# Pose Free 3D Gaussian Splatting

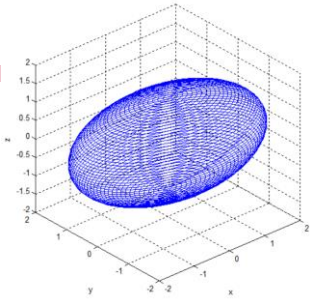
2024 summer seminar

***Sogang University***
*Vision & Display Systems Lab, Dept. of Electronic Engineering*

***Presented By***
*염수웅*

# Outline

- Background
  - 3D Gaussian Splatting
  - Point Cloud Reconstruction using monocular depth
- Paper1
  - COLMAP-Free 3D Gaussian Splatting – [CVPR 2024]
- Paper2
  - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose - [SIGGRAPH 2024]



# Background

- 3D Gaussian Splatting-[SIGGRAPH 23]
  - 입력 이미지, 카메라 파라미터, 3D point cloud 를 입력으로 받아 모델 최적화
    - SfM 기반의 방법론으로 추정된 point cloud 를 mean 으로 하는 3D Gaussian 모델링

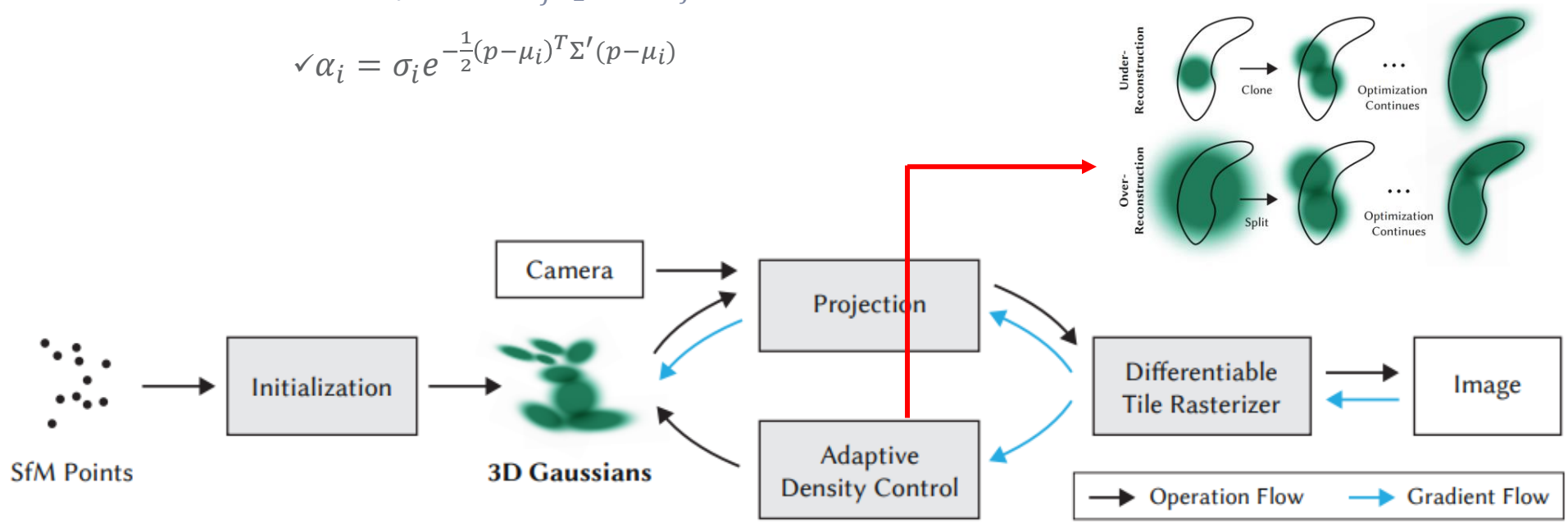
$$G(p) = e^{-\frac{1}{2}(p-\mu)^T \Sigma (p-\mu)}$$

