

智慧化企業整合_Project2

SR-NET 與 ESRGAN 生成圖像之比較

第一組

110034532 白哲睿

110034535 黃于瑞

110034561 洪聖博

110034564 楊笠笙

一、背景介紹.....	3
(一)、背景說明.....	3
(二)、5W1H 分析.....	3
二、方法介紹.....	3
(一)、SRGAN.....	3
(二)、ERSGAN.....	4
(三)、PSNR (Peak Single-to-Noise Ratio).....	6
三、個案研究.....	9
(一)、資料集介紹.....	9
(二)、資料前處理.....	10
(三)、SR-NET 模型建立.....	12
(四)、參數優化.....	14
(五)、方法比較.....	15
(六)、ESRGAN 優化原始圖像.....	16
四、結論.....	18
(一)、貢獻.....	18
(二)、侷限性.....	18
(三)、適用性.....	18
(四)、未來改善.....	18
五、參考文獻.....	19

一、背景介紹

(一)、背景說明

希望可以透過機器學習中 SRGAN 將生成式對抗網路 (GAN) 用於超解析度技術 (Super-Resolution) 問題來幫助提高照片解析度不足的問題。其出發點是傳統的方法一般處理的是較小的放大倍數，當影像的放大倍數在 4 以上時，很容易使得到的結果顯得過於平滑，而缺少一些細節上的真實感。因此 ESRGAN 使用 GAN 來生成影像中的細節。SRGAN 的基礎上進行了改進，包括改進網路的結構、判決器的判決形式，以及更換了一個用於計算感知域損失的預訓練網路。

(二)、5W1H 分析

Why	一些較舊的照片或是受限於手機硬體鏡頭的因素，導致照片無法呈現更清晰的細節
What	透過機器學習，不論是 CNN 或是 ESRGAN 對照片進行去模糊化
Where	課程歷史教材需要採用具有年代的照片或是手機相片優化程式。
When	拍出照片的當下又或是需要觀察照片細節的時候。
Who	一般手機用戶又或是歷史學者。
How	利用機器學習 CNN 或是 ESRGAN 進行相片去模糊清晰化。

二、方法介紹

(一)、SRGAN

超解析度技術 (Super-Resolution) 是指從觀測到的低解析度影像重建出相應的高解析度影像，在監控裝置、衛星影像和醫學影像等領域都有重要的應用價值。SR 可分為兩類：從多張低解析度影像重建出高解析度影像和從單張低解析度影像重建出高解析度影像。基於深度學習的 SR，主要是基於單張低解析度的重建方法，即 Single Image Super-Resolution (SISR)。

SRGAN (Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, arxiv, 21 Nov, 2016) 將生成式對抗網路 (GAN) 用於 SR 問題。其出發點是傳統的方法一般處理的是較小的放大倍數，當

影像的放大倍數在 4 以上時，很容易使得到的結果顯得過於平滑，而缺少一些細節上的真實感。因此 SRGAN 使用 GAN 來生成影像中的細節。

SRGAN 使用的生成式網路和判別式網路分別如下：

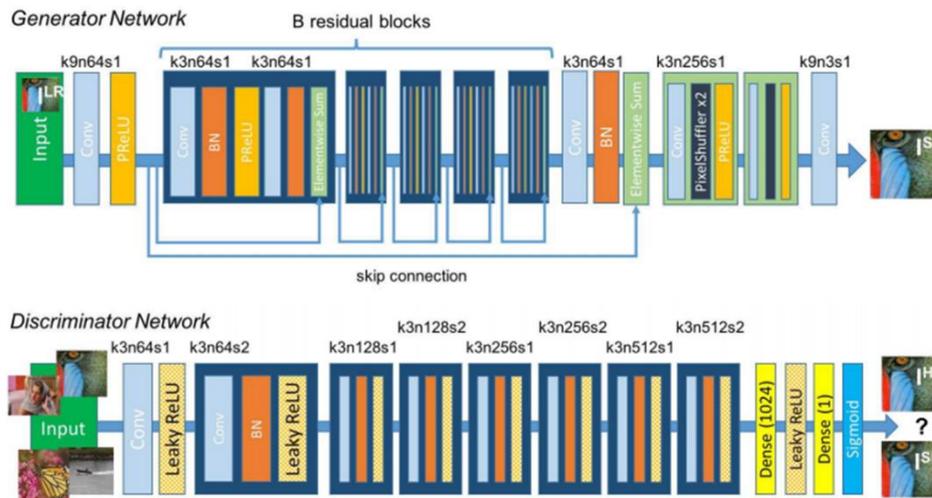


Figure 4: Architecture of Generator and Discriminator Network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer. https://blog.csdn.net/sinat_36197913

在生成網路部分(SRResNet)部分包含多個殘差塊，每個殘差塊中包含兩個 3×3 的卷積層，卷積層後接批規範化層(batch normalization, BN)和 PReLU 作為啟用函式，兩個 2×2 亞畫素卷積層(sub-pixel convolution layers)被用來增大特徵尺寸。在判別網路部分包含 8 個卷積層，隨著網路層數加深，特徵個數不斷增加，特徵尺寸不斷減小，選取啟用函式為 LeakyReLU，最終通過兩個全連線層和最終的 sigmoid 啟用函式得到預測為自然影像的概率。

(二)、ERSGAN

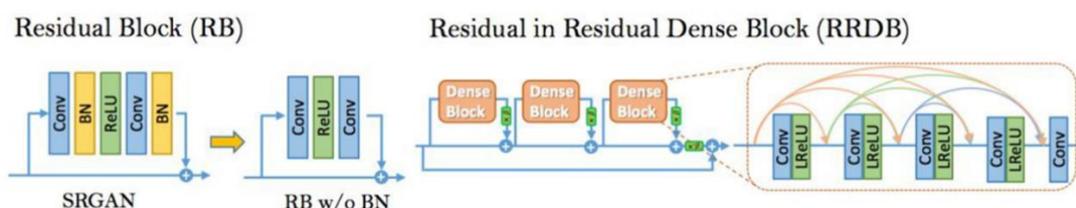
超解析度生成對抗網路 (SRGAN) 是一項開創性的工作，能夠在單一圖像超解析度中生成逼真的紋理。這項工作發表於 CVPR 2017。但是，放大後的細節通常伴隨著令人不快的偽影。為了更進一步地提升視覺質量，作者仔細研究了 SRGAN 的三個關鍵部分：1) 網路結構；2) 對抗性損失；3) 感知域損失。並對每一項進行改進，得到 ESRGAN。

具體而言，提出了一種 Residual-in-Residual Dense Block (RRDB) 的網路單元，在這個單元中，去掉了 BN (Batch Norm) 層。此外，借鑒了 Relativistic GAN 的想法，讓判別器預測圖像的真實性而不是圖像“是否是 fake 圖像”。

ESRGAN 的改進可分為三點：

1. 網絡的基本單元從基本的殘差單元變為 Residual-in-Residual Dense Block (RRDB)
2. GAN 網絡改進為 Relativistic average GAN (RaGAN)
3. 改進感知域損失函式，使用激活前的 VGG 特征，這個改進會提供更尖銳的邊緣和更符合視覺的結果

生成器的結構如下：



對於提升 SRGAN 重構的圖像質量，作者主要對生成器 G 做出如下改變：

1. 去掉所有的 BN 層
2. 把原始的 block 變為 Residual-in-Residual Dense Block (RRDB)，這個 block 結合了多層的殘差網絡和密集連接

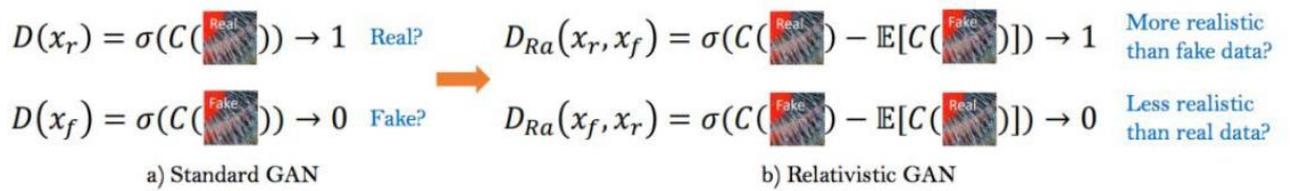
對於不同的基於 PSNR 的任務（包括超解析度和去模糊）來說，去掉 BN 層已經被證明會提高表現和減小計算複雜度。BN 層在訓練時，使用一個 batch 的資料的均值和方差對該 batch 特征進行歸一化，在測試時，使用在整個測試集上的資料預測的均值和方差。當訓練集和測試集的統計量有很大不同的時候，BN 層就會傾向於生成不好的偽影，並且限制模型的泛化能力。BN 層在網絡比較深，而且在 GAN 框架下進行訓練的時候，更會產生偽影。這些偽影偶爾出現在迭代和不同的設置中，違反了對訓練穩定性能的需求。所以為了穩定的訓練和一致的性能，去掉了 BN 層。此外，去掉 BN 層也能提高模型的泛化能力，減少計算複雜度和記憶體占用。

使用一些技巧來訓練深層網路：

1. 對殘差訊號進行 scaling，即將殘差信息乘以一個 0 到 1 之間的數，用於防止不穩定
2. 更小的初始化，作者發現當初始化引數的方差變小時，殘差結構更容易進行訓練

判別器的部分也基於 Relativistic GAN 進行了改進。判別器 D 使用的網絡是 VGG 網絡，SRGAN 中的判別器 D 用於估計輸入到判別器中的圖像是真實且自然圖像的概率，而 Relativistic 判別器則嘗試估計真實圖像相對來說比 fake 圖像更逼真的概率。

如圖所示：



(三)、PSNR (Peak Single-to-Noise Ratio)

峰值訊號與雜訊比 (Peak Signal-to-Noise Ratio)，一般我們都會講它的縮寫—PSNR。PSNR 是一種用來表示訊號最大可能功率和影響它表示精度的破壞性雜訊功率的比值，而在影像裡面我們就可以用 PSNR 這種比較客觀(有一個量化數據)的方法來計算影像的失真。

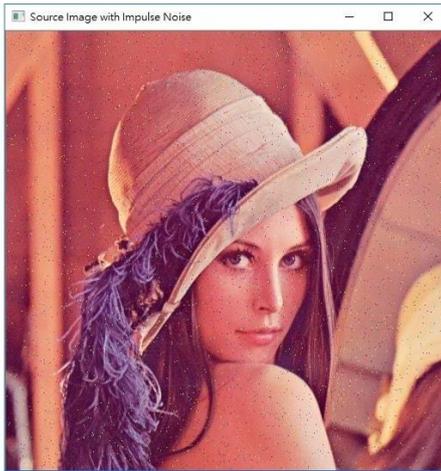
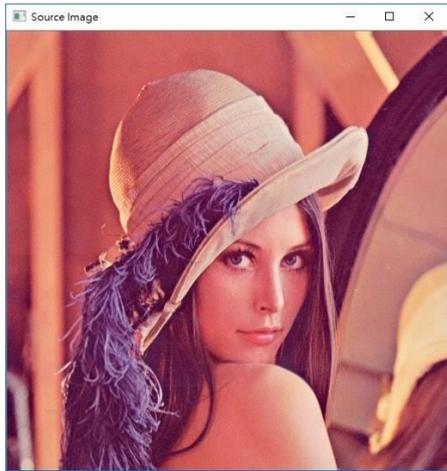
像是在影像壓縮，有分為失真壓縮跟非失真壓縮。如果是失真壓縮的話，影像在經過壓縮、解壓縮之後，必定會有一些資訊遺失。這時候我們就會想知道，解壓縮出來的圖像跟原圖到底差多少？進而去評估這個壓縮算法好不好。其他還有像是在資訊隱藏 Data hiding 的領域也會用到 PSNR。

PSNR 的定義如下：

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) = 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right)$$

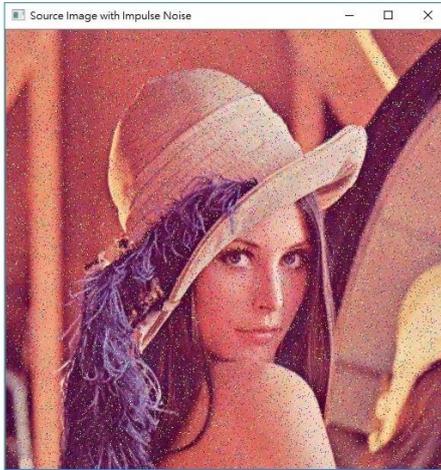
看到 log 之後你大概就知道這東西的單位是用一分貝(dB) 來算。其中的 MAX I 為訊號的最大強度，在每個 pixel 點用 8-bit 表示的影像裡面就是 255。MSE 就是統計學裡面講的均方誤差(mean-square error)。

最後我們用在【影像處理】雜訊與濾波 Noise and Filter 中所提到的 Impulse noise 為例，來計算在圖像中灑上濃度為：1%、5%、10%、20%、50%、100%，其 PSNR 值各為多少，詳細結果可參考下方圖片。



Impulse Noise
Pa: 0.005
Pb: 0.005
PSNR: 25.2171

Jason Chen's Blog



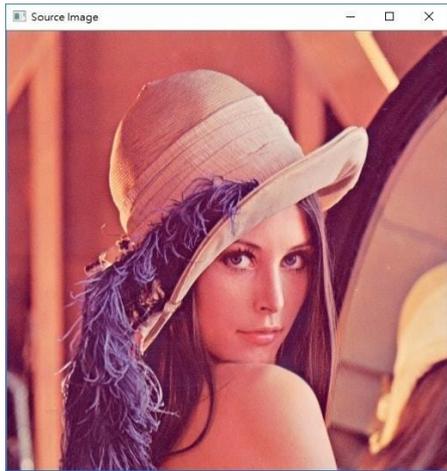
Impulse Noise
Pa: 0.025
Pb: 0.025
PSNR: 18.1606

Jason Chen's Blog



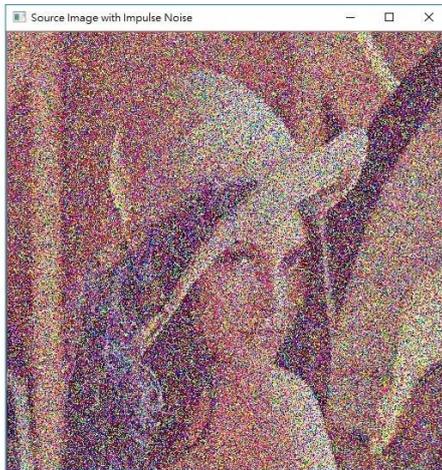
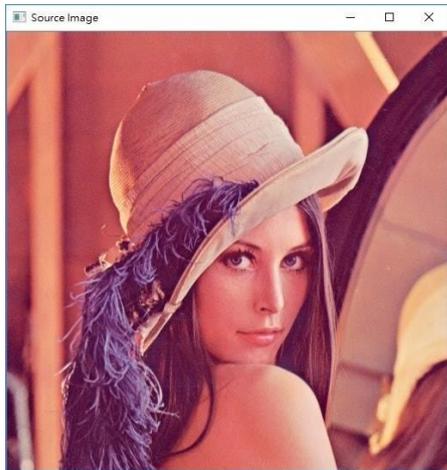
Impulse Noise
Pa: 0.05
Pb: 0.05
PSNR: 15.1875

Jason Chen's Blog



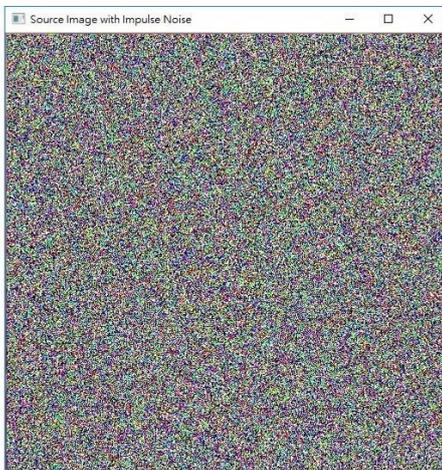
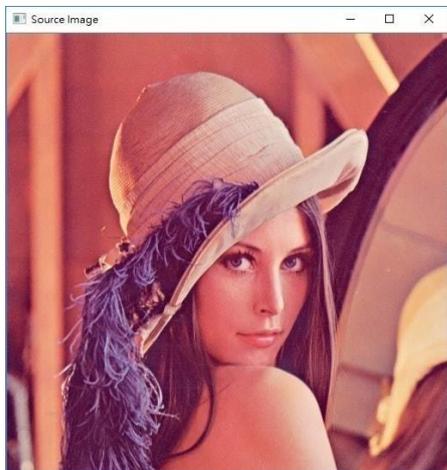
Impulse Noise
Pa: 0.1
Pb: 0.1
PSNR: 12.1646

Jason Chen's Blog



Impulse Noise
Pa: 0.25
Pb: 0.25
PSNR: 8.19091

Jason Chen's Blog



Impulse Noise
Pa: 0.5
Pb: 0.5
PSNR: 5.18667

Jason Chen's Blog

三、個案研究

(一)、資料集介紹

本次 Project 所使用的 dataset 是現有的資料，原始檔案有 900 張清晰之照片，而每一張的照片內容都不相同、照片的大小也不相同，照片主題從小狗、小貓、浣熊之類的小動物，到高樓大廈、寺廟、歷史遺跡等建築物的照片也有，充分展現了此數據集的多樣性。

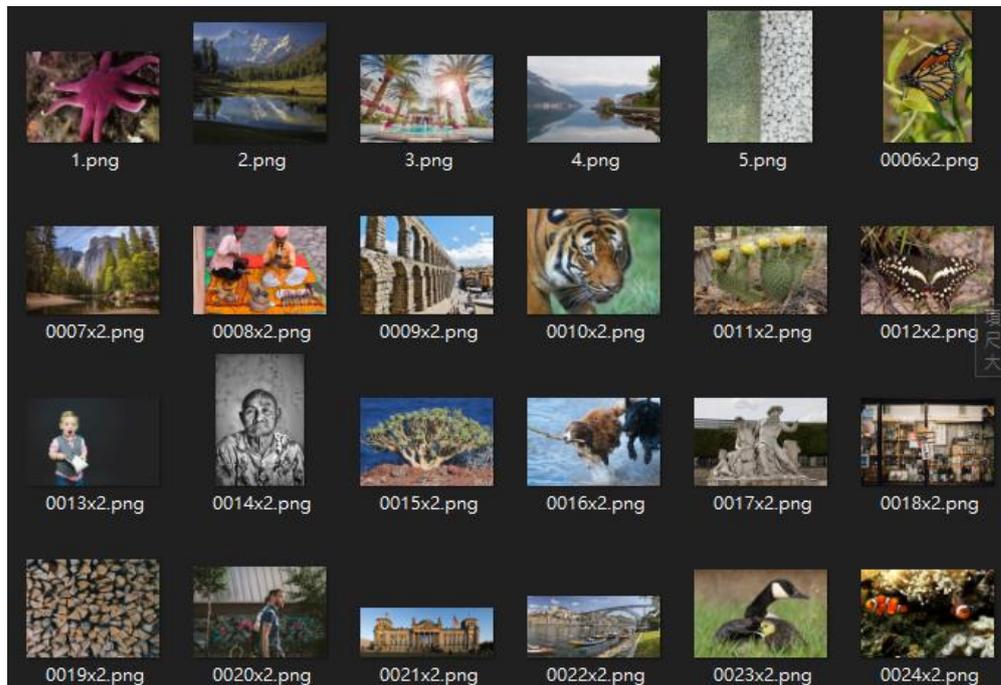
以下為一些圖示的範例：





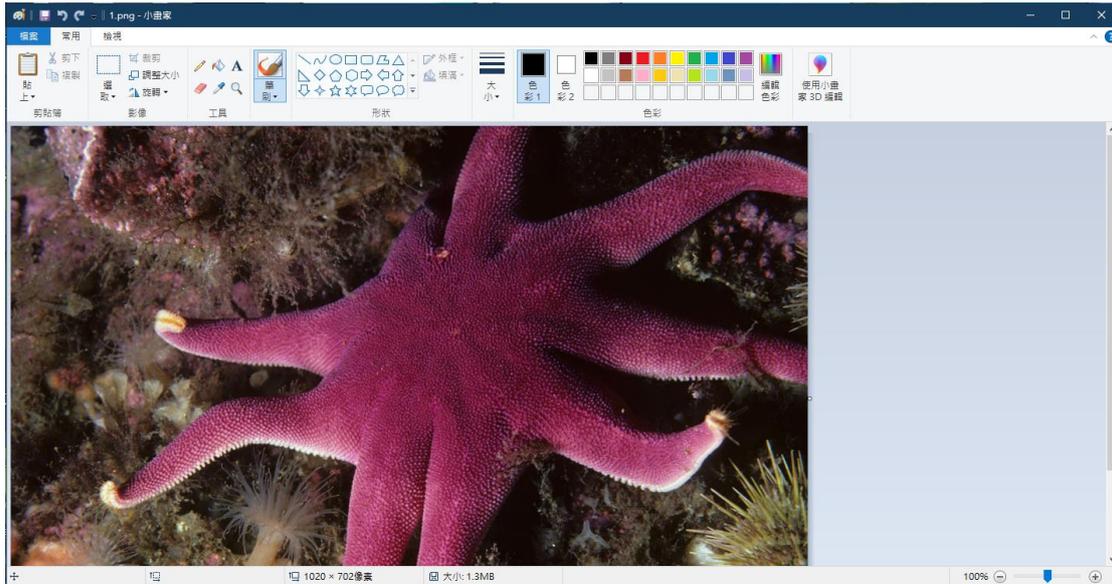
(二)、資料前處理

因為本次 Project 的主題為將模糊的影像清晰化，在與原始圖像進行比對，所以必須將原始 div2K 數據集的圖像進行模糊化，首先需要將 900 張數據集的照片進行編號，如下圖所示：

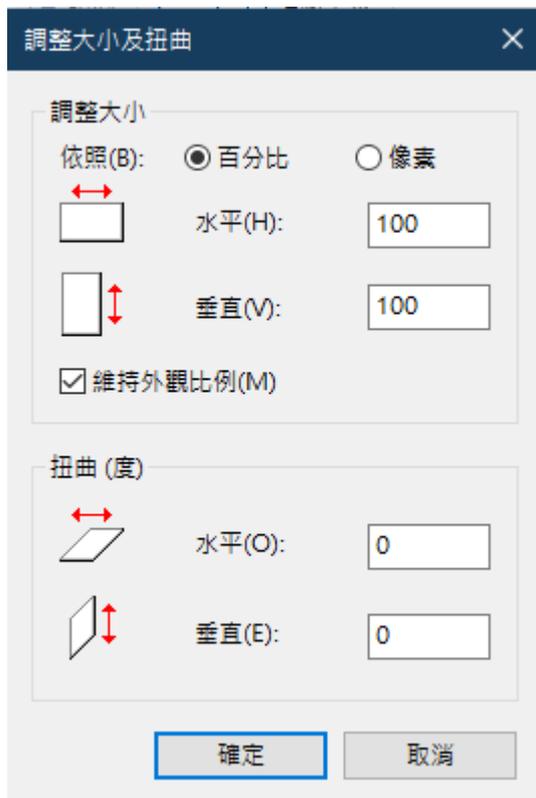


這些照片為清晰之照片，先放入一個標記為 HR 的資料夾進行存放。

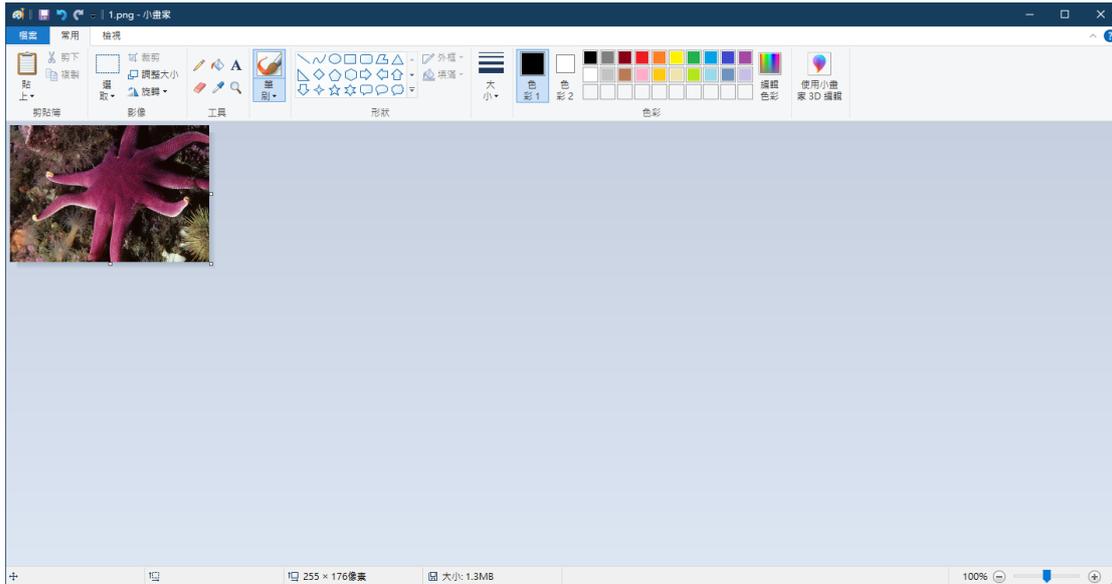
第二步將這些照片逐一進行模糊化，使用的方法為將原始圖像進行等比例縮小四倍，在不損失其原始畫質的情形下，下圖為操作範例：



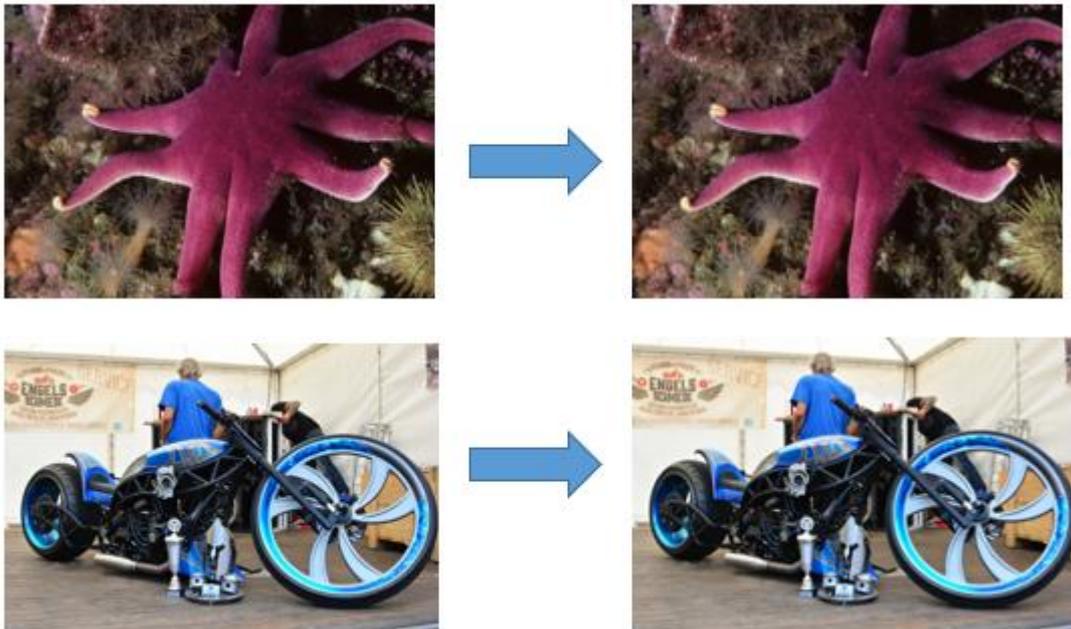
先將原始圖像拉進小畫家的介面中，並點選上方調整大小的選項



在水平及垂直的項目中，將 100 調整為四分之一的大小也就是 25，此時圖像就會縮小為原本的四分之一倍。



接著將剩餘的照片進行相同的步驟，就會完成一個新的經過模糊化的數據集，將這些照片放入一個標記為 LR 的資料夾中，因為將原始圖像等比例縮小四分之一，所以在將修改後的照片放大成與原始圖像相同時，其畫素會相對模糊，下圖有幾張範例可以說明：



最後將模糊化之圖像以 8:1:1 的比例分為訓練集、validation 及測試集，用於 SR-NET 及 ESRGAN 之訓練。

(三)、SR-NET 模型建立

本專題的主題為 SR-NET 與 ESRGAN 生成圖像進行比較，在上面的章節中以

完整介紹了ESRGAN，在此也說明SR-NET的架構。

人臉超分辨(Face super-resolution (SR))指的是從低分辨率(HR)的人臉圖像中解析出相應的高分辨率(LR)人臉圖像，當前大多數人臉超分辨方法都是通過CNN來實現。

BN對於優化神經網路，加快訓練速度甚至在提高準確度、降低損失度方面都能發揮積極作用，可以降低dropout的使用，當然想要取得理想的效果也需要反覆地嘗試各種組合。

其模型架構如下圖所示：

```
def down(filters, kernel_size, apply_batch_normalization = True):
    downsample = tf.keras.models.Sequential() # Sequential() 定義模型的開始
    downsample.add(layers.Conv2D(filters, kernel_size, padding = 'same', strides = 2)) # 2D 卷積層
    if apply_batch_normalization:
        downsample.add(layers.BatchNormalization())
    downsample.add(keras.layers.LeakyReLU())
    return downsample

def up(filters, kernel_size, dropout = False): # dropout 對抗過擬合的正則化方法
    upsample = tf.keras.models.Sequential()
    upsample.add(layers.Conv2DTranspose(filters, kernel_size, padding = 'same', strides = 2)) # Conv2DTranspose 反卷積, 特徵還原成圖片
    if dropout:
        upsample.dropout(0.2)
    upsample.add(keras.layers.LeakyReLU())
    return upsample
```

首先先定義下採樣的部分，下採樣不論是filter, kernel size的部分都可在後面實驗設計時進行改動，而padding的部分為預設值0，步伐的部分設置為2，下採樣的部分是將模糊化的圖像中透過卷積層進行特徵提取；在上採樣的部分，上述的參數也能夠調整，也在後面的實驗設計中會進行介紹，Conv2DTranspose違反卷積，其目的是將剛剛採集的特徵還原成圖片，而此專題就是要將模糊化的特徵提取，在轉為更高畫質的圖像。

而下圖為完整的模型架構：

```

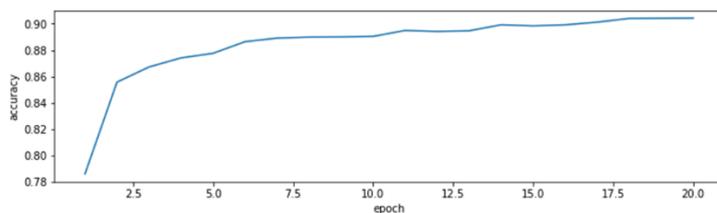
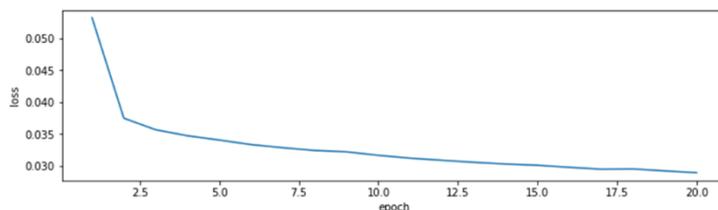
def model():
    inputs = layers.Input(shape= [SIZE, SIZE, 3])
    d1 = down(128, (3, 3), False) (inputs)
    d2 = down(128, (3, 3), False) (d1)
    d3 = down(256, (3, 3), True) (d2)
    d4 = down(512, (3, 3), True) (d3)
    d5 = down(512, (3, 3), True) (d4)
    #upsampling
    u1 = up(512, (3, 3), False) (d5)
    u1 = layers.concatenate([u1, d4])
    u2 = up(256, (3, 3), False) (u1)
    u2 = layers.concatenate([u2, d3])
    u3 = up(128, (3, 3), False) (u2)
    u3 = layers.concatenate([u3, d2])
    u4 = up(128, (3, 3), False) (u3)
    u4 = layers.concatenate([u4, d1])
    u5 = up(3, (3, 3), False) (u4)
    u5 = layers.concatenate([u5, inputs])
    output = layers.Conv2D(3, (2, 2), strides = 1, padding = 'same')(u5)
    return tf.keras.Model(inputs=inputs, outputs=output)

```

(四)、參數優化

	Level 1	Level 2	Level 3
Learning rate	0.01	0.001	0.0001
Optimizer	AdaDelta	Adam	Adagrad
Activate function	Tanh	ELU	ReLu
Filters	32	64	128

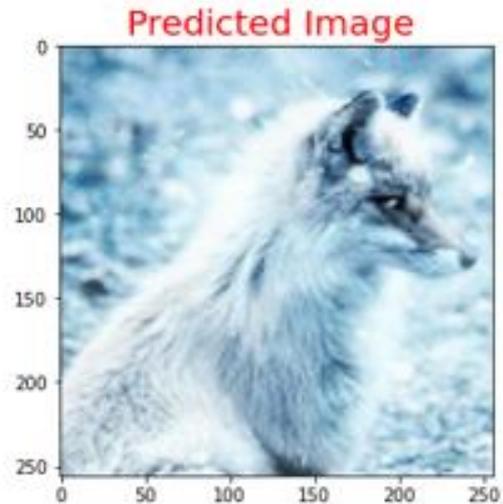
	Learning rate	Optimizer	Activate function	Filters	Accuracy	Loss
1	0.01	RMSprop	Tanh	32	0.4665	0.8670
2	0.01	Adam	Sigmoid	64	0.3716	0.7261
3	0.01	Adagrad	ReLu	128	0.8505	0.0385
4	0.001	RMSprop	Sigmoid	128	0.9037	0.0282
5	0.001	Adam	ReLu	32	0.8930	0.0297
6	0.001	Adagrad	Tanh	64	0.8984	0.0296
7	0.0001	RMSprop	ReLu	64	0.8666	0.0353
8	0.0001	Adam	Tanh	128	0.8601	0.0363
9	0.0001	Adagrad	Sigmoid	32	0.8239	0.0414



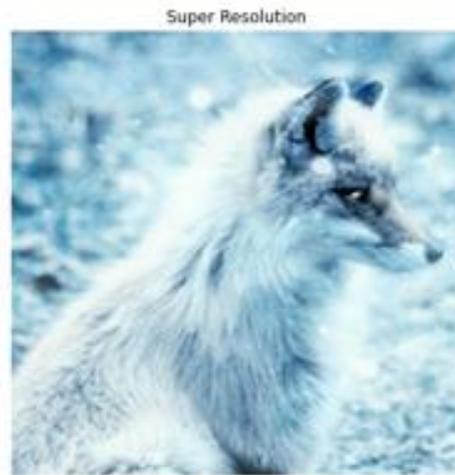
(五)、方法比較

在 SR-NET 中選定準確率與失誤率函數最優化的參數，與 ESRGAN 兩兩互相對於相同的數據集進行生成圖像進行比較，因為 GAN 的模型在調整參數的部分較難達成，所以使用原始的 pre-trained weight 直接進行生成。

在結果的部分，SR-NET 生成的圖像與原始的清晰圖像相比，其 PSNR 值大約落在 19-22 分左右，而 ESRGAN 的生成圖像與原始清晰圖像相比，其 PSNR 值大約落在 28-30 分左右，在上面的章節也介紹了 PSNR，雖然在清晰化的過程會因為其圖片的細節程度而在 PSNR 值有所差異，但整體來說，ESRGAN 將模糊影像生成清晰化的圖像能力比 SR-NET 的能力較佳。



CNN生成圖像
PSNR : 19.285225



ESRGAN生成圖像
PSNR : 28.029171

在此圖可以發現，ESRGAN 相對於 SR-NET 生成的圖像，還原了更多原始圖像的細節，例如狼毛髮的末端，或是狼與背景之間的焦距也拿捏的較好。

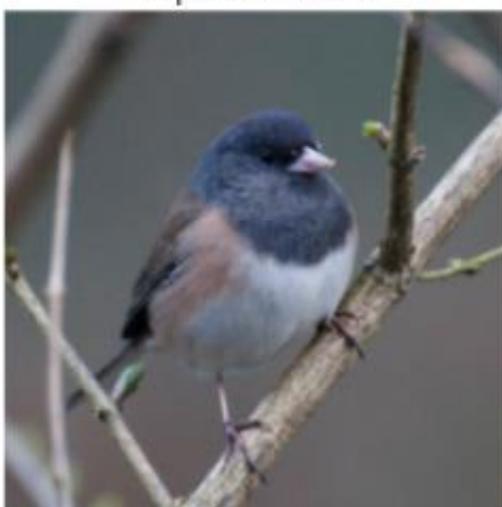
(六)、ESRGAN 優化原始圖像

ESRGAN 除了可以還原經過模糊化的照片，也可以為原本的圖像進行細節上的增加，提升原始圖系的畫質與一些相關細節，可由下圖表示：

Original Image



Super Resolution



Original Image



Super Resolution



圖像左邊為原圖，右邊為經過ESRGAN清晰化之圖像。

四、結論

(一)、貢獻

ESRGAN 在經過 div2k 數據及訓練後，也可對於一般的照片進行清晰化。
利用去除批量正規化層使照片去除原本 SRGAN 所產生的偽影瑕疵。

(二)、侷限性

這次因為時間的關係，所以只對於 CNN 進行參數調整，因為 GAN 模型的參數較難調整，因為要同時調整判別器與生成器之間的參數，如果其中一方設定的太高或太低，會導致另一方的模型崩潰，有時間上與技術上的困難。

(三)、適用性

可以應用於手機相機模組。
藉由優化原始圖像細節，對於物件偵測有更好的運用。

(四)、未來改善

GAN 模型在現今機器學習中十分熱門，也有好幾種不同的模型，可以透過更大量的資料輸入，與事實的在每個階段存檔，在模型崩潰前找出準確率與損失函數最滿意的參數調整。

五、参考文献

- Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., ... & Change Loy, C. (2018). Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops* (pp. 0-0).
- Kang, X., Liu, L., & Ma, H. (2020). ESR-GAN: Environmental Signal Reconstruction Learning With Generative Adversarial Network. *IEEE Internet of Things Journal*, 8(1), 636-646.
- Takano, N., & Alaghband, G. (2019). Srgan: Training dataset matters. *arXiv preprint arXiv:1903.09922*.
- Nagano, Y., & Kikuta, Y. (2018, July). SRGAN for super-resolving low-resolution food images. In *Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management* (pp. 33-37).