視訊壓縮期末書面報告　姓名:葉冠宏　學號:108753208

題目:High-Fidelity Generative Image Compression(NeurIPS 2020)

一. Introduction

這篇論文我們主要的目的是要用 Generative Adversarial Networks 及 auto encoder 的概念，以訓練出符合人眼視覺品質的影像壓縮系統。

我們會針對 normalization layers, generator, discriminator 和 perceptual losses 等做詳加討論。

最後，我們會對於實驗結果做進一步的詮釋。

二. Related work

1. [47] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. Full resolution image compression with recurrent neural networks. arXiv preprint arXiv:1608.05148, 2016.

其用的是 RNN 的架構，而我們這篇論文則會代入 GAN 的模型。

2. [39] Oren Rippel and Lubomir Bourdev. Real-time adaptive image compression. In Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research,pages 2922–2930, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

他雖然有用到 adversarial loss 的概念，但其並沒有對 compression 所產生的 loss 和 reconstruct 後的圖的品質做更詳細的討論。

三. Method

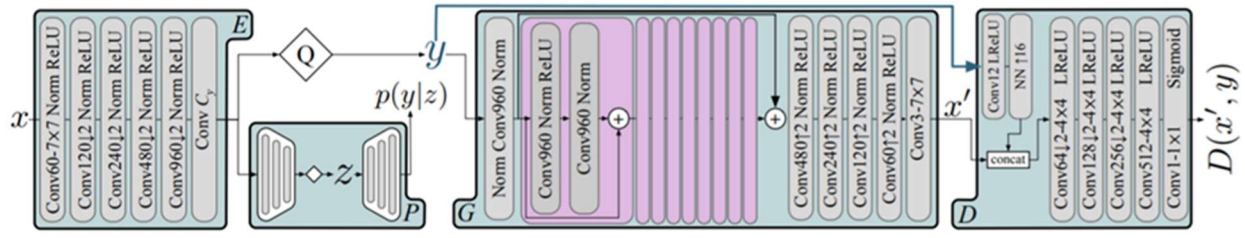

Figure 2: Our architecture. *ConvC* is a convolution with C channels, with 3×3 filters, except when denoted otherwise. $\downarrow 2$, $\uparrow 2$ indicate strided down or up convolutions. *Norm* is ChannelNorm (see text), *LReLU* the leaky ReLU [56] with $\alpha=0.2$, *NN*$\uparrow$16 nearest neighbor upsampling, Q quantization.

parameter: Adam for 2 000 000 steps

上圖是這篇 paper 所使用的架構。整個大架構來說，其使用的是 conditional GAN 的 model。而從部分架構來講，E 到 G 這中間使用的是 Auto encoder。E 代表的是 encoder，G 代表的是 Auto encoder 中的 decoder 或者也可以當作 conditional GAN 當中的 generator。D 則代表的是 conditional GAN 當中的 discriminator。G 跑出來的結果我們叫做 x'。

這裡的 P 是一個機率的 model，用 side information z 來決定 Q quantization matrix，以影響壓縮後的 feature vector y，決定壓縮的 bit 需要幾個。

最後，我們也會把 x' 和 y 一起送進 discriminator。

在這個模型中，我們使用的 Norm 是 channel norm。我們的 D 用的是 single-scale patch-discriminator。其他一些設定如 3x3 filter，leaky ReLU(alpha=0.2)，Adam ，nearest neighbour upsampling。

3-1 Conditional GANs

Conditional GAN 是一種 generative model ，基於條件分布 p(X|S)。 每個 datapoint x 都會有一個附加資訊 s ，例如 class labels 或 semantic map。x, s 是互相關聯的，基於一個聯合分佈 pX,S。

我們訓練的 G 和 D 互為 rivaling networks。我們的目的是要訓練一個模型使 G 產生的生成 image 可以欺騙過 D 使其相信為真的 image。所以最後即使經過 auto encoder 的壓縮，我們再經過 decoder reconstruct 出來的 image 如果可以騙過 D 使相信圖片為真，則代表當初 decode, encode 出來的圖片品質是好的。

因此我們可以得出以下的 loss (non-saturating loss) function，以達到我們的目的。如下:

$$\mathcal{L}_G = \mathbb{E}_{y\sim p_Y}[-\log(D(G(y,s),s)],$$
$$\mathcal{L}_D = \mathbb{E}_{y\sim p_Y}[-\log(1-D(G(y,s),s)] + \mathbb{E}_{x\sim p_{X|s}}[-\log(D(x,s))].$$

