 multilabel prediction, 另外使用Dragan 的gradient penalty來穩定training。並在 discriminator loss 加上對real image class 的prediction error, 交替更新Discriminator 與 discriminator 來進行網路訓練。



AC-GAN  
(Present Work)

## Improve the result

原本只使用cGAN來進行訓練，發現結果圖片不夠清晰，不太符合敘述，並容易有扭曲的情形，網路中使用以下方法來改善結果：

- ACGAN :  
使用ACGAN 網路架構，在原本的discriminator 加上一個classifier 的branch來predict input 的class，發現結果比原本在DCGAN加上condition更符合input 敘述。
- Dragan :  
利用dragan所提出出之方法，在discriminator 加上gradient penalty，來穩定training 過程。ref: <https://arxiv.org/abs/1705.07215>
- Resnet :  
使用SRResnet之網路架構，作為更好的generator與discriminator，可以學到更為細緻的feature，生成之圖品質更好。
- 96 \* 96 image :  
適用96 \* 96 image 作為網路input 與output，能使結果得到更為清晰的影像。
- 在discriminator加上 spectral normalization可以使training更加穩定，較不會發生mode collapse之情形。  
Ref: <https://arxiv.org/abs/1705.10941>

## Experiment settings and observation

在training data 中發現有圖片沒有標註眼睛、頭髮顏色，或同時有多個頭髮、眼睛顏色之情況，另外bicolored 雙色眼睛的圖片夜可能會造成網路產生混淆，所以在preprocessing 過程中直接將這街圖片捨棄。另外對於對剩下的圖片進行水平翻轉，與順逆時針5度的旋轉，來進行data augmentation，最後產生45628張圖片的training data。

Training minibatch size=64，Generator input 為將頭髮眼睛顏色分別的onehot vector concat 成 23(12 + 11)維的 vector，輸出 96 \* 96 \* 3 的image，Discriminator input 為 64 \* 96 \* 96 \* 3，最後輸出一個scalar表示input 為real 的分數，以及23維的vector 表示每一個label的分數，透過 Discriminator loss = DCGAN loss + classification loss + gradient penalty，generator loss = DCGAN loss +classification loss，透過交替更新 generator 與 generator 即可完成GAN 網路訓練。

另外在discriminator 加上spectral normalization可以讓training更加穩定，且生成的圖片更為清晰，從loss可以觀察到，比起dragan與wgan更穩定，generator loss 不會上升太快，且到500epoch仍尚未發生mode collapse之情形，且生成之圖片更為清晰。

**Generated Result:**

