Neural Radiance Fields

곽상열 [a] 문성현 [b]
[a] 컴퓨터공학과, 국민대학교, 서울, 대한민국
[b] 컴퓨터공학과, 국민대학교, 서울, 대한민국

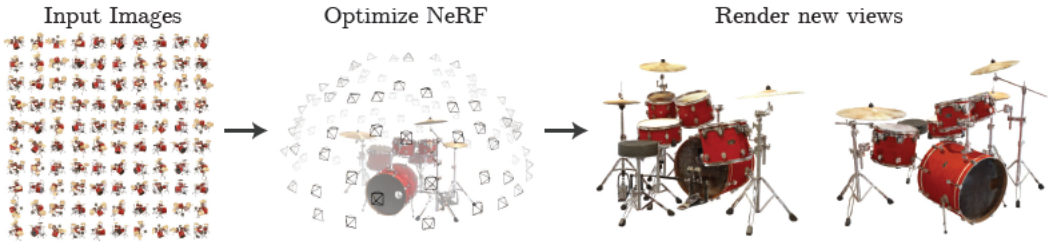

Figure 1: NeRF is a method that optimize a continuous 5D neural radiance field representation and use volume rendering to render the scene from any viewpoint.

1 Introduction

최근 이미지 생성 및 합성을 수행하는 View Synthesis 영역에서는 3차원 상의 제한된 시점에서 얻어진 이미지들을 통해 새로운 시점에서의 영상을 생성해내는 기술이 각광받고 있다. Representing Scenes as Neural Radiance Fields for View Synthesis 통칭 Neural Radiance Fields(NeRF)라고 불리는 이 기술은 이전까지 View Synthesis에서 보여주었던 기법과는 다른 새로운 기법을 제시하고 있다. 제한된 시점에서 수집한 이미지에서 3차원 점 (x, y, z)에서 Viewing Direction(θ, ϕ) 방향으로 뻗어나가는 Ray를 출력해주는 5차원 함수를 정의한다. 이렇게 얻어낸 5차원 함수(x, y, z, θ, ϕ)를 Convolution layer가 없는 Deep fully connected layer인 MLP(Multi Layer perceptron)의 입력으로 주어 최적화를 진행한다. 이 때 MLP Network의 output으로는 2D pixel 상에 해당하는 RGB 값과 불투명도가 출력된다. Ray 상에서 샘플을 뽑을 때에는 Uniform하게 구간을 나누고 그 사이에서 random하게 샘플을 뽑고, 각 샘플들에 대해 네트워크를 통과시킨다. 여기서 얻은 샘플들 사이의 밀도를 이용하여 밀도가 높은 부분에 대하여 추가적인 샘플링을 진행한다. 이후 이 샘플들에 대해 Volume Rendering을 진행하여 최종적인 이미지를 생성하게 된다.

2 Related Work

2.1 *Neural 3D shape representations*

최근 논문들에서는 xyz 좌표를 Signed Distance Function이나 Occupancy Fields로 mapping 하는 Deep Network를 최적화하여 continuous 3D shape의 Implicit Representation을 하는 연구를 진행하고 있다. 여기서 Signed Distance Function은 3차원 좌표와 어떠한 표면과의 가장 가까운 거리를 반환하는 함수이다. 또한 Occupancy fields란 3차원 공간상에서 장애물의 존재를 나타내는 binary random variable들의 field이다. 하지만 이러한 모델들은 GT 3D geometry 값을 알아야 한다는 단점이 있다. 그래서 이후의 모델들은 이 단점을 개선한 differentiable rendering function을 이용하는데 이 함수는 2D image만을 이용하여 neural implicit shape representation을 최적화 한다. 이러한 기술들이 복잡하고 고해상도의 geometry를 표현할 수는 있다고 하나 실제로는 간단한 모양을 낮은 복잡도로만 표현해서 찰흙같은 렌더링만 가능하게 했다. 따라서 저자는 5D radiance fields를 encoding하기 위한 네트워크를 최적화하는 다른 전략이 복잡한 장면에 대한 더 높은 해상도의 geometry와 appearance를 표현하여 사실과 같은 novel view를 render할 수 있다는 것을 보여주고 있다.

2.2 *View synthesis and image-based rendering*

View에 대한 조밀한 sampling, 즉 거의 모든 시야각에 대한 예시들이 주어지면 간단한 light field interpolation만으로도 photorealistic한 novel view를 reconstruction 할 수 있다. 컴퓨터 비전 및 컴퓨터 그래픽스 커뮤니티는 조밀한 samplig이 아닌 sparse한 sampling에서 새로운 장면을 합성하는 novel view synthesis를 이미지에서 geometry와 appearance를 예측하는 방법을 통해 수행한다. 첫 번째 방법은 diffuse 또는 view-dependent한 외관을 가진 장면의 mesh 기반 표현을 사용해 수행한다. rasterizer나 pathtracer는 경사하강법을 사용하여 입력 이미지를 재현하는 방식으로 mesh 표현을 최적화 할 수 있다. 하지만, 다른 각도에서 바라본 mesh 이미지를 합성하는 경우 mesh를 경사하강법을 통해 최적화하기는 어렵다. 또 다른 방법은 입력 RGB 이미지 셋에서 고품질의 photorealistic한 view를 합성하기 위한 volumetric representation이다. volumetric한 접근은 복잡한 모양과 다양한 물체들을 사실적으로 나타낼 수 있고, gradient에 기반한 최적화에 적합하며, mesh 방법보다 시각적인 면에서 더 안정적으로 느껴지는 경향이 있다. 더 최근의 연구들은 여러 장면의 large dataset을 사용하여 volumetric representation을 예측하는 Deep Network를 학습한 다음 alpha-compositing 또는 learned compositing ray를 사용하여 novel view synthesis를 수행한다.

3 Preliminary

Neural Radiance Fields에서 사용되는 기술은 크게 Neural Radiance Fields, Volume Rendering, Positional Encoding이다. Neural Radiance Fields는 Implicit Neural Representation의 일종이며, Volume Rendering은 Rendering의 한 방식으로 각 샘플의 RGB값과 불투명도를 이용하여 Rendering하는 방법이다. 마지막으로 Positional Encoding은 discrete한 5차원의 input을 continuous한 고차원으로 embedding해주는 Encoding 방법이다.

3.1 *Neural Radiance Fields*

Neural Radiance Fields 이전의 *CNN*과 같은 네트워크는 어떤 특정한 Spatial Size를 가지는 Regular Grid 혹은 Tensor 형태로 이미지를 표현한다. 하지만 Implicit Representation에서는 이미지를 나타내는 함수 f를 정의하고, 입력으로써 이미지 상에 좌표값을 받아서 해당 좌표값에서의 RGB값을 출력으로 내어준다. 이 Implicit Representation을 Neural Net으로 구성한 경우 Implicit Neural Representation(INR)이라고 한다. Neural Radiance Fields는 네트워크에 3차원 위치 정보 (x, y, z)와 Viewing Direction(θ, ϕ)를 입력으로 주어 각 픽셀의 RGB값과 Volume Density 값인 σ을 출력으로 하는 Implicit Neural Representation이다.

3.2 *Volume Rendering*

Volume Rendering 과정은 World Coordinate 상에서 3차원 카메라 위치 (x, y, z)와 Viewing Direction(θ, ϕ)이 주어져 Ray가 결정되었을 때, Ray 상에 존재하는 점들의 RGB값과 불투명도를 식 (1)에 적용하여 각 픽셀의 RGB값을 구한다.

$$C(r) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t)\mathbf{c}(\mathbf{r}(t), \mathbf{d}), \qquad T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right) \tag{1}$$

각 픽셀의 RGB값을 정할 때에는 각 샘플들의 RGB값에 보고자 하는 가장 가까운 점인 t_n부터 해당 샘플까지의 모든 점들의 불투명도를 반영해서 얻어지는 투명도인 T와 해당 샘플의 불투명도(밀도)를 이용한 가중치 합을 이용한다. 하지만, 실제로 적분을 할 때에는 Ray 상에 존재하는 무수히 많은 점들을 모두 고려해서 계산하는 것이 불가능하다. 따라서 Neural Radiance Fields에서는 이러한 한계점을 극복하기 위해 Ray 구간을 등간격으로 나눠 샘플링하는 Uniform Sampling 과정을 제안하고 있다. Neural Radiance Fields에서는 Uniform Sampling을 진행 하고, sampling된 점들 사이의 밀도를 근사해서 2D Image상에 존재하는 각

픽셀의 RGB값을 계산한다. Ray 상에 존재하는 유한한 개수의 점을 샘플링할 때에는 $[t_n, t_f]$에서 구간을 Uniform하게 나누고, 그 사이에서 Random Sampling을 진행한다.

$$t_i \sim \mathcal{U}\left[t_n + \frac{i-1}{N}(t_f - t_n), t_n + \frac{i}{N}(t_f - t_n)\right] \tag{2}$$

식(2)를 통해 원래는 연속적으로 나타나져야 하는 적분값을 유한한 개수의 샘플링 된 점으로 바꾸어 준다. Uniform Sampling의 과정은 아래와 같다.

- Sampling 할 유한한 N 개의 점을 사전에 정의하고 Ray상에서 가장 가까운 점 t_n과 가장 먼 점 t_f를 정의한다.
- Ray 상에서 Uniform distribution을 통해서 $[t_n, t_f]$ 구간을 등간격으로 나누고 각 구간별로 하나씩 점을 random하게 sampling한다.

그렇다면 Uniform Sampling에서는 픽셀의 RGB값을 결정하기 위해서 어떻게 샘플들 사이에 존재하는 점들의 불투명도 값을 반영할까? Uniform Sampling에서는 한 샘플이 가지는 불투명도가 다음 샘플까지 유지된다고 가정한 다음, 샘플 사이의 간격과 불투명도를 곱해주는 식 (4)를 이용하여 T를 정의한다.

$$\hat{C}(\mathbf{r}) = \sum_{i=1}^{N} T_i(1 - exp(-\sigma_i\delta_i))\mathbf{c}_i \tag{3}$$

$$T_i = \exp\left(-\sum_{j=1}^{i-1}\sigma_j\delta_j\right), \qquad \delta_i = t_{i+1} - t_i \tag{4}$$

Neural Radiance fields에서는 Uniform Sampling에 더해, 물체가 있는 영역에 대해서 조금 더 집중적으로 점들을 Sampling하는 Hierarchical Volume Sampling을 제안한다. 물체가 있는 영역을 찾기 위해서는 Uniform Sampling 된 구간 사이의 밀도를 이용한다. 밀도가 높은 영역이 물체가 존재하는 영역이라고 생각하고 그 구간에 대해서 Random Sampling을 진행한다.

$$\hat{C}_c(\mathbf{r}) = \sum_{i=1}^{N_c} w_i C_i, \qquad w_i = T_i(1 - \exp(-\sigma_i\delta_i)) \tag{5}$$

3.3 *Positional Encoding*

Positional Encoding이란 식 (6)을 이용하여 Neural Radiance Field에 입력으로 들어가는 3차원 위치,Viewing Direction 정보를 단순히 정수 형태로 값을 넣어주는 것이 아니라 sin,cos 함수를 사용해서 고차원의 값으로 mapping하는 것이다.

$$\gamma(p) = \left(\sin(2^0\pi p)\cos(2^0\pi p), \cdots, \sin(2^{L-1}\pi p)\cos(2^{L-1}\pi p)\right). \quad (6)$$

만약 인접한 픽셀 간의 RGB값의 차이가 굉장히 큰 경우 Positional Encoding 없이 MLP Network를 통과시켰을 때 출력으로 얻어지는 RGB값의 차이는 크지 않다. 하지만 Positional Encoding을 사용하면 인접 픽셀 간의 RGB값의 차이가 큰 경우에도 극명한 차이를 보이도록 Rendering 된다.

4 Implementation

Neural Radiance Fields의 구현은 크게 세가지로 나눌 수 있다. 입력에 대한 전처리, 즉 Positional Encoding을 진행하는 Preprocessing 부분, 네트워크를 통과시켜 샘플 별 RGB값과 밀도(불투명도) σ를 얻는 Network부분, Network에서 얻은 RGB와 σ를 이용하여 Volume Rendering을 하는 Postprocessing 부분이 그것이다.

4.1 *Preprocessing*

먼저 input으로 들어온 3차원상의 좌표 (x, y, z)와 Viewing Direction (d_x, d_y, d_z)에 대하여 Positional Encoding을 진행한다. 3차원상의 좌표를 encoding 할 때에는 식 (6)에 대해서 L에 10을 적용하였고, Viewing Direction을 encoding할 때에는 식 (6)에 대해서 L에 4를 적용하였다. Positional Encoding의 비교시험 결과는 Figure 2와 같다.

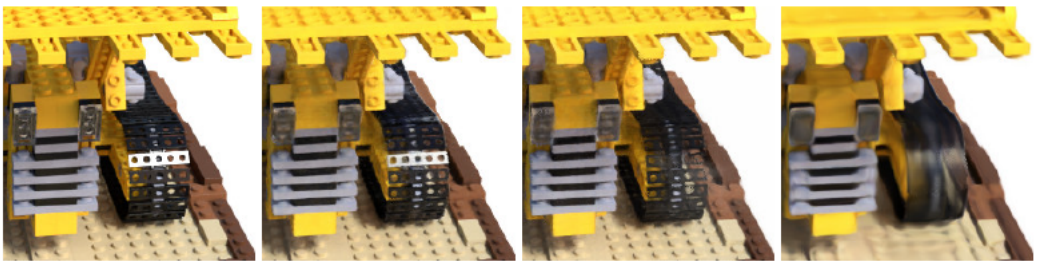

Figure 2: Visualization of full model benefits from removing positional encoding.

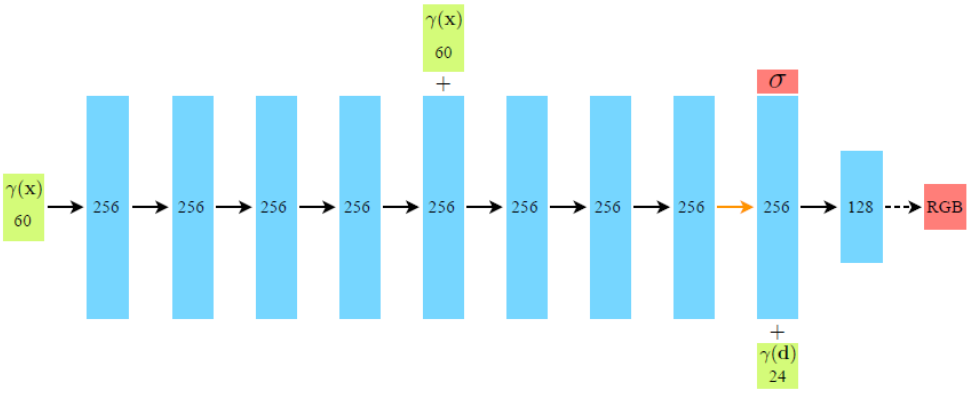

Figure 3: A visualization of fully connected network architecture.

4.2 *Network*

네트워크(Figure 3)는 MLP로 구성되어 있으며 처음부터 3차원 위치 정보와 Viewing Direction 모두를 MLP네트워크의 입력으로 넣어주는 것이 아니라 3차원 위치정보만을 MLP의 입력으로 넣어준다. MLP네트워크를 통과하면서 손실되는 정보의 양을 최소화 시켜주기 위해 skip connection을 사용하여 3차원 위치정보를 한 번 더 MLP네트워크의 넣어준다. 이후 Fully Connected layer를 통과한 값들은 Alpha Linear, Feature Linear + RGB Linear 2개의 MLP Branch를 통과한다. 이 때 그 값들 자체가 Alpha Linear를 통과하게 되면 밀도(불투명도) σ를 얻게 되고, 그 값에 Viewing Direction 정보를 concatenate 시켜 Feature Linear와 RGB Linear를 통과하게 되면 RGB값을 얻게 된다. Neural Radiance Fields에서는 Camera의 위치와 방향을 Input으로 주고 RGB과 σ를 얻는다. 이후 Hierarchical Volume Sampling을 이용하게 되는데 먼저 Implicit Neural Representation 네트워크를 통과시켜 각 Ray 상에 있는 픽셀별 RGB값과 밀도(불투명도)값을 얻는다. 이후 각 픽셀별로 Uniform Sampling을 이용하여 간격을 나누고 그 사이에서 Random Sampling을 진행하여 샘플을 뽑고 이 값들에 대해 interpolation을 진행하여 각 샘플별 RGB값과 밀도(불투명도)σ를 얻는다. 그리고 앞에서 구한 밀도(불투명도)가 높은 부분 사이에서 다시 Random Sampling을 진행하여 샘플을 뽑고 다시 interpolation을 진행하여 각 샘플별 RGB값과 밀도(불투명도)σ를 얻는다.

4.3 *Postprocessing*

Uniform Sampling에서 얻은 sample들로 Volume Rendering을 진행하는 것을 Coarse Network, Hierarchical Volume Sampling을 통해 얻은 sample들과 coarse network에서 얻은 sam-

ple들로 Volume Rendering을 진행하는 것을 Fine Network라고 한다. Coarse와 Fine Network에서 얻은 RGB값을 통해 Loss식 (7)을 이용하여 Backpropagation을 진행해 MLP 네트워크를 학습한다.

$$\mathcal{L} = \sum_{\mathbf{r}\in\mathcal{R}} \left[\left|\left| \hat{C}_c(\mathbf{r}) - C(\mathbf{r}) \right|\right|_2^2 + \left|\left| \hat{C}_f(\mathbf{r}) - C(\mathbf{r}) \right|\right|_2^2 \right] \tag{7}$$

5 Experiments

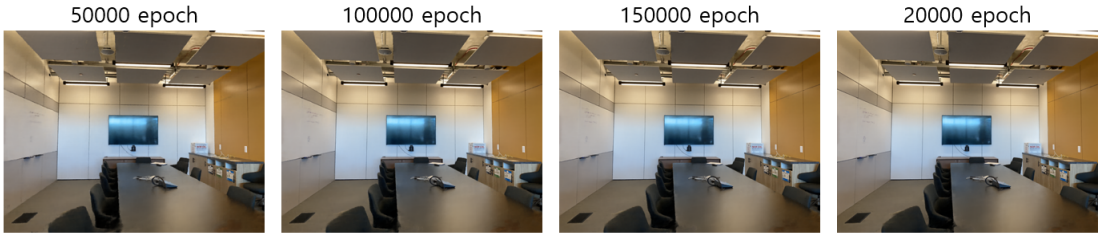

Figure 4: Results per epoch

논문에서는 여러 데이터셋을 주었는데, 그 중 Room data를 이용하여 실험을 진행하였다. 실험을 진행 할 때는 각 각도마다 41장의 room 이미지 사진을 사용해서 네트워크를 학습시켰다. 50000 epoch로 학습시켰을 때에는 의자의 팔걸이 그리고 전체적인 물건들의 재질, 박스 위에 글자들이 뭉개지는 aliasing을 관찰 할 수 있었다. Figure 4를 보면 epoch이 진행 될 수록 뭉개지는 aliasing 현상이 개선되었으며, 200000 epoch까지 학습했을 때는 뭉개지는 현상이 많이 개선되어 박스 위에 글자들이 어떤 글자인지까지 식별할 수 있게 되는 걸 볼 수 있었다.

6 Conclusion

저희 팀은 이번 프로젝트를 계기로 NeRF: Representing scenes as neural radiance fields for view synthesis라는 논문을 읽고 각 부분에 대한 저희의 분석을 작성해보았습니다. Related Work와 Preliminary의 경우 논문을 참조하여 작성하였으며, Implementation의 경우 직접 코드를 보고 이해한 내용을 작성하였습니다. Experiments를 통해 NeRF의 한계를 보았고, 이를 개선한 Mip-NeRF라는 논문도 살펴보게 되었습니다. 또한, 차후 NeRF를 이용한 연구 주제에 대한 아이디어도 얻어갈 수 있었습니다.

References

Barron, J. T., B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan (2021). Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5855–5864.

Mildenhall, B., P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pp. 405–421. Springer.