D(x)表示 D 判斷真實圖片是否真實的機率，對於 D 來說，這個值越接近 1 越好。D(G(z))是 D 判斷 G 生成的圖片的是否真實的機率，對於 D 來說，這個值越接近 0 越好，而對於 G 來說，這個值越接近 1 越好，所以最後的 optimal 會在機率為 0.5 的地方。

我們可以看到 loss of G 中 ，如果為生成 image，當 D(G(y)) 越接近 1，其 loss 越小。而在 loss of D 中，當 D(G(y)) 越接近 1，1-當 D(G(y)) 越接近 0，你所產生的 loss of D 值越大。

對 loss of D，如果你餵的是真實 image，D(x) 越接近 1，使取 log 加負號後值越小，所以 loss 越小。

3-2 Autoencoder

image 在經過 encoder E 作用後，我們得到 quantized latent y = E(x) (lossy compression)。然後我們用 decoder G 去得到 lossy reconstruction x' = G(y)。因為是 lossy compression，所以我們可以去計算 distortion，用函數 d(x, x'), e.g., d = MSE。

這裡，我們用 probability model P 和 entropy coding algorithm (e.g., arithmetic coding) 來儲存 y losslessly。即 r(y) = − log(P(y)) (bit/s)。

最後我們用 CNN 的架構來訓練 E,G 和 P。所以我們的 loss function 如下:

$$\mathcal{L}_{EG} = \mathbb{E}_{x \sim p_X} [\lambda\, r(y) + d(x, x')].$$

可以看到，當我們使用的 bits rate 越高，即 r(y) 越大，其所產生的 loss 越大，或是如果你 reconstructed 後的 image 如果用 d 的 function 判別和原來差越多，你的 loss 也會越大。所以這邊就會有一個 rate-distortion trade-off。

3-3 Loss function

在 d(x,x')中，我們是用以下的 function 來去判斷 x 和 x' 之間的差。

$$d = k_M \mathrm{MSE} + k_P d_P,$$

這邊的 dp 我們是用 LPIPS(perception distortion) 這個 metric。LPIPS 可以測量 feature space 之間的距離，且有針對預測 distorted patches 的 similarity 做調整，即可以幫助預測人眼感官對於 distortion 的 human score。

LPIPS 的模型架構如下:

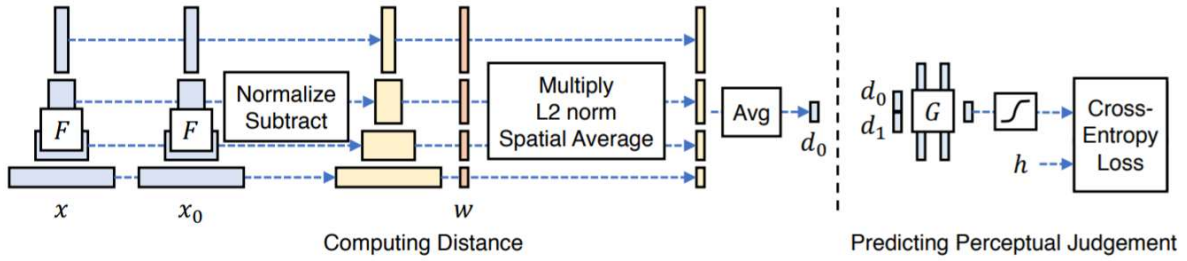

Figure 3: **Computing distance from a network** (Left) To compute a distance d_0 between two patches, x, x_0, given a network $\mathcal{F}$, we first compute deep embeddings, normalize the activations in the channel dimension, scale each channel by vector w, and take the ℓ_2 distance. We then average across spatial dimension and across all layers. (Right) A small network $\mathcal{G}$ is trained to predict perceptual judgment h from distance pair (d_0, d_1).

以下為使用 LPIPS 及沒有使用 LPIPS 的前後比較:

(a) no d_P (b) d_P = VGG (c) d_P = LPIPS

Figure 8: Effect of varying the perceptual distortion d_P. All models were also trained with an MSE loss and a GAN loss.

