DLCV HW4 Report

R11921078 李鎮宇

1. (15%) Please explain:

A. the NeRF idea in your own words

NeRF 主要解決的問題是視圖合成(view synthesis)，也就是透過多張不同拍攝角度的影像合成場景新視圖的技術。使用的方法分成兩部分，第一部分是沿著相機拍攝的射線進行空間座標點的採樣，將這些採樣點座標透過 MLP 映射得到對應的 RGB 和 density，形成 neural radiance field。第二部分將這些帶有 RGB 和 density 資訊的座標點，透過 volume rendering 的計算合成出不同視角下的二維影像。

B. which part of NeRF do you think is the most important

我覺得是 neural radiance field 的設計，包含裏頭的 positional encoding，因為它能讓模型學習到高頻的資訊，再者透過 MLP 將這些點投射到 RGB 和 density，可以很好地表示場景在空間中的幾何結構以及外觀，後續計算出的影像也能和原圖像計算 loss 以訓練模型。

C. compare NeRF's pros/cons w.r.t. other novel view synthesis work

Pros.

與一些傳統視圖合成的方法(ex. Multi-View Stereo)相比，NeRF 可以透過參數化的方式定義 ray，進而學習不同光線與視角下場景的細微變化。

Cons.

在場景表達的部分，因為 DVGO 和 Instant NGP 都使用 voxel hashing 的方式 encoding 每個座標點，讓每個座標點帶有的特徵比 NeRF 用 positional encoding 來得多，因此後續 NeRF 要使用 8 層 256 維的 MLP 預測每個點的 RGB 和 sigma，而 DVGO 和 Instant NGP 可以只使用很少層的 MLP，所以運算速度相對 NeRF 快很多。

2. (15%) Describe the implementation details of *your NeRF model* for the given dataset. You need to explain your ideas completely.

Neural Radiance Field Scene Representation (nerf.py)

將空間中每一條射線上所有採樣點的 xyz 利用 L=10 的 positional encoding 轉換成高維資訊輸入給 MLP，MLP 的架構我使用的是 link 2，和論文提出的架構相同[1]，如圖 1 所示，有 8 層的 Linear layer，每一層的輸出都是 256 維，並加上 ReLU 作為 activation。在第 5 層輸入的時候加入 skip connection，減少梯度消失的問題。經過 8 層 linear layer 後再接一層(256, 256)的 layer，輸出的結果後續分成兩條路，一條是經過(256, 1)的 layer 得到 sigma(density)；另一條則是和 direction embedding(L=4 的 positional encoding) concat 在一起，經過後續的 layer 得到 RGB 的資訊。

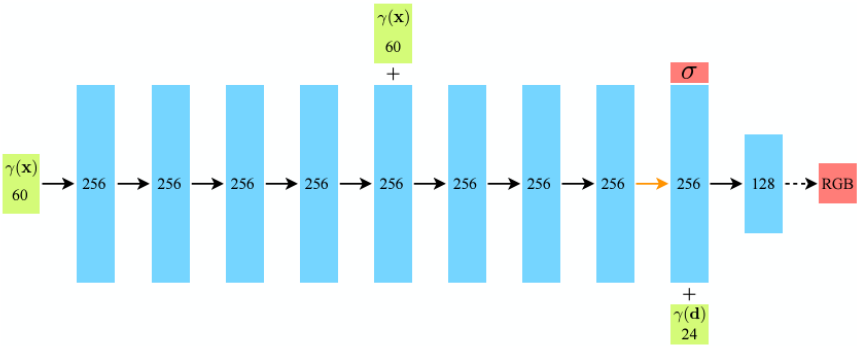


圖 1、MLP 架構[1]

Volume Rendering with Radiance Fields (rendering.py)

這邊用到方法是 ray tracing volume densities(下方公式)，ray 上的每一個點都會有自己的 RGB(以 c 表示)和 density(以 sigma 表示)，每個點利用本身的 density 從 near 積分到該點位置即可算出透光率(以 T 表示)，接著再將每一點的透光率、density、RGB 從 near 積分到 far，就能得到這個光線打到物體後在影像中呈現的顏色和光澤。

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt\,, \text{ where } T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right)$$

Coarse to Fine (rendering.py)

sigma 代表的是點的 density，可以把目前在 ray 上所有點(稱為 coarse)的 density 統計成一個 CDF，在 CDF 較大的區段表示這一段應該是有物體的部分，因此對該段再 sample 一些點(稱為 fine)，兩種點資訊分別交由兩個 MLP 去訓練，藉此優化合成的效果。

3. (15%) Given novel view camera pose from metadata.json, your model should render novel view images. Please evaluate your generated images and ground truth images with the following three metrics (mentioned in the NeRF paper). Try to use at least three different hyperparameter settings and discuss/analyze the results.

- **PSNR**

  Peak Signal-to-Noise Ratio，考量的是影像間的像素差異，最大像素值(MAX)和兩影像像素均方平均值(MSE)的比值，PSNR 越大越好。

- **SSIM**

  Structural Similarity Index，考量圖像的亮度(l)、對比度(c)和結構(s)，以下 x,y 代表要衡量的兩張圖，C1, C2, C3 皆為常數。

影像灰階平均值：$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$，衡量亮度：$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$。

影像灰階標準差：$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2}$，衡量對比度：$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$。

影像灰階協方差：$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$，衡量結構：$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x + \sigma_y + C_3}$。

SSIM = $l(x,y)\, c(x,y)\, s(x,y)$，SSIM 越接近 1 越好。

- **LPIPS**

  Learned Perceptual Image Patch Similarity，計算方法如圖 2 所示，將影像送入神經網路(VGG、AlexNet 等)進行特徵提取，每一層的特徵經過 w 縮放後計算 L2 norm distance，平均後得到距離 d，兩張影像的 d 訓練一個小模型 G 去學習影像間距離的感知，考慮三種不同的訓練方式進行感知判斷，分別是 lin, tune, scratch，這三者的結果整合在一起統稱為 LPIPS，LPIPS 越小越好。

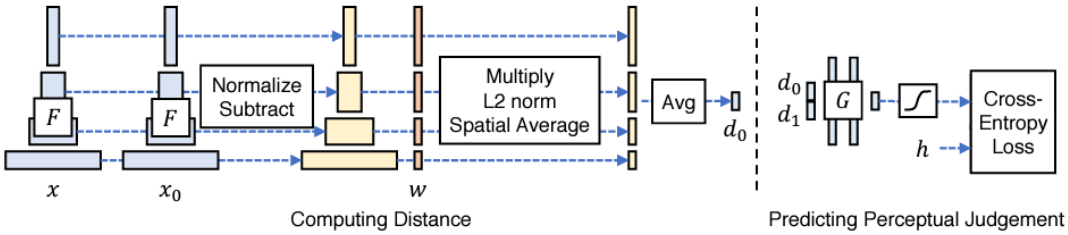
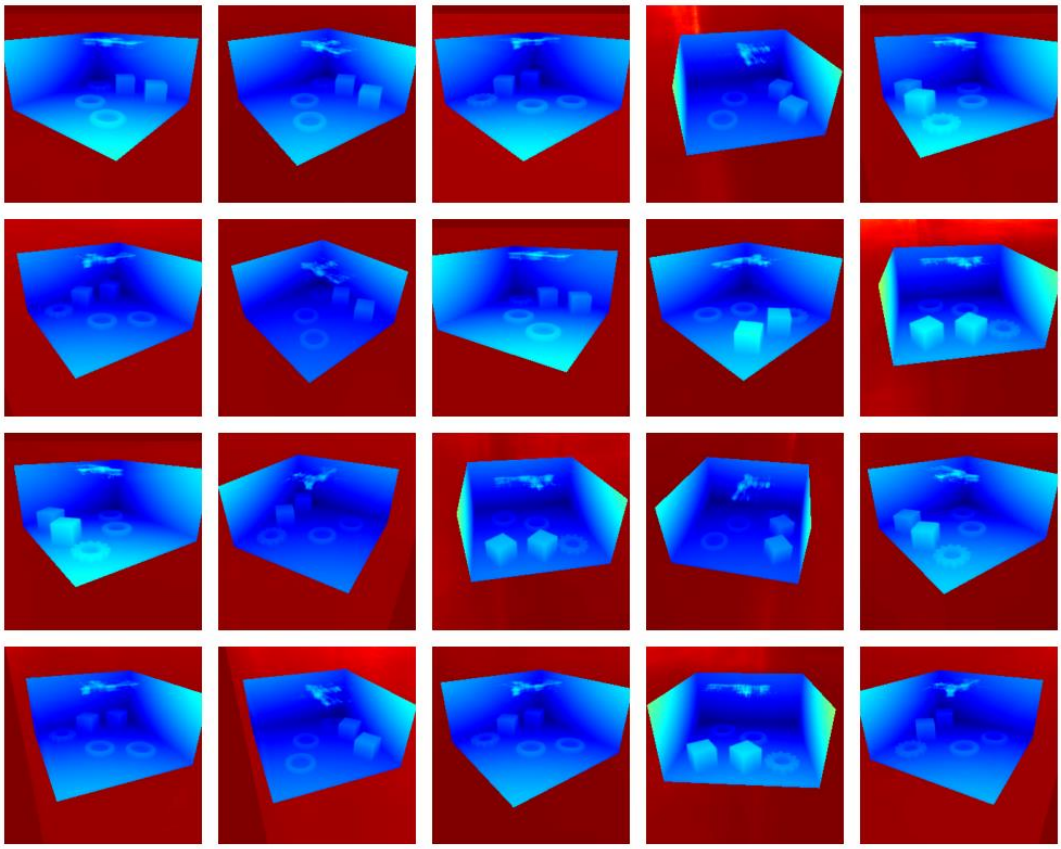


圖 2、Computing distance from a network[2]

Setting	PSNR	SSIM	LPIPS(vgg)
(exp1) N_samples: 64 N_importance: 128 skip: [4] white back: True noise std: 1	39.1382	0.9852	0.1661
(exp2) N_samples: 128 N_importance: 256 skip: [4] white back: True noise std: 1	40.5865	0.9879	0.1515

(exp3) N_samples: 128 N_importance: 256 skip: [2, 5] white back: True noise std: 1	40.6290	0.9878	0.1535
(exp4) N_samples: 128 N_importance: 256 skip: [4] white back: False noise std: 0	43.9746	0.9943	0.1002

4. **(15%) With your trained NeRF, please implement depth rendering in your own way and visualize your results.**



Reference:

[1] NeRF: Representing Scenesas Neural Radiance Fields for View Synthesis

[2] The Unreasonable Effectiveness of Deep Features as a Perceptual Metric