$$\Sigma = RSS^T R^T$$

$$\Sigma' = JW\Sigma W^T J^T$$

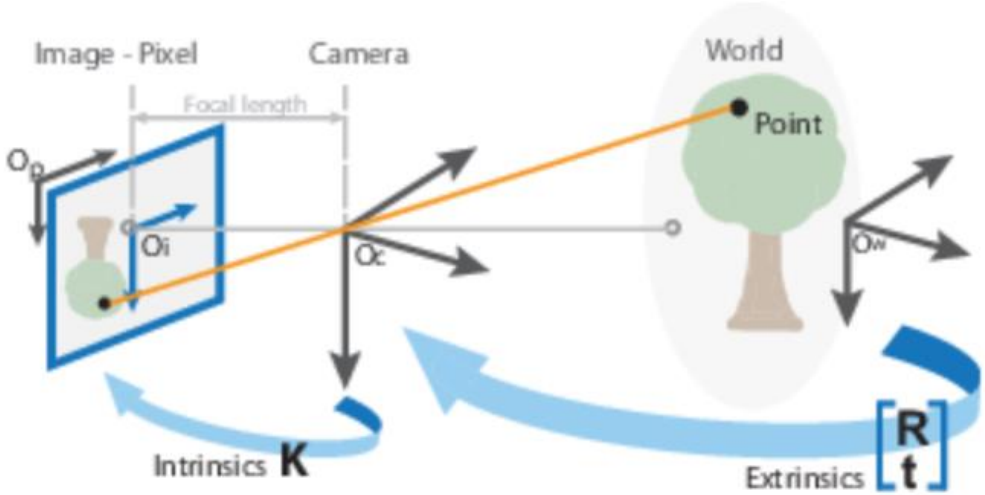
$$C(p) = \sum_{i \in N} c_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j)$$

$$\checkmark \alpha_i = \sigma_i e^{-\frac{1}{2}(p-\mu_i)^T \Sigma' (p-\mu_i)}$$



# Background

- Point Cloud Reconstruction using monocular depth
  - Monocular depth 를 이용한 point cloud 계산
    - 이미지 좌표에 intrinsic 의 inverse 를 곱해서 카메라 좌표계로 변환
    - Homogeneous coordinate 로 표현된 카메라 좌표계에 depth map 을 곱해서 3d point 획득
    - 3d point 는 카메라 좌표계 기준이기 때문에 C2W transformation 을 통해 point cloud 획득
  - 마찬가지 방법으로 ray=o+td 또한 계산 가능
    - o: camera translation (world 좌표계)
    - d: direction vector (해당 픽셀에 intrinsic inverse 곱하고 C2W 변환 수행한 값)

