DL HW2 Report 電機系碩士班 林淞祐111061562

**Problem 1:**

**(1)AE model architecture**

**Encoder**

Conv2d

in\_channel:32

out\_channel:64

Hin, Win =25

Hout, Wout = 25

Leaky

Relu

(slope=0.01)

Conv2d

in\_channel:3

out\_channel:32

Hin, Win =50

Hout, Wout = 50

Maxpool2D

Hin, Win=50

Hout, Wout =25

Data

**Encoder**

Leaky

Relu

(slope=0.01)

Latent Code

Maxpool2D

Hin, Win=25

Hout, Wout =12

Full connected

In\_feature = 64\*12\*12

Out\_feature = 256

**decoder**

**decoder**

MaxUnpool2D

Hin, Win=25

Hout, Wout =50

Leaky

Relu

(slope=0.01)

TransedConv2d

in\_channel:32

out\_channel:3

Hin, Win =50

Hout, Wout = 50

Sigmoid

Full connected

In\_feature = 256

Out\_feature = 64\*12\*12

Leaky

Relu

(slope=0.01)

TransedConv2d

in\_channel:64

out\_channel:32

Hin, Win =24

Hout, Wout = 25

MaxUnpool2D

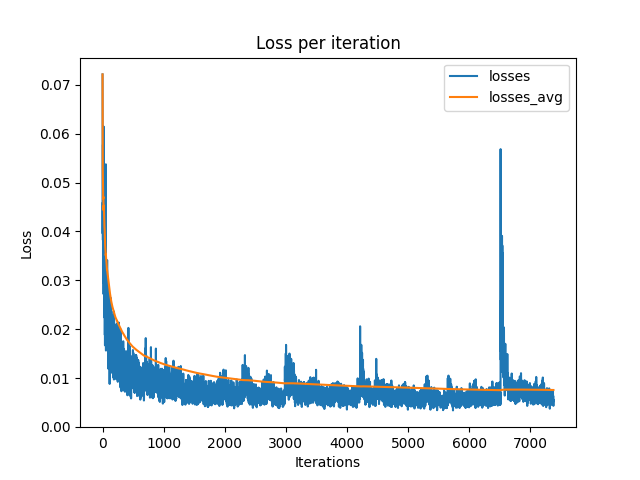
Hin, Win=12

Hout, Wout =24

Reconstructed

Data

AE model training Loss(MSELoss):



AE model discussion:

1.condtion:

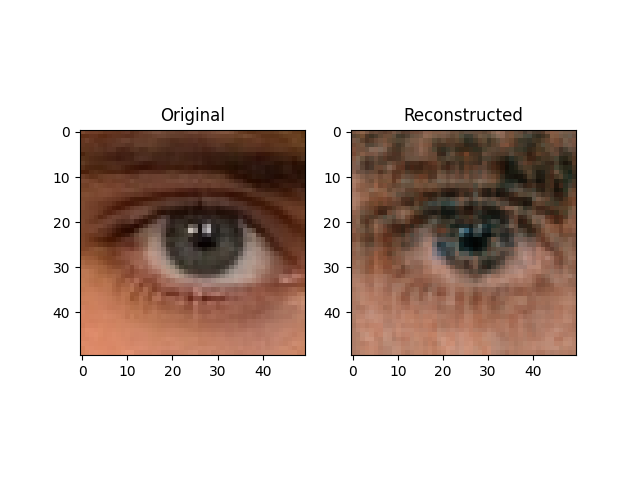
Epoch =20, batch\_size=4, learning\_rate=0.005

2.result:

PSNR=21.9617, SSIM=0.7218 , reconstruct Loss: 0.0075

3.disscussion:

會使用conv2D以及MaxPool2D是因為針對圖片的學習中，這兩種hidden layers比起full connected，能夠更好的去fit住圖片中更有意義的特徵，藉由保留圖片中每個特徵與特徵之間的空間及距離，因此在連續的hidden layer計算中，能夠更精準地傳達圖片特徵至下一層連接層。但由於conv2D消耗的運算資源相對比較大，所以在encoder最後使用full connected來編成最後的latent code, 這樣做的目的除了可以減少運算資源外，主要是測試latent code 能夠壓縮至何種程度，卻仍然能夠有效地重建出原始圖片。經由實驗過後，可以發現將原來為每張3\*50\*50的RGB圖片，編成1\*256的latent code 之後，重建出來的眼睛與原圖片的眼睛雖然較為模糊，但是顏色上以及輪廓上的表現都能夠人眼能夠看出的範圍下表現出來。



**(1)VAE model architecture**

**Encoder**

Conv2d

in\_channel:32

out\_channel:64

Hin, Win =25

Hout, Wout = 25

Leaky

Relu

(slope=0.01)

Conv2d

in\_channel:3

out\_channel:32

Hin, Win =50

Hout, Wout = 50

Maxpool2D

Hin, Win=50

Hout, Wout =25

Data

**Encoder**

Mean value

Leaky

Relu

(slope=0.01)

Maxpool2D

Hin, Win=25

Hout, Wout =12

Full connected

In\_feature = 64\*12\*12

Out\_feature = 256

Flatten

Log

variance

**Encoder**

Reparameterize

Output = Mean + (eps\*std)

(std=exp(log\_var/2), eps=randn\_like(std))

Latent Code

**decoder**

MaxUnpool2D

Hin, Win=12

Hout, Wout =24

TransedConv2d

in\_channel:64

out\_channel:32

Hin, Win =24

Hout, Wout = 25

Leaky

Relu

(slope=0.01)

Full connected

In\_feature = 256

Out\_feature = 64\*12\*12

**decoder**

Reconstructed

Data

Sigmoid

TransedConv2d

in\_channel:32

out\_channel:3

Hin, Win =50

Hout, Wout = 50

Leaky

Relu

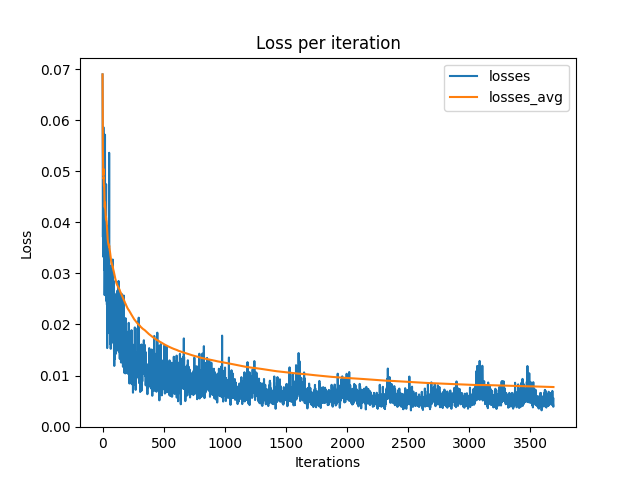
(slope=0.01)

MaxUnpool2D

Hin, Win=25

Hout, Wout =50

VAE model training Loss(MSELoss):



VAE model discussion:

1.condtion:

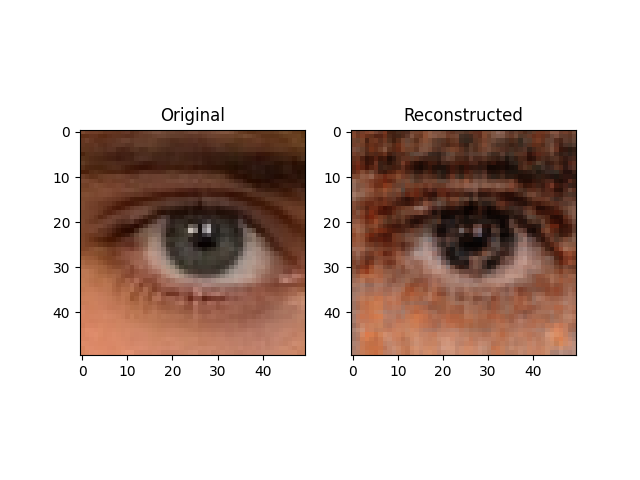
Epoch =10, batch\_size=4, learning\_rate=0.005

2.result:

PSNR=24.09, SSIM=0.7297 , reconstruct Loss: 0.0073

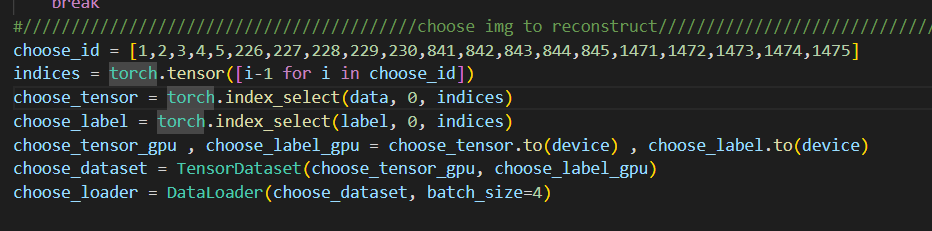
3.disscussion:

在VAE的架構中，多了Reparameterize的過程，函式包括兩個參數:1. mean of encoder’s latent space encoding, 2. the encoder’s latent space，在這裡使用的是simple variational autoencoder，由於標準的AE將圖片的feature壓縮成latent code之後，在還原的過程中，只能還原出那些已經訓練過的特徵，無法產生出新的圖片，相反的VAE壓縮成的latent vector space具有連續性，能夠幫助生產出新圖片。由數據可以看出，VAE的表現確實比標準AE表現來的優異，從reconstructed 的圖片也可以看出，兩者雖然都能夠劃出原圖片的眼睛輪廓，但是舉眉毛以及睫毛的部分來說，可以發現VAE相較於AE，更能重塑出圖片的顏色以及細節。

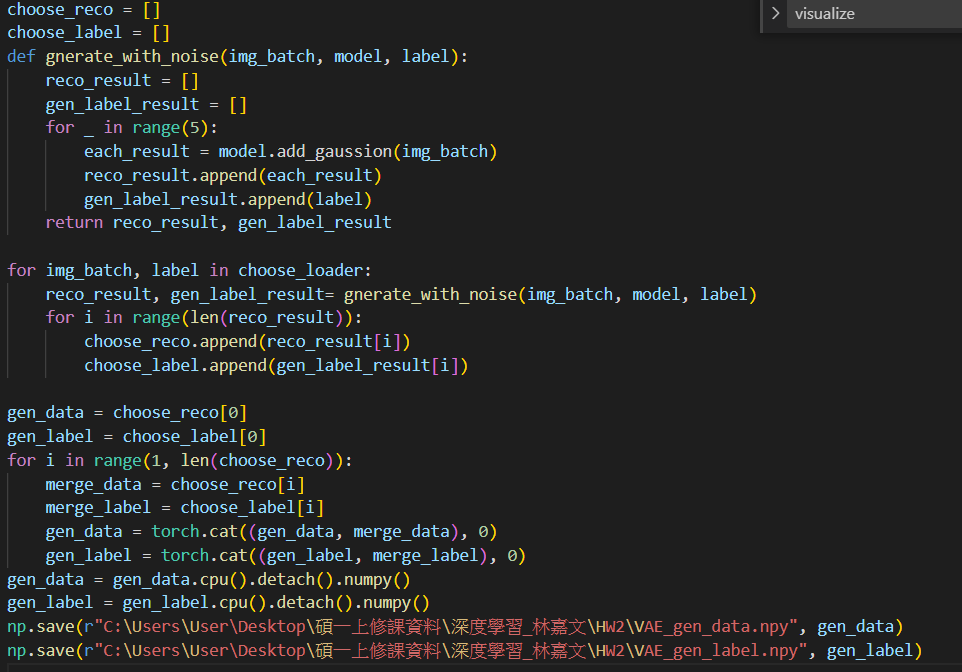


**Problem 2: construct the new sample with gaussion noise(VAE)**

(1)

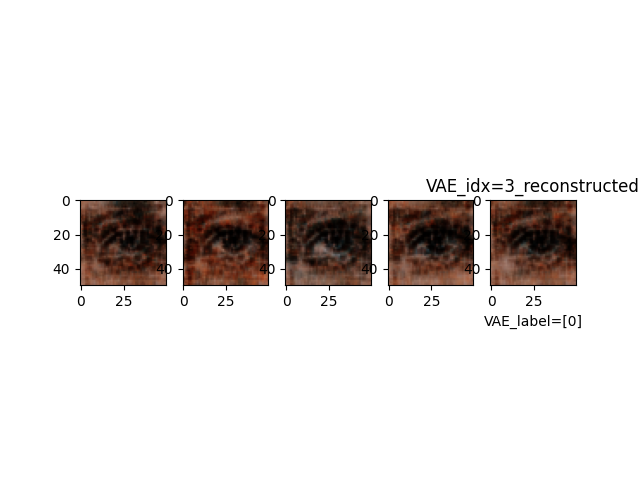
依題意選擇要求的圖片，並且重新group出一個新的choose\_loader，方便處理後續資料。

(2)

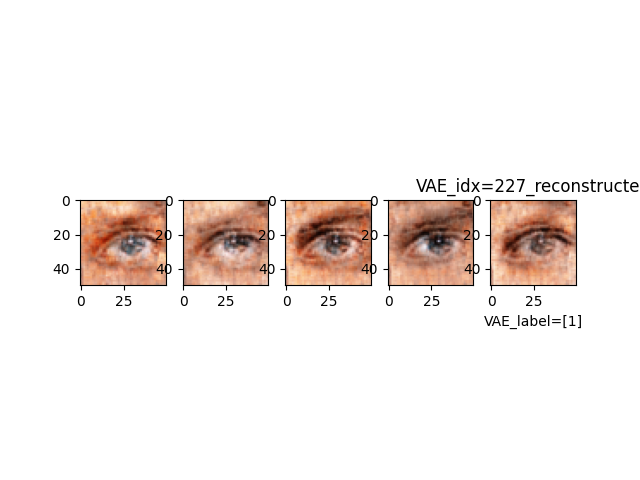
batch\_size=4, 共5個batches, 每個batch generate 5次，所以共有4\*5\*5=100張圖片，最後將一張一張圖片存成ndarray類別，並存成.npy file。

(3)constructed image with different gaussion noise

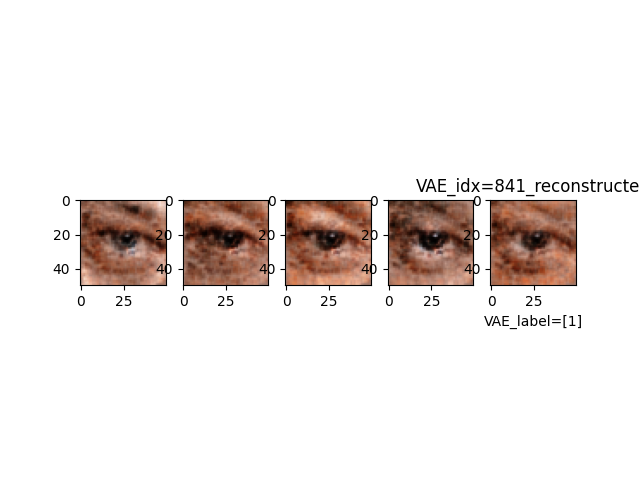
第3張



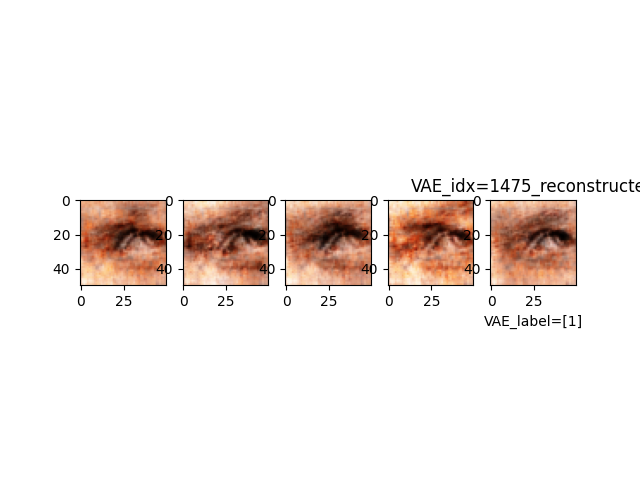
第227張



第841張



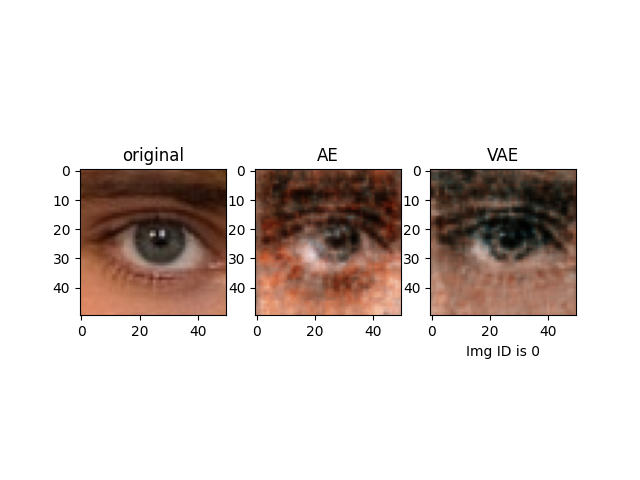
第1475張

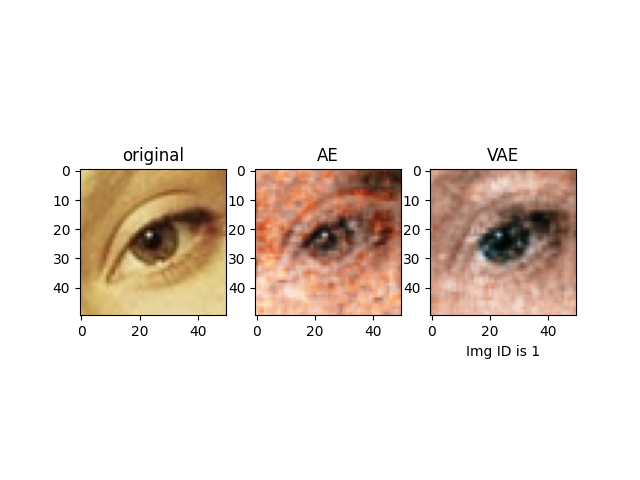


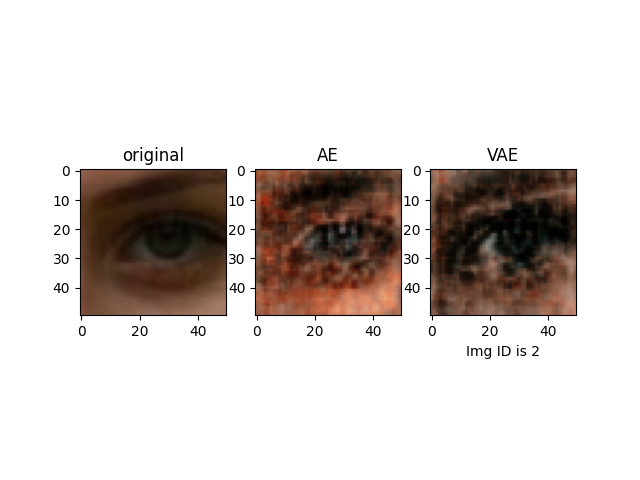
**Problem 3: hyperparameters adjusting (batch\_size are all 4)**

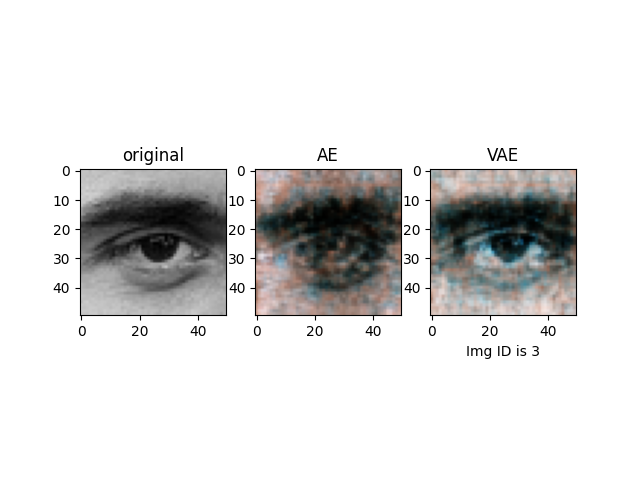
|  |  |  |
| --- | --- | --- |
| Epochs | AE | VAE |
| 5 | Avg\_loss = 0.106 , PSNR=22.99 | Avg\_loss = 0.0096 , PSNR=23.01 |
| 10 | Avg\_loss = 0.0085, PSNR=22.84 | Avg\_loss = 0.0073 , PSNR=24.09 |
| 15 | Avg\_loss = 0.0076 , PSNR=22.89 | Avg\_loss = 0.0069, PSNR=23.53 |

**Problem 4: compare AE & VAE with same gaussian noise**







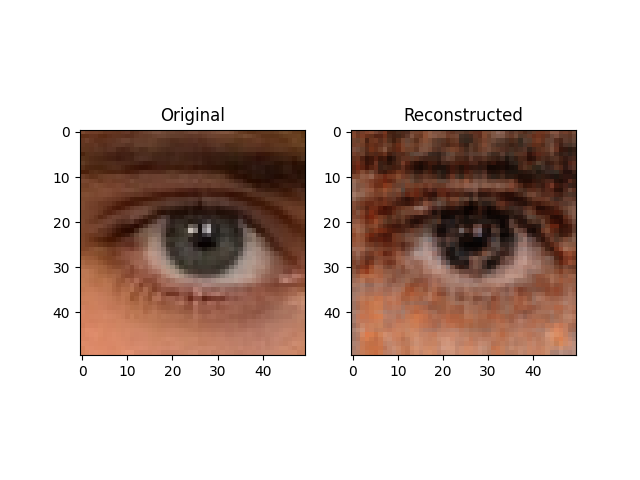


從以上的4個cases來比較，可以發現AE跟VAE最大的差異在於眼睛輪廓以及顏色上的表現，VAE能將原圖的眼睛形狀特徵表達得比較好，AE有時會過於表達某些方面的特徵(眼白、陰影)，反而讓生成的圖片與原圖片出現較大的差異，進而使損失提升。

**Problem 5-1:**

因為降維的關係，AE&VAE通常使用比較小的latent維度，大部分輸入圖片的資訊以及特徵很難通過bottleneck的地方，除此之外，model在每個batch迭代時，會試著去將低loss，因此AE&VAE只能獲得較模糊的圖片。但藉由藉由提高latent的維度，可以得到非常好的重建效果，如下比較圖所示，但也因此失去降維的目的。

Latent\_code\_shape = 1 \* 256:



Latent\_code\_shape = 64 \* 48 \* 48:

