NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis

(ECCV 2020)

박세현 박일상 이수현 이은비 이지훈 정명지

Why?

- 3D 재구성 분야
  - 3D 게임, VR/AR 등의 다양한 컨텐츠의 생성으로 주변 사물의 입체화 등에 대한 관심이 높아지고 있음
- 프로젝트 주제와 관련이 있음
  - 가구를 3D로 구성하는 방법

Prior Work

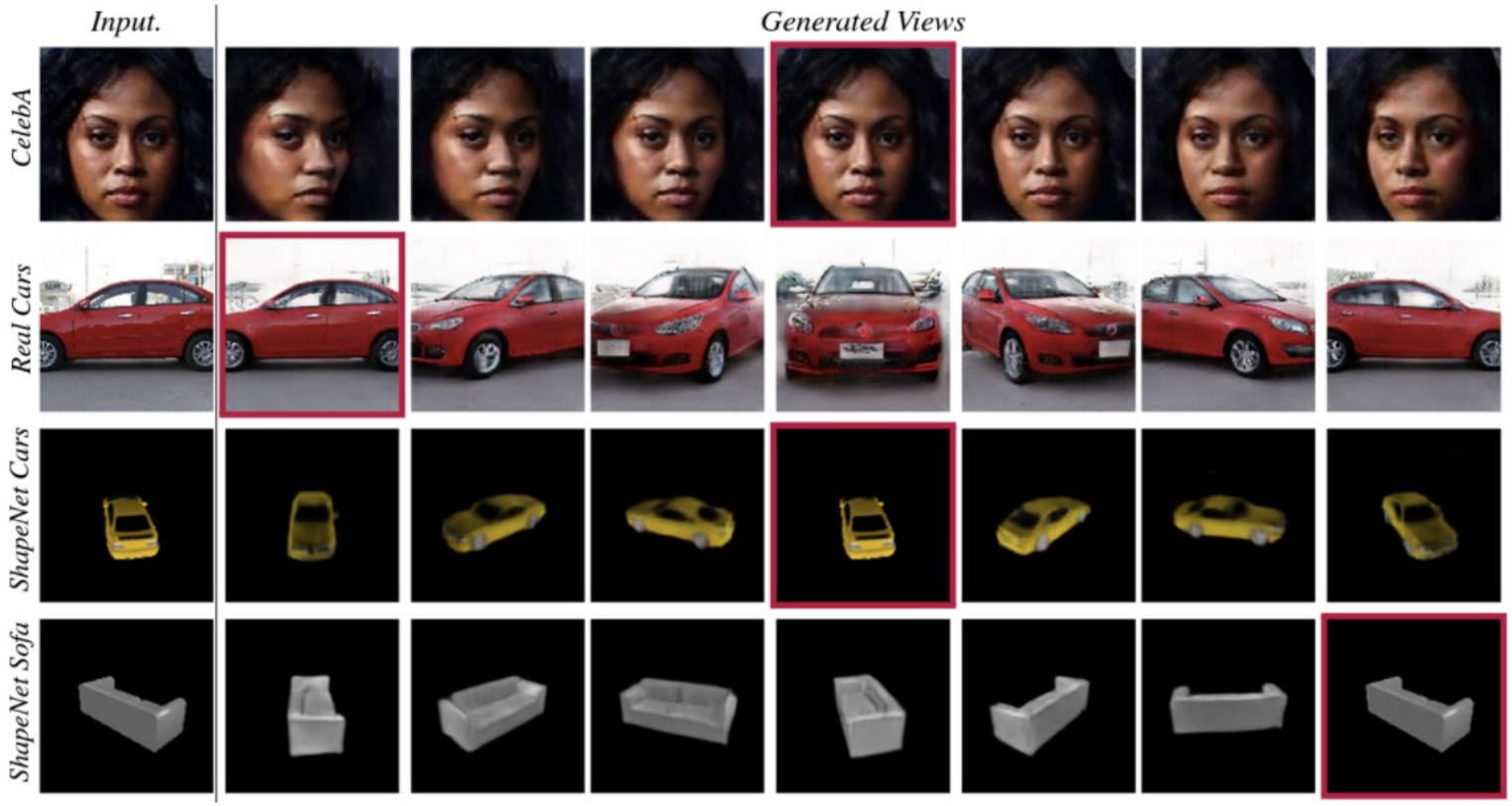
- 2D 이미지를 바탕으로 해서 3D 데이터로 만드는 작업
  - Single-view
  - Multi-view

  novel view synthesis 해결을 위해 GAN을 사용하여 2D space image를 예측하는 방식으로 진행하였음
- 3D 데이터를 바탕으로 새로운 방향의 Rendering 이미지를 만들어내는 방식
- 이전 3D representation 연구들은 3d geometry를 필요로 한다는 점에서 데이터 구축등에 어려움이 있음

Prior Work

- 2D 이미지
  - Singl
  - Mult

  novel vie
- 3D 데이터
- 이전 3D
  축등에 어



하였음

!

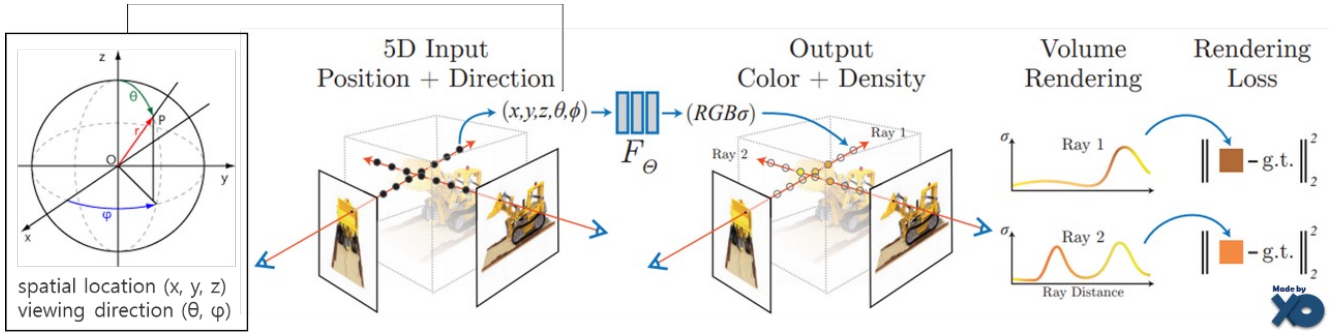
ᅦ이터 구

SK telecom
SKT FLY AI Challenger

Prior Work

- 기존 view synthesis 방법 → 3D 공간에서 특정 위치의 RGB-A값 추정 (주로 3D CNN)
  - Scalability 측면에서 단점
  - 공간복잡도와 시간복잡성이 더 나은 해상도(High Resolution)로 확장이 불가능

Neural Radiance Field Scene Representation



(a) Scene을 통과하는 ray를 따라가, 3D point의 sampled set을 생성함

(b) 위에서 생성된 3D point와 해당 2d viewing direction을 MLP의 input(5D 좌표) 으로 넣어, color, volume density를 output 산출

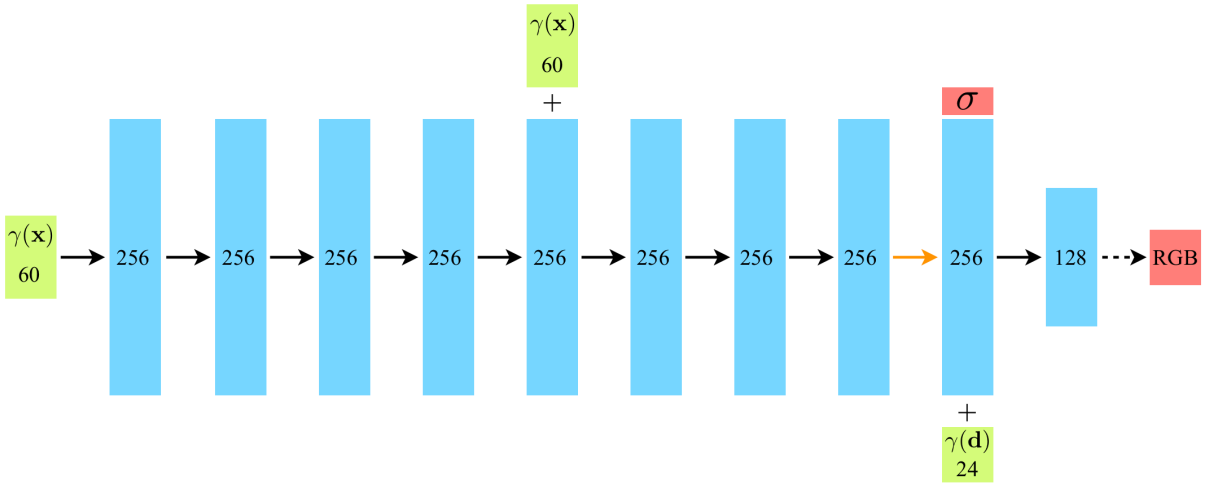
(c) 2D image에 산출된 color, density를 volume rendering 기술로 합성

(d) 위 과정은 미분 가능함. → gradient descent 사용가능

관찰된 이미지와 rendering 된 이미지의 사이를 최소화하여 모델 최적화

SK telecom
SKT FLY AI Challenger

Neural Radiance Field Scene Representation



- Input vector(초록색) / hidden layer(파란색) / output vector(빨간색) / 숫자 : 벡터의 차원 / '+' : vector concatenation
- 검은(실선) 화살표 : with ReLU activator. 주황 화살표 : no activator. 검은(점선) 화살표 : with sigmoid activator

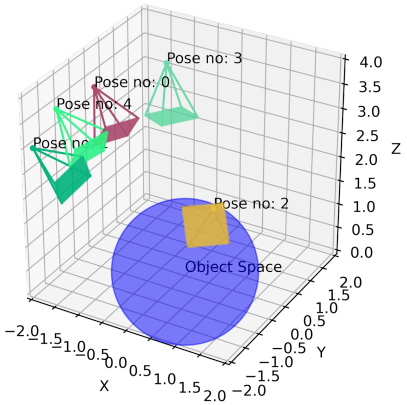
- MLP 모델 구조, fully-connected layer
- Positional encoding
  $\gamma(x)$: 60. $L = 10$ 일 때, (x, y, z) 에서 각각 20차원으로 늘려지므로 총 60차원이 됨
  $\gamma(d)$: 24. $L = 4$ 일 때, $(\theta, \phi) \rightarrow (\mathrm{X}, \mathrm{Y}, \mathrm{Z})$로 변환 후 각각 8차원으로 늘려지므로 총 24차원이 됨

- Input 값은 256채널로 구성된 fully-connected ReLU layer 8개를 지남
- view direction($\gamma(d)$)이 들어가기 전, layer output으로 256차원 feature vector와 density(σ)가 나옴
- view direction($\gamma(d)$)은 feature vector와 결합되어, 128채널의 fully-connected ReLU layer에 의해 처리됨
- 마지막 layer에서 (with sigmoid activation) d방향의 ray에서 볼 때, 위치 x에서 방출되는 RGB값을 출력함

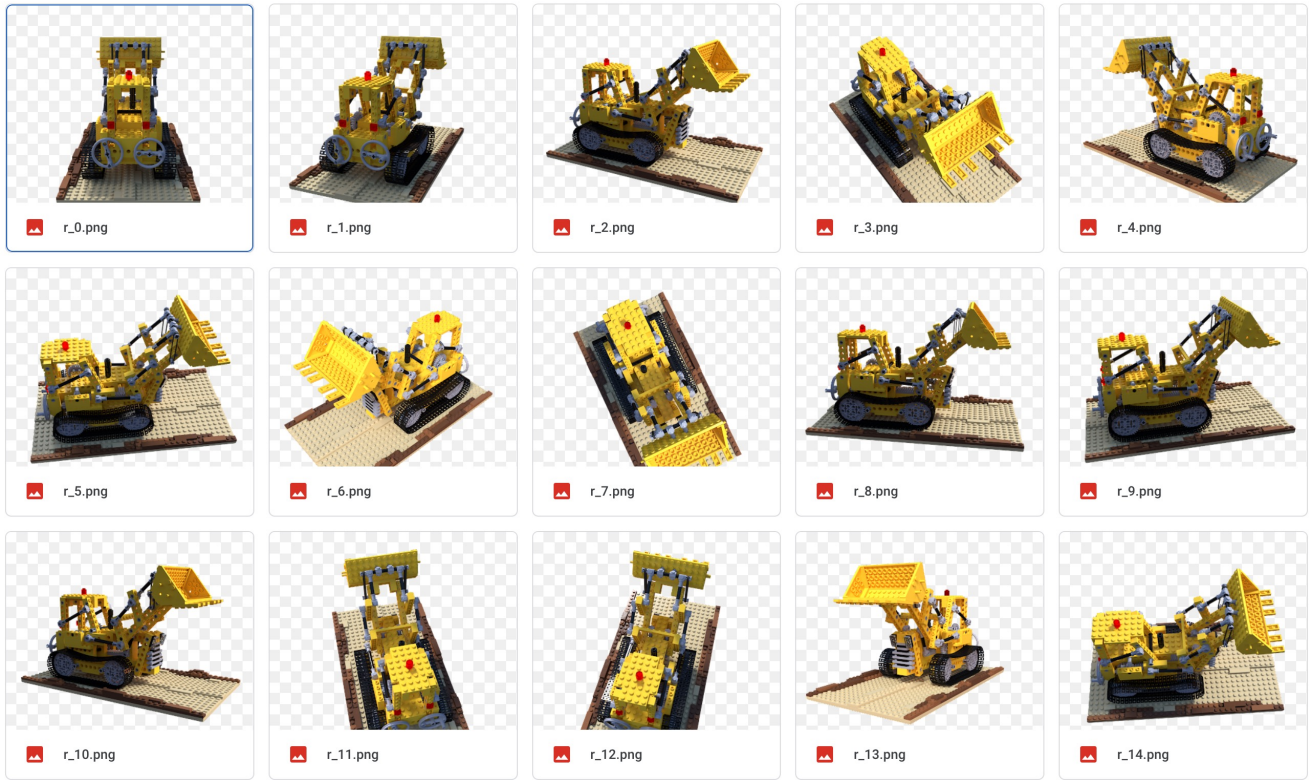
* view direction : ray는 (x, y) 그리고 ray의 방향 값인 z값을 지니고 있다. 이 ray를 normalization을 해준 것.

SK telecom
SKT FLY AI Challenger

Dataset




```
{
    "camera_angle_x": 0.6911112070083618,
    "frames": [
        {
            "file_path": "./train/r_0",
            "rotation": 0.012566370614359171,
            "transform_matrix": [
                [
                    -0.9999021887779236,
                    0.004192245192825794,
                    -0.013345719315111637,
                    -0.05379832163453102
                ],
                [
                    -0.013988681137561798,
                    -0.2996590733528137,
                    0.95394366979599,
                    3.845470428466797
                ],
                [
                    -4.656612873077393e-10,
                    0.9540371894836426,
                    0.29968830943107605,
                    1.2080823183059692
                ],
                [
                    0.0,
                    0.0,
                    0.0,
                    1.0
                ]
            ]
        },
        {
```

SK telecom
SKT FLY AI Challenger

Rendering

- Rendering이란 물체의 표면 성질과 빛의 상호 작용을 2D 평면의 RGB 픽셀 값을 결정하는 과정을 말함
- Rendering 방식

  Surface → 점, 선, 삼각형, 다각형 또는 2D 및 3D 스플라인으로 개체 표면을 표현

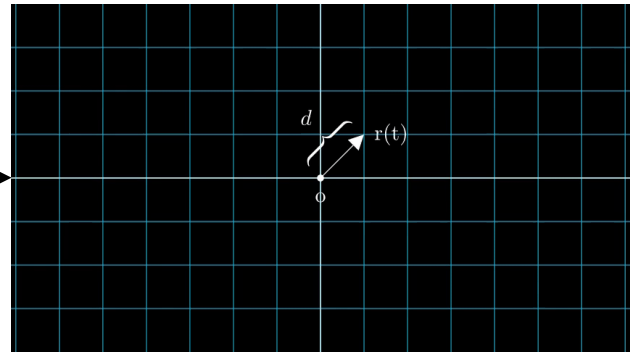
  Volume → 개체의 표면 및 내부 빛 표현 가능

Volume Rendering

- Volume rendering은 3D discretely sampled dataset을 투영시킴
- Ray casting 되는 공간의 모든 Voxel에 대한 RGBA(투명도)를 얻음
  이 결과는 RGB 색상으로 변환되어 2D 영상의 해당 픽셀에 기록되고 전체 2D 영상이 렌더링될 때 까지 모든 픽셀에 대해 반복함
- NeRF에서는 Multi-layered perceptron을 통해 RGB와 volume density를 예측하여 3D scene을 구성함

Volume Rendering

$$C(r) = \int_{t_n}^{t_f} T(t)\,\sigma(r(t))c(r(t), d)dt$$



- $C(r)$ → 객체의 어떤 점에서의 RGB
- $r(t) = o + td$ → o는 광선의 원점. d는 광선의 방향
- $\sigma(r(t))$ → Volume density 점 t에서 끝나는 ray의 미분 확률로 해석 가능
- $c(r(t), d)$ → 점 t에서의 color
- $T(t) = \exp(-\int_{t_n}^{t} \sigma\big(r(s)\big)ds)$ → 투과율

Stratified Sampling

- Sampling Rule으로 Stratified Sampling을 사용함
  - evenly spaced division에서 각각 uniform sampling 하는 것으로, MLP 가 고정된 location set 에만 overfitting 되는 것을 방지함

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

- Sampling을 거친 후의 Volume Rendering

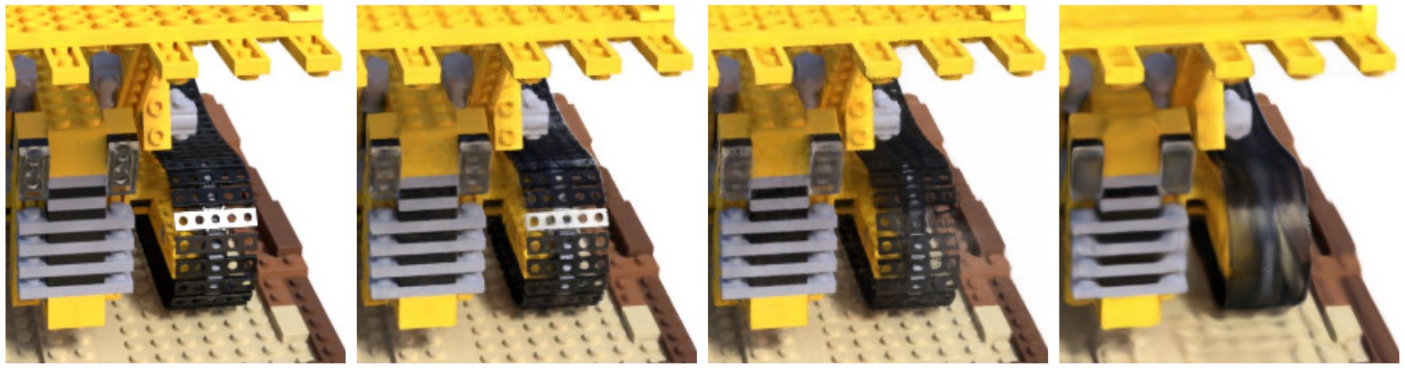
$$\hat{C}(\mathrm{r}) = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i \delta_i))c_i, where\ T_i = \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j)$$

Positional encoding

NeRF에서는 입력데이터 $P = (x, y, z, \theta, \emptyset)$를 고주파 함수로 인코딩함
*카메라 위치 $(x, y, z) \rightarrow L = 60$, 바라보는 방향 $\{(\theta, \phi) \rightarrow (X, Y, Z)\} \rightarrow L = 4$

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

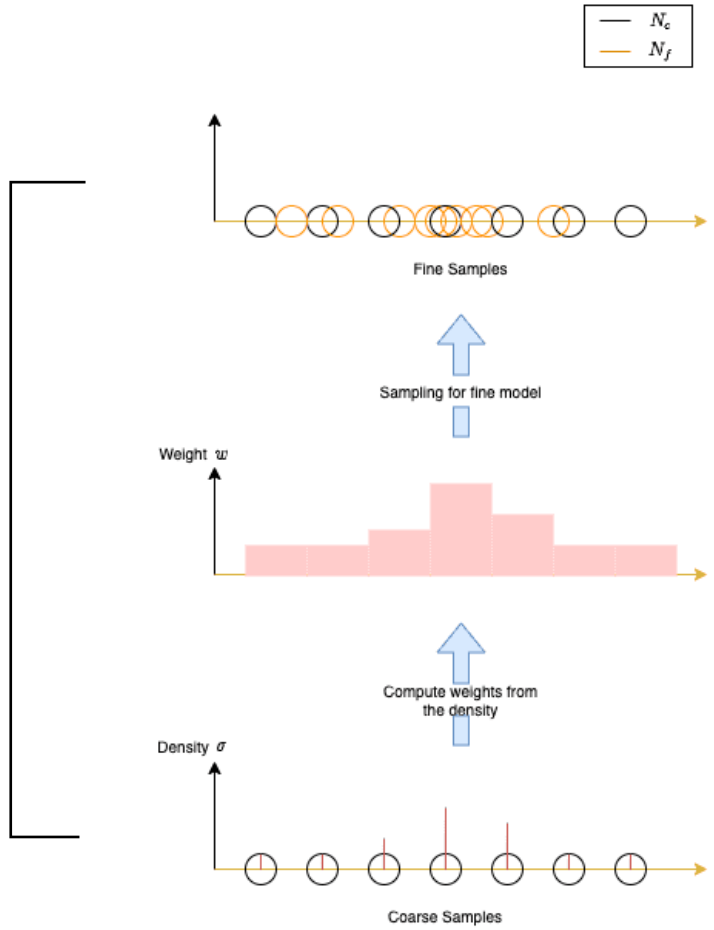
- Fourier Feature로 Low Frequency, High Frequency 도메인을 고려함
- NeRF의 Layer는 weighted sum이기 때문에 비슷한 포인트 입력이 된다면 MLP 출력 결과는 매우 비슷하게 나오게 됨
- 고주파 특성을 반영하여 정교한 고화질의 이미지가 생성된다.



Ground Truth | Complete Model | No View Dependence | No Positional Encoding

Hierarchical volume sampling

계층적 구조



Fine Network

$$C(r) = \int_{t_n}^{t_f} T(t)\,\sigma(r(t))c(r(t), d)dt$$

Predicted

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

Predicted

Data ground truth

Coarse Network: density와 Transmittance의 PDF로 부터 ray 내 분포로부터 sampling

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

$\delta_i = t_{i+1} - t_i$ 사이의 Density σ를 PDF(확률 밀도 함수)로 변환

SK telecom
SKT FLY AI Challenger

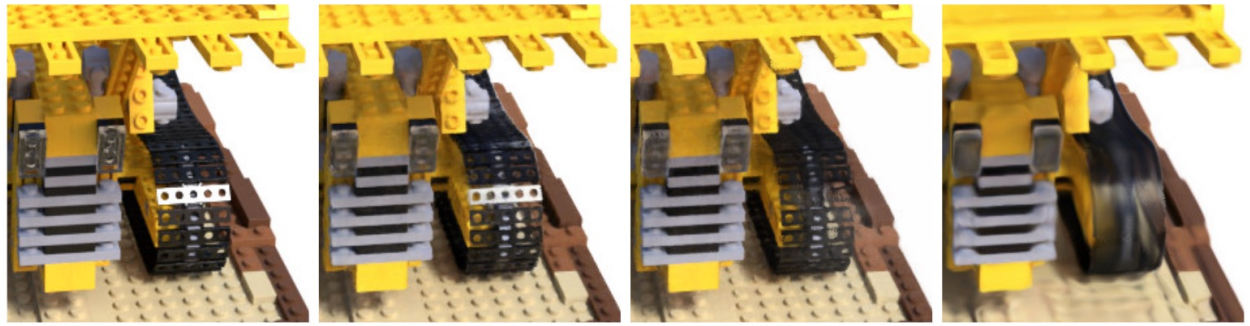
Result

Result

변형된 영상 화질 손실량 평가

	Input	#Im.	L	(N_c, N_f)	PSNR↑	SSIM↑	LPIPS↓
1) No PE, VD, H	xyz	100	-	(256, -)	26.67	0.906	0.136
2) No Pos. Encoding	$xyz\theta\phi$	100	-	(64, 128)	28.77	0.924	0.108
3) No View Dependence	xyz	100	10	(64, 128)	27.66	0.925	0.117
4) No Hierarchical	$xyz\theta\phi$	100	10	(256, -)	30.06	0.938	0.109
5) Far Fewer Images	$xyz\theta\phi$	25	10	(64, 128)	27.78	0.925	0.107
6) Fewer Images	$xyz\theta\phi$	50	10	(64, 128)	29.79	0.940	0.096
7) Fewer Frequencies	$xyz\theta\phi$	100	5	(64, 128)	30.59	0.944	0.088
8) More Frequencies	$xyz\theta\phi$	100	15	(64, 128)	30.81	0.946	0.096
9) Complete Model	$xyz\theta\phi$	100	10	(64, 128)	**31.01**	**0.947**	**0.081**

영상 화질 손실 평가 이미지 유사도 평가



Ground Truth Complete Model No View Dependence No Positional Encoding

Limitation of NeRF

- Training, rendering 속도가 매우 느림
  → Instant NGP (SIGGRAPH 2022), PointNeRF(CVPR 2022)
- 입력 이미지수가 50장 이상으로 매우 많음 → PixelNeRF (CVPR 2021)
- 입력 Camera Parameter를 축소 → BARF (ICCV 2021)
- Rendering Quality와 해상도 → PointNeRF (CVPR 2022)
- Static scene에 대해서만 Rendering 가능 → D-NeRF (CVPR 2021)
- Relighting & Scene Editing 불가능 → NeRD (ICCV 2021)
- Generalization 불가능 → PixelNeRF (CVPR 2021)
- Single-Scale (Multi-Scale 불가) → MipNeRF (CVPR2021)

Thank you

SK telecom
SKT FLY AI Challenger