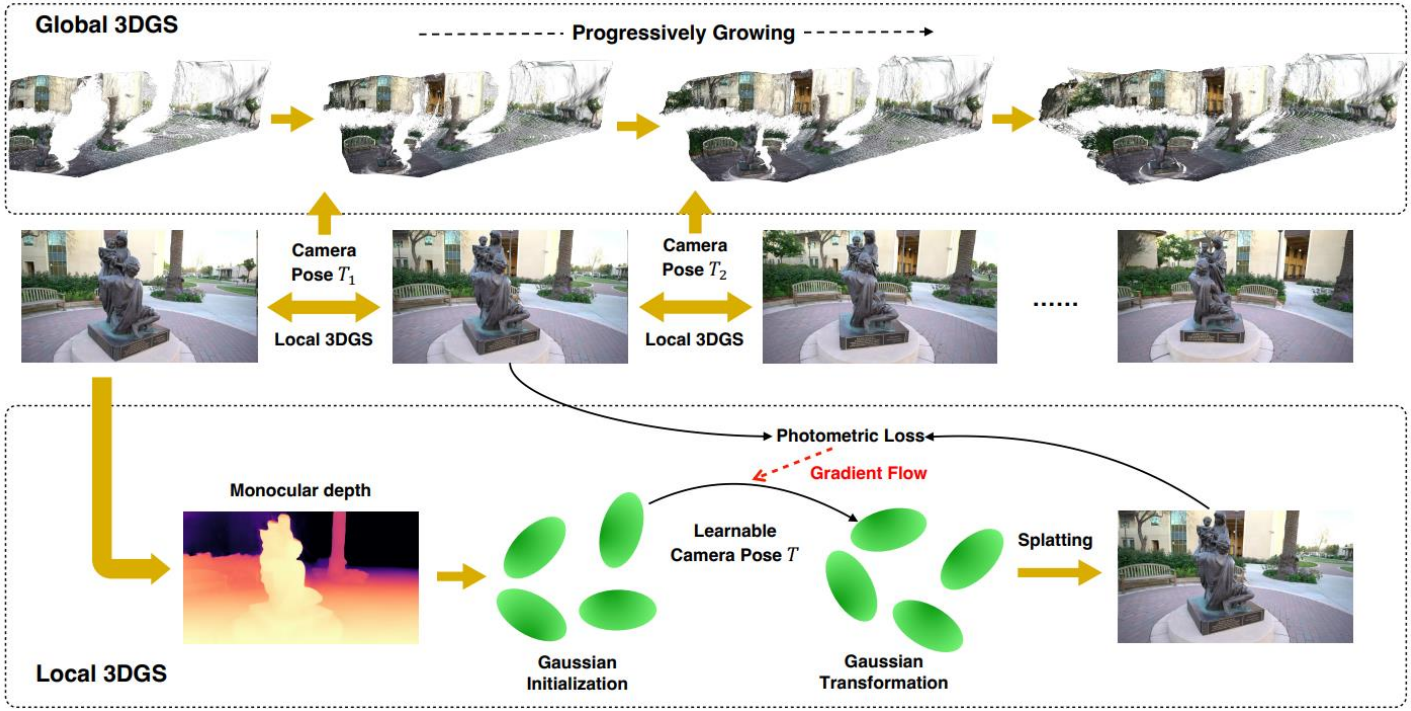


# Paper1 - COLMAP-Free 3D Gaussian Splatting

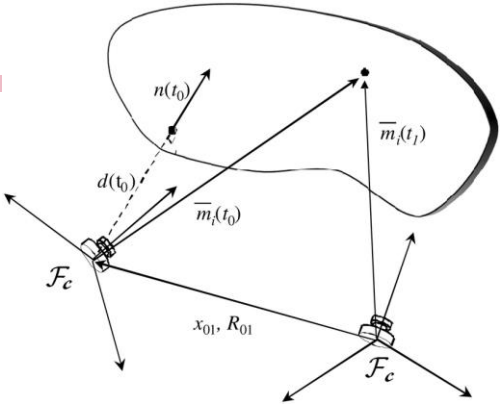
- Introduction
  - 사전에 정확히 계산된 카메라 포즈를 필요로하는 3D 복원 및 렌더링 분야
    - NeRF 나 Gaussian Splatting 모델과 같이 3차원 장면을 렌더링 하는 분야는 사전에 정확하게 계산된 카메라 포즈가 있어야만 동작이 가능
  - 카메라 포즈 없이도 렌더링이 수행될 수 있는 연구의 등장
    - NeRF 기반 방법론 연구가 활발하게 이루어지던 작년까지 카메라 포즈를 추정하는 동시에 3D 렌더링을 수행하는 방법이 꾸준하게 발표되어짐
      - BARF : GT 카메라와 유사한 초기 camera pose 가 필요
      - NeRFmm : forward-facing scene 에 한정적으로 수행 가능
      - Nope-NeRF : 학습시간이 매우 길고(약 30시간) rotation 이 크게 변하면 어려움
    - 하지만 implicit 표현에 의존하는 NeRF 의 특성상 3D 구조와 카메라 포즈를 동시에 최적화 하는 것이 어려움
      - 명시적으로 표현되는 3차원 기하학적 구조가 수학적으로 정의되어야 카메라 포즈를 최적화하기에 수월함
      - NeRF는 MLP 학습에 의존하며 카메라 포즈를 직접적으로 추정하는 것이 아니라 ray tracing 을 업데이트하는 간접적인 방법을 통해 추정

# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Main Method
  - Overview
    - Explicit 한 표현을 사용하는 3D Gaussian Splatting 방법을 활용하여 카메라 포즈 추정
      - 명시적인 기하학적 표현과 연속적인 비디오 프레임을 활용
        - ✓입력 프레임을 연속적으로 처리하며 한번에 하나의 프레임을 가지고옴으로써 3D Gaussian 을 점진적으로 성장시켜나감



# Paper1 - COLMAP-Free 3D Gaussian Splatting




- Main Method
  - Local 3DGS for Relative Pose Estimation
    - 한 쌍의 영상을 입력 받아 상대적인 카메라 포즈를 추정하는 단계
      - Timestep $t$ 의 프레임 $I_t$ 와 timestep $t+1$의 프레임 $I_{t+1}$ 간의 카메라 변환 행렬 추정
    - 3D Gaussian 의 강체 변환과 카메라 포즈 사이의 관계에 착안하여 Local 3DGS 방법 설계
      - 3D Gaussian 의 mean($\mu$) 값인 point cloud 의 위치 벡터는 pixel plane 의 위치($\mu_{2D}$)로 다음과 같이 projection 되어 짐

        ✓ $\mu_{2D} = \frac{K(W\mu)}{(W\mu)_z}$ → Monocular depth 를 이용한 3D reconstruction 의 역연산

        - $W$: world to camera transformation (Camera Pose)
      - 해당 변환은 $\mu$ 가 $W$ 에 의해 강체 변환 후 projection 되는 것과 동일하게 해석 가능

        ✓ $\mu' = W\mu$

        ✓ $\mu_{2D} = \frac{K\mu'}{(K\mu')_z}$
      - 카메라 포즈와 3D Gaussian 강체 변환의 관계를 활용하여 Local 3DGS 설계

# Paper1 - COLMAP-Free 3D Gaussian Splatting

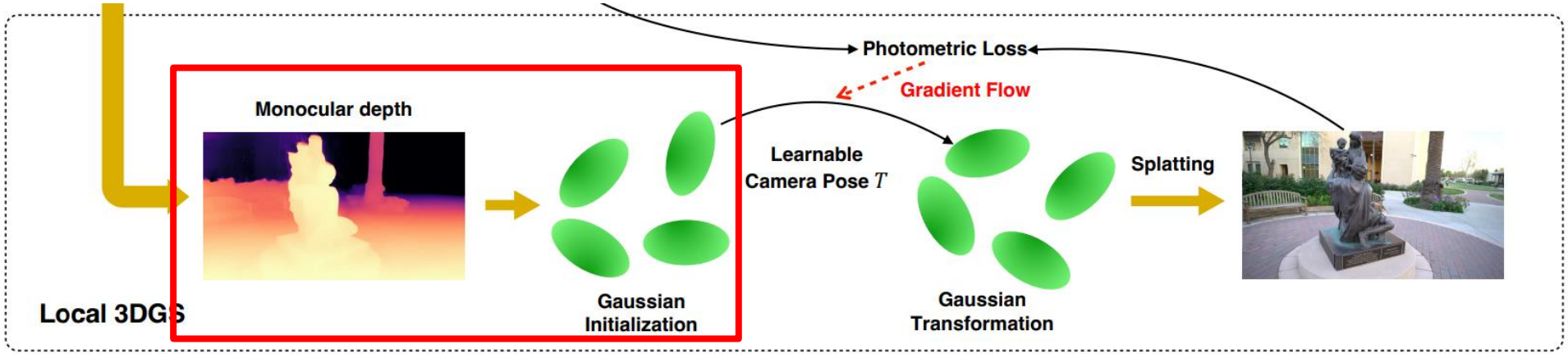
- Main Method
  - Local 3DGS for Relative Pose Estimation
    - Local 3DGS process
      - 3D Gaussian initialization from a single view

✓사전 학습된 단안 깊이 추정 모델 이용하여 기준 이미지 $I_t$ 의 깊이맵 추정

✓Intrinsic parameter 와 깊이맵 이용하여 point cloud reconstruction 수행

✓해당 point cloud 를 gaussian 의 초기값으로 하는 3D Gaussian $G_t$ 학습

- $G_t^* = arg \min_{c_t, r_t, s_t, \alpha_t} \mathcal{L}_{rgb}(\mathcal{R}(G_t), I_t)$
  - $\mathcal{R}$: 3DGS rendering process, $\mathcal{L}_{rgb}$: 3DGS training loss function



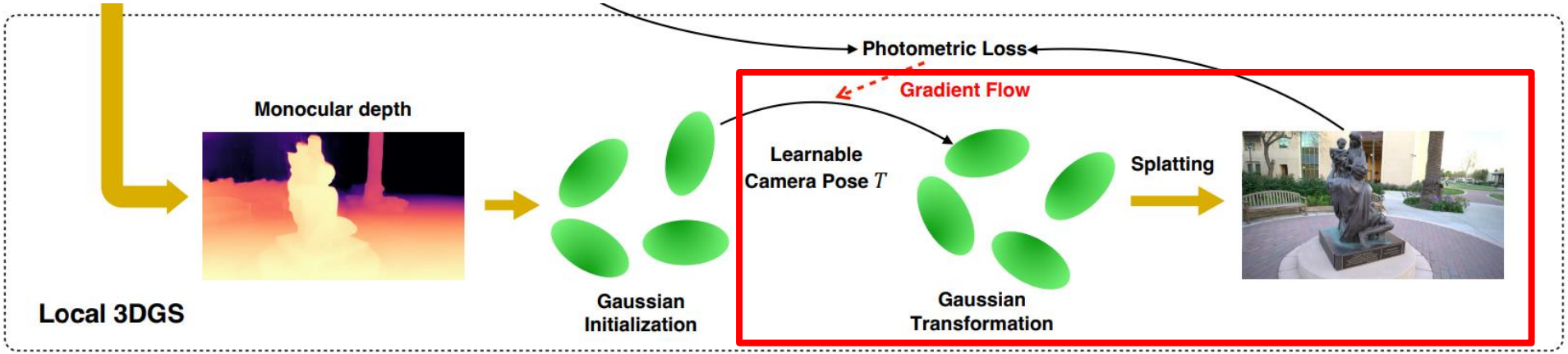
# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Main Method
  - Local 3DGS for Relative Pose Estimation
    - Local 3DGS process
      - Pose Estimation by 3D Gaussian Transformation

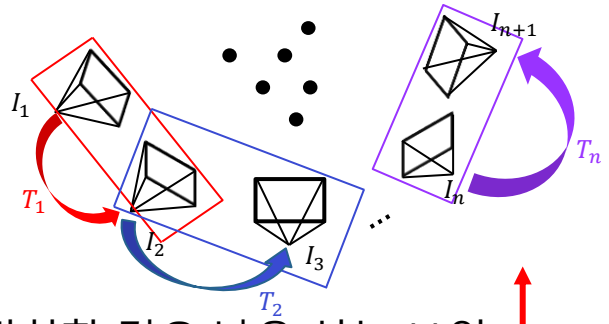
3x3 rotation matrix
&
3x1 translation vector
두 구성요소로 이루어진 3차원 공간상의
강체 변환 행렬

✓학습된 3D Gaussian $G_t^*$ 을 다음 프레임 $I_{t+1}$ 의 image space 로 강체 변환 수행
- 강체 변환은 learnable SE-3 affine transformation $T_t$ 를 통해 수행
- $G_{t+1} = T_t \odot G_t$

✓변환행렬은 렌더링된 이미지의 photometric loss 를 최적화함으로써 학습되어짐
- $T_t^* = arg \min_{T_t} \mathcal{L}_{rgb}(\mathcal{R}(T_t \odot G_t), I_{t+1})$



# Paper1 - COLMAP-Free 3D Gaussian Splatting

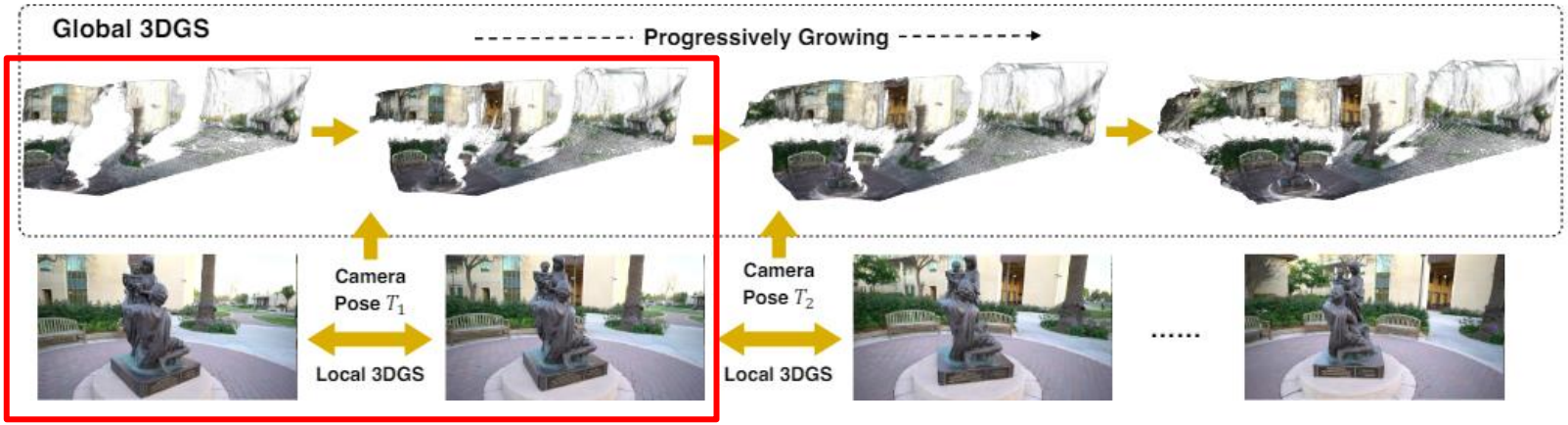



- Main Method
  - Global 3DGS with Progressively Growing
    - Local 3DGS 만 이용하여 추정한 카메라 포즈로 3DGS 최적화할 경우 낮은 성능 보임
      - Local 3DGS 를 이용한 전체 카메라 포즈 추정 방법
        - ✓연속적인 프레임들에 대해 한 쌍 단위로 상대 포즈들 추정 후 전체 align 조정
          - $I_1$을 기준 카메라 포즈(identity matrix) 로 하여 전체 카메라 포즈 추정 가능
            - $W_1 = Identity, W_2 = T_1, W_3 = T_1 \times T_2, \ldots, W_{n+1} = T_1 \times T_2 \times \cdots \times T_n$
      - 각 쌍에서 상대적 카메라 포즈가 독립적으로 추정
        - ✓Monocular depth 에 의한 reconstruction 이 문제
          - 전체 장면에 대한 consistency 가 없기 때문
        - ✓인접한 각 쌍의 정보만이 활용되어짐
          - 전체 카메라에 대한 정보를 참조하지 않음
        - ✓렌더링 성능 및 포즈 추정 성능을 통해 확인 가능

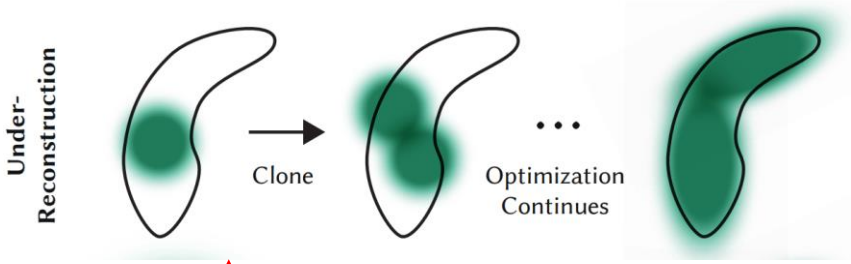
| scenes | w.o. growing | | | |
|---|---|---|---|---|
| | PSNR | SSIM | $RPE_t$ | $RPE_r$ |
| Church | 22.01 | 0.72 | 0.044 | 0.122 |
| Barn | 25.20 | 0.85 | 0.152 | 0.232 |
| Museum | 20.95 | 0.70 | 0.079 | 0.212 |
| Family | 22.30 | 0.77 | 0.065 | 0.028 |
| Horse | 23.47 | 0.81 | 0.147 | 0.066 |
| Ballroom | 23.36 | 0.79 | 0.056 | 0.073 |
| Francis | 22.20 | 0.69 | 0.147 | 0.161 |
| Ignatius | 21.05 | 0.67 | 0.24 | 0.058 |
| mean | 22.57 | 0.75 | 0.116 | 0.119 |

# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Main Method
  - Global 3DGS with Progressively Growing
    - Local 3DGS 학습함과 동시에 전체 카메라 구조를 활용하는 progressively growing 적용
    - Progreesive Growing process
      - $I_t$와 $I_{t+1}$에 대해 Local 3DGS 수행하여 상대적 포즈와 initial 3D Gaussian 획득
      - 추정된 포즈와 initial 3D Gaussian 을 이용하여 3D Gaussian 모델 학습
        - ✓$I_t$ 를 기준 카메라로 하며 $I_t$와 $I_{t+1}$ 을 super vision 하여 3D Gaussian 업데이트



# Paper1 - COLMAP-Free 3D Gaussian Splatting




- Main Method
  - Global 3DGS with Progressively Growing
    - Progreesive Growing process
      - 전체 프레임에 대해 해당 과정을 점진적으로 수행
        - ✓새로운 프레임을 입력받는 순간마다 densification 수행
          - 이전 단계까지 학습된 3D Gaussian 은 새로운 프레임의 모든 구간을 커버할 수 없음
          - 3D 장면에서 빈 공간을 채우는 under reconstruction 수행
    - 초기 point cloud 에서 시작해 전체 장면을 커버하는 완성된 point cloud 로 점진적 성장


| scenes | w.o. growing | | | | Ours | | | |
|---|---|---|---|---|---|---|---|---|
| | PSNR | SSIM | $RPE_t$ | $RPE_r$ | PSNR | SSIM | $RPE_t$ | $RPE_r$ |
| Church | 22.01 | 0.72 | 0.044 | 0.122 | 30.23 | 0.93 | 0.008 | 0.018 |
| Barn | 25.20 | 0.85 | 0.152 | 0.232 | 31.23 | 0.90 | 0.034 | 0.034 |
| Museum | 20.95 | 0.70 | 0.079 | 0.212 | 29.91 | 0.91 | 0.052 | 0.215 |
| Family | 22.30 | 0.77 | 0.065 | 0.028 | 31.27 | 0.94 | 0.022 | 0.024 |
| Horse | 23.47 | 0.81 | 0.147 | 0.066 | 33.94 | 0.96 | 0.112 | 0.057 |
| Ballroom | 23.36 | 0.79 | 0.056 | 0.073 | 32.47 | 0.96 | 0.037 | 0.024 |
| Francis | 22.20 | 0.69 | 0.147 | 0.161 | 32.72 | 0.91 | 0.029 | 0.154 |
| Ignatius | 21.05 | 0.67 | 0.24 | 0.058 | 28.43 | 0.90 | 0.033 | 0.032 |
| mean | 22.57 | 0.75 | 0.116 | 0.119 | **31.28** | **0.93** | **0.041** | **0.069** |

# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Result
  - 정량 결과

| scenes | Ours | | | Nope-NeRF | | | BARF | | | NeRFmm | | | SC-NeRF | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $RPE_t \downarrow$ | $RPE_r \downarrow$ | ATE↓ | $RPE_t$ | $RPE_r$ | ATE | $RPE_t$ | $RPE_r$ | ATE | $RPE_t$ | $RPE_r$ | ATE | $RPE_t$ | $RPE_r$ | ATE |
| Church | 0.008 | 0.018 | 0.002 | 0.034 | 0.008 | 0.008 | 0.114 | 0.038 | 0.052 | 0.626 | 0.127 | 0.065 | 0.836 | 0.187 | 0.108 |
| Barn | 0.034 | 0.034 | 0.003 | 0.046 | 0.032 | 0.004 | 0.314 | 0.265 | 0.050 | 1.629 | 0.494 | 0.159 | 1.317 | 0.429 | 0.157 |
| Museum | 0.052 | 0.215 | 0.005 | 0.207 | 0.202 | 0.020 | 3.442 | 1.128 | 0.263 | 4.134 | 1.051 | 0.346 | 8.339 | 1.491 | 0.316 |
| Family | 0.022 | 0.024 | 0.002 | 0.047 | 0.015 | 0.001 | 1.371 | 0.591 | 0.115 | 2.743 | 0.537 | 0.120 | 1.171 | 0.499 | 0.142 |
| Horse | 0.112 | 0.057 | 0.003 | 0.179 | 0.017 | 0.003 | 1.333 | 0.394 | 0.014 | 1.349 | 0.434 | 0.018 | 1.366 | 0.438 | 0.019 |
| Ballroom | 0.037 | 0.024 | 0.003 | 0.041 | 0.018 | 0.002 | 0.531 | 0.228 | 0.018 | 0.449 | 0.177 | 0.031 | 0.328 | 0.146 | 0.012 |
| Francis | 0.029 | 0.154 | 0.006 | 0.057 | 0.009 | 0.005 | 1.321 | 0.558 | 0.082 | 1.647 | 0.618 | 0.207 | 1.233 | 0.483 | 0.192 |
| Ignatius | 0.033 | 0.032 | 0.005 | 0.026 | 0.005 | 0.002 | 0.736 | 0.324 | 0.029 | 1.302 | 0.379 | 0.041 | 0.533 | 0.240 | 0.085 |
| mean | **0.041** | 0.069 | **0.004** | 0.080 | **0.038** | 0.006 | 1.046 | 0.441 | 0.078 | 1.735 | 0.477 | 0.123 | 1.890 | 0.489 | 0.129 |

- $RPE_t$
  - 위치 변화에 초점을 맞춘 evaluation metric 으로 두 포즈 사이의 거리 변화의 오차
    - ✓ $RPE_t = \|(T - \hat{T}) \cdot p\|$
- $RPE_r$
  - 회전 오차에 초점을 맞춘 evaluation metric 으로 두 포즈 사이 회전 각도 변화의 오차
    - ✓ $RPE_r = angle(R \cdot \hat{R}^{-1})$
- $ATE$
  - 실제 포즈와 추정 포즈 사이 유클리드 거리와 회전 오차를 계산한 evaluation metric
    - ✓ $ATE = \sqrt{\frac{1}{N}\sum_i^N \|p_i - \hat{p_i}\|^2}$

# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Result
  - 정량 결과

| scenes | Ours | | | Nope-NeRF | | | BARF | | | NeRFmm | | | SