

Introduction to Image Processing HW2 Report

1. 作業選擇與參考論文

我這次作業是選擇以改良 HW2 模型的方式來完成本次的作業，我選擇的是助教於參考文獻中提供的第一篇論文，Implicit Neural Representations with Periodic Activation Functions，我參考此論文所提出之想法來改善我 HW2 的 NeRF 模型。

2. 作業連結

[GitHub 程式碼連結](#)、[成果與資料集連結](#)

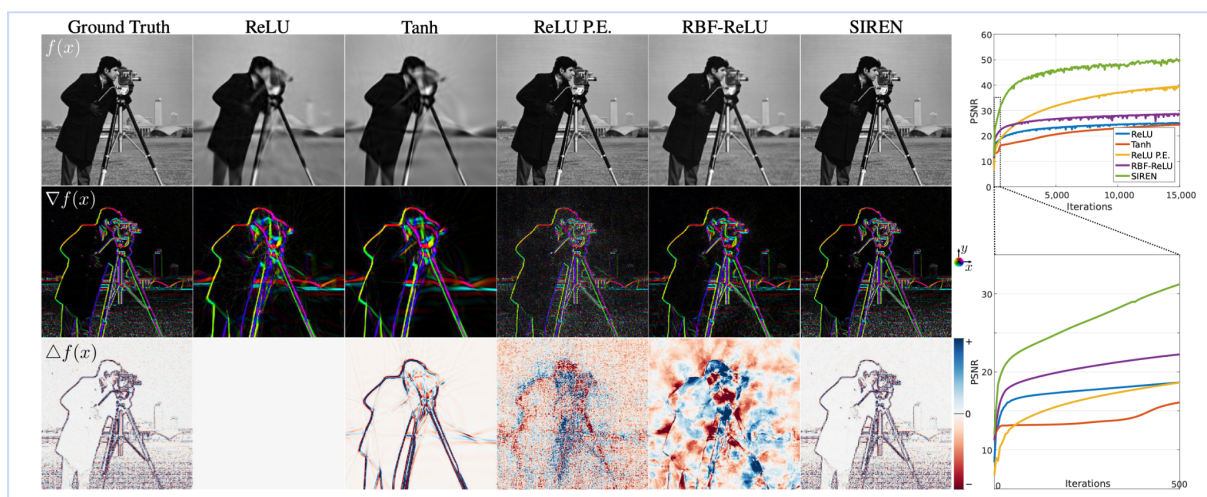
3. 論文介紹

我選擇使用 Implicit Neural Representations with Periodic Activation Functions 來改善我的模型，而此篇論文的核心概念在於使用週期函數來代替現有的激活函數 (activation function)。

論文中提到，使用隱式表示法 (Implicit Neural Representations) 來表示信號是現今常用的強大方法，但目前用於此隱式表示的神經網路架構大多無法對含有細節的訊號進行建模，但這些細節又是至關重要的。因此本論文提出了使用週期性函數，也就是 sin 或 cosine 來代替現有的像是 ReLU 等激活函數。

而為什麼使用 ReLU 當激活函數不好表示訊號中的高頻細節呢？因為 ReLU 的二次微分在每個地方都為 0，所以對於高階訊號自然不好模擬，所以為了能夠讓激活函數能夠表示訊號中的細節，論文提出使用 sin, cos 能夠讓高階導數不為 0 的週期激活函數，來提升模型模擬訊號細節的表現。

除此之外，使用週期函數 sin, cos 當激活函數也有助於模型的收斂速度，與其他激活函數的比較如下圖所示。下圖比較了不同的隱式網路架構和實際圖像的擬合狀況，右邊則是各函數的一階與二階導數。



4. 模型架構修改

在參考 [siren 的程式碼](#) 以及比對 HW2 的模型架構後，我修改 HW2 模型架構如下：

首先，於定義 NeRF 神經網路架構的 `models/nerf.py` 中宣告一個將用於取代激活函數 ReLU 的 class，我將他命名為 Siren，而此激活函數將會輸入經過 sin 函數後 forward。

```
class Siren(nn.Module):
    def __init__(self) -> None:
        super().__init__()

    def forward(self, x):
        return torch.sin(x * 1.0)
```

接著，因為 Siren 這個激活函數可以捕捉輸入訊號中的高頻細節，所以我將 NeRF 中 每層 xyz encoding layer 的激活函數都取代為 Siren (原本是 ReLU)，希望能讓 encoding layers 能夠更好的捕捉輸入訊號 (圖片經過 embedding 的結果) 的高頻細節。

```
# xyz encoding layers
for i in range(D):
    if i == 0:
        layer = nn.Linear(in_channels_xyz, W)
    elif i in skips:
        layer = nn.Linear(W+in_channels_xyz, W)
    else:
        layer = nn.Linear(W, W)
    layer = nn.Sequential(
        layer,
        # nn.ReLU(True),
        Siren()
    )
    setattr(self, f"xyz_encoding_{i+1}", layer)
```

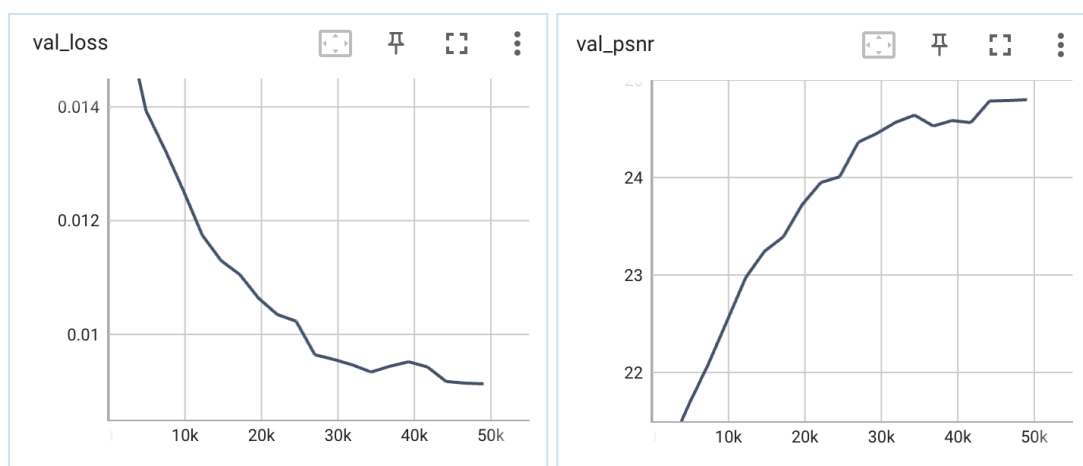
除此之外，我也將 direction encoding layers 的激活函數改為 Siren。

```
# direction encoding layers
self.dir_encoding = nn.Sequential(
    nn.Linear(W+in_channels_dir, W//2),
    # nn.ReLU(True),
    Siren()
)
```

5. 改良模型之訓練結果

Training

下圖為使用 Siren 為激活函數的模型，訓練 10 epochs 的 log，可以看到 loss 有逐漸收斂，psnr 也有上升，所以這樣的改動沒有什麼大問題！



Testing

下圖比較了原始架構(左圖)與使用 siren(右圖)各跑 10 epochs 的結果，可以發現對於地板及周圍的細節紋路(高頻細節)，使用 siren 的成效明顯較好，紋路更加清楚、顏色對比也更明顯，有成功捕捉到圖片的高頻細節，不像原始架構有點模糊。不過對於拍攝的物體表面，會出現一些噪點，導致有種解析度下降的感覺。



原始架構



Siren

除此之外，我還有測試如果將 Siren 中的 \sin 函數換成 \cos 的成果怎麼樣，下圖是原始架構、使用 \sin 函數與使用 \cos 函數各跑 10 epochs 的結果。可以發現使用 \cos 的 Siren 成效似乎介於原始架構與 \sin 的 Siren 之間，對於地板紋路細節呈現比原始架構好，但沒有使用 \sin 的 siren 好；而對於被拍攝的物體，噪點也是比原始架構高一點，但比 \sin 的 Siren 清晰。



原始架構



cos 的 Siren



sin 的 Siren

不過在觀看[合成的 GIF](#) 後，我會覺得使用 cos 的 Siren 是三者裡面表現最好的，因為除了他比原始架構保留更多細節外，他在被拍攝物體表面上的光影變化也比較自然（雖然模糊了一點點），而且又不會到使用 sin 的 Siren 那麼模糊。

6. 結果分析與討論

從結果可以發現使用 Implicit Neural Representations with Periodic Activation Functions 所提出的使用週期函數 sin, cos 當激活函數真的能夠改善模型對於細節的呈現，雖然可能會出現一些雜訊。而且對於使用的週期函數選擇，對於模型成效也會有影響。

這次的作業讓我學習到如何根據一些現有的研究改善自己的模型，我覺得蠻酷的！不過在過程還是花不少時間在研究要怎麼將論文所提的想法套用到自己的模型上，是個充實的作業！