可以觀察到使用 LPIPS 這個 metric 之後，整個圖片變得平滑許多。

得到上述結論之後，我們可以得到以下的 loss function:

$$\mathcal{L}_{EGP} = \mathbb{E}_{x\sim p_X}[\lambda r(y) + d(x,x') - \beta \log(D(x',y))],$$
$$\mathcal{L}_D = \mathbb{E}_{x\sim p_X}[-\log(1-D(x',y))] + \mathbb{E}_{x\sim p_X}[-\log(D(x,y))].$$

這裡，我們在第一式的 loss function 之中多加了- β (log(D(x’,y))) 這一項。直覺是當你在同樣的 vector y 作用下可以經由 discriminator 判別 x’ 為更真實的圖片的話，log(D(x’,y)) 值越大，- β (log(D(x’,y))) 使 loss 下降。

得到以上式子後，我們可以發現這裡面的 hyperparameter 有 kM, kP , β。對於固定 λ，不同的

kM, kP , β 會造成模型有不同的 bitrates，使最後實驗的時候比較困難。因此，我們選定一個 “rate target” hyper-parameter rt，然後把變數 λ 替代為 adaptive term λ’。當 r(y) > rt 時，λ’ = λ(a)。當 r(y) < rt 時，λ’= λ(b)。而如果我們設定 λ(a) >> λ(b)，則最後學到的模型平均 bitrate 會接近 rt。

我們藉由這樣的 loss function 來去訓練 E, G, P, D 的 network。

3-4 Probability model P

我們用的 P model 是用以下[6] 所提出的論文架構。我們用附加資訊 z 去模擬 y 的 distribution，再加上 uniform noise U(−1/2, 1/2) 去估計 p(y|z)，藉以影響 quantization 的結果。

但當我們把 y 餵給 G 的時候，我們用 rounding 而不是 noise，因為這樣可以確保 G 看到的是一樣的 quantization noise

[6] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In International Conference on Learning Representations (ICLR),2018.

3-5 Norm

在其他有些 model 中，他們用的是 instance norm。但我們發現，當你 input image 的 resolution 不同時，其可能會有 darkening artifacts。這是由於 Instance Norm 中 spatial averaging 所造成。其所呈現的 darkening artifacts 結果如下圖:

Instance Norm in Auto-Encoder We visualize the darkening caused by InstanceNorm in E, G (see Section 3.3) in the inset figure, where a model is evaluated on an image at a resolution of 512×512px as well as at the training resolution (256×256px). We also explored using BatchNorm [19] but found that this resulted in unstable training in our setup.

因此，我們的 model 採用的是 channel norm。他是對 channel 之間做 normalized，input 是 C×H×W dimension 的 batch fchw，附加的 parameter 是 per-channel offsets αc, βc。其公式如下:

$$f'_{chw} = \frac{f_{chw} - \mu_{hw}}{\sigma_{hw}}\alpha_c + \beta_c, \quad \text{where} \quad \mu_{hw} = 1/C \sum_{c=1}^{C} f_{chw}$$

$$\sigma^2_{hw} = 1/C \sum_{c=1}^{C} (f_{chw} - \mu_{hw})^2,$$

有關於 Instance Norm 及 Channel Norm 的示意圖如下:

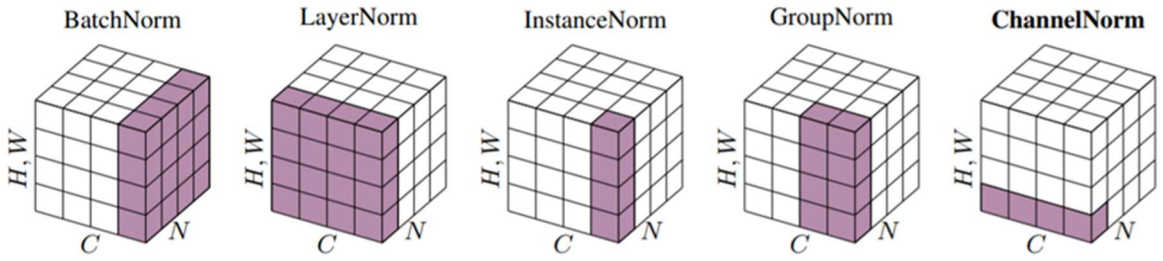

Figure A4: Visualizing which axes different normalization layers normalize over. N is the batch dimension, C the channel dimension, and H, W is the spatial dimensions. If the shaded area spans over an axis, this axis is normalized over. For example, BatchNorm normalizes over space and batches, LayerNorm over space and channels. Our normalization layer, ChannelNorm, normalizes over channels only. *Figure adapted from [55].*

四. Experimental results

如圖為我們針對不同 model 以及評估的指標所呈現的結果:

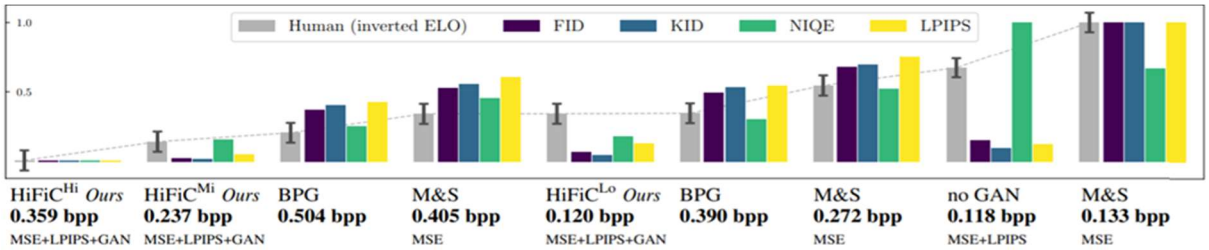

Figure 3: Normalized scores for the user study, compared to perceptual metrics. We invert human scores such that **lower is better** for all. Below each method, we show *average* bpp, and for learned methods we show the loss components. "no GAN" is our baseline, using the same architecture and distortion d as *HiFiC (Ours)*, but no GAN. "M&S" is the *Mean & Scale Hyperprior* MSE-optimized baseline. The study shows that training with a GAN yields reconstructions that outperform BPG at practical bitrates, for high-resolution images. Our model at 0.237bpp is preferred to BPG even if BPG uses $2.1\times$ the bitrate, and to MSE optimized models even if they use $1.7\times$ the bitrate.

我們所用的 model 是 HiFiC。而在當中，我們可以設定 rt 為 0.14 或 0.3 或 0.45，分別用 HiFiClo,HiFiCmi, HiFiChi 所代表。

M&S 是一個 deep-learning based ，non-autoregressive 的 model，其所用的 probability model P 和我們的 model 一樣，但其用的是 mean squared error 來訓練 loss function，由 Minnen et al 所提出。

BPG 是一個 non-learned ，based on H.265 的 codec。

No GAN 是我們的 baseline, 除了 GAN 以外，其他部分和我們的模型一樣。

從上圖中，我們可以看到有不同顏色的指標，這些指標值越低，代表 reconstructed image 和原來 image 看起來的差距越小，代表越好。所以從圖表中，我們看到我們的 model HiFiC 的表現是來的比其他 model 越好的。除此之外，我們也用了更小的 Bpp(bits per pixel) 去壓縮，表現更好。而 HiFiClo, HiFiCmi, HiFiChi 三者來看，當我們可以允許較高的 rate target 值，允許較多 bit 去壓的話，其表現也會越好。我們也注意到，即使沒有用 GAN，但如果有用到 LPIPS 的 metric，其表現仍有進步。

下圖為我們所做的實驗結果，可以看到在經過我們的模型之後，壓縮後再 reconstructed 出來的 image，其人眼視覺上和原來幾乎看不出差異。

Original image　　HiFiC Hi- bpp:0.503　　HiFiC Mi-bpp:0.322　　HiFiC Lo-bpp:0.162

如下圖，我們可以觀察到，同樣的 β 下，當你的 bit rate 越少，對應的也必須犧牲你的 distortion rate，所以你的 distortion 越高，Fid 上升(越少代表越好)。而在不同的 β 下 ，我們可以看到當 β 越大，代表我們越重視 reconstructed 的 quality，所以 Fid 會較少。

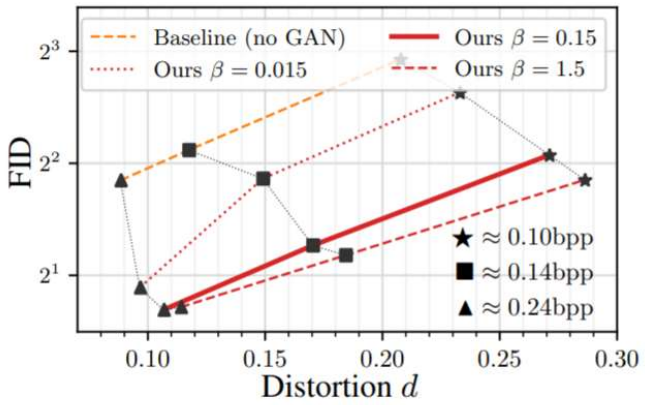

Figure 5: Distortion-perception trade-off.

五. Conclusion

在這篇 paper 之中，我們使用了 GAN 及 Auto encoder ，以及 probability model，和 LPIPS metric 的概念。最終所達到的結果為，我們用了更少的 bits rate 卻達到了更好的 reconstructed image quality。

Reference:

https://arxiv.org/abs/2006.09965

https://arxiv.org/pdf/1801.03924.pdf

https://github.com/tensorflow/compression/tree/master/models/hific