Lab2

0412237陳逸豐

* 1. Introduction (5%)

這次lab要實作出deep image prior這篇論文的內容，Requirement 1是要做出CNN對不同的input都能夠訓練出很好的網路，Requirement 2是做image denoising，Requirement 3 是要做 super resolution。

* 2. Experiment setup (5%)
  + The detail of your model

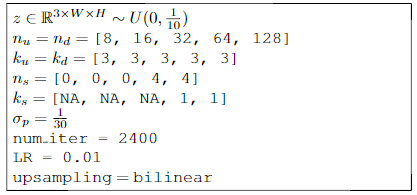
這次lab所使用的model是Hourglass network，也就是Autoencoder，再加上shortcut connection。Model也是由多個block組成，block又分成兩種encode跟decode，

Encode block是由一層conv，一層downsample，一層BN，一層leaklyRelu，一層conv，一層BN，一層leaklyRelu所組成。Decode block是由一層upsample，一層conv，一層BN，一層leaklyRelu，一層conv，一層BN，一層leaklyRelu，一層upsample組成。Encode跟decode中間又有shortcut連接，根據不同的requirement，shortcut會有不同

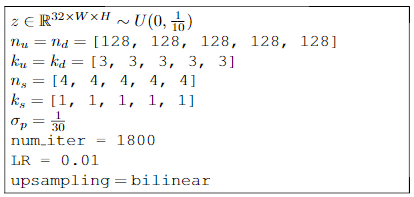
* + Report all your training hyper-parameters

如spec所說的

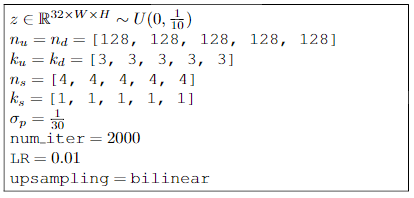
Requirement 1:



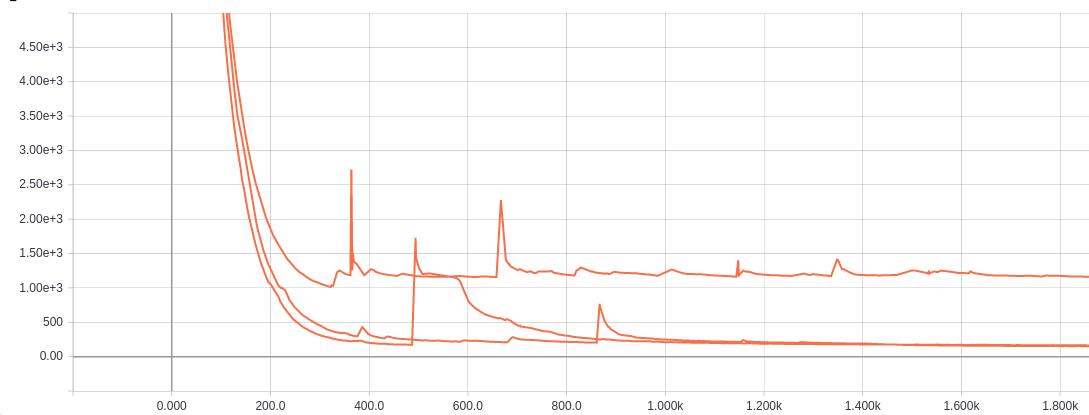
Requirement 2:



Requirement 3:



* 3. Result
  + Requirement 1 (10%, 20%)
    - Training loss curve (you need to record training loss every iteration)



the lowest is image, the middle one is image+noise, the highest is image+shuffle

* + Requirement 2 (10%, 20%)
    - Visualize the progress of inverted image

Iter360:



Iter720:

iter1080:

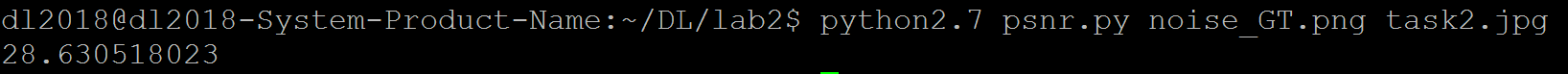


Iter1440:



* + - Final image and its PSNR





* + Requirement 3 (10%, 20%)
    - Visualize the progress of inverted image

Iter400:

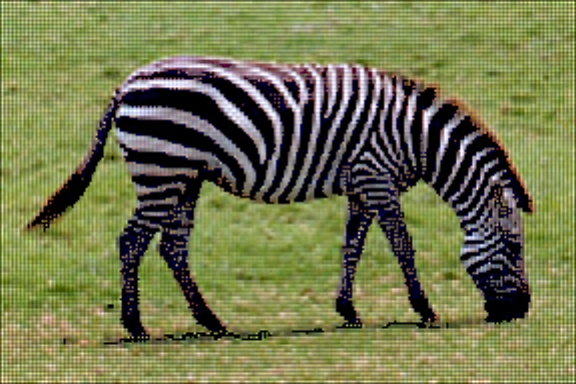


Iter800:

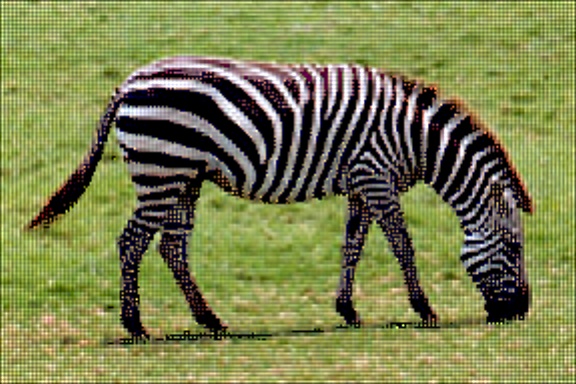


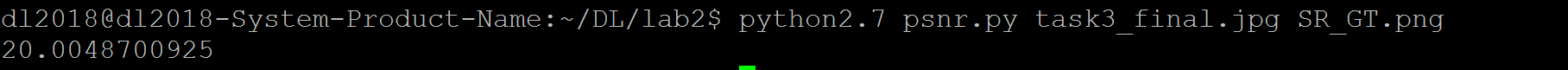
Iter1200:



Iter1600: 

* + - Final image and its PSNR





* 4. Discussion (10%, 20%)

從實驗過程中可以發現CNN對於處理影像辨識的問題非常厲害，即使input是張黑白的雜訊，經過多次的iteration後就能夠訓練出類似target的圖片，並且從過程中也可以發現，他可以從架構先開始學習，逐漸學習出影像中的細節，在Requirement 2跟3，如果讓他繼續跑的話，效果應該會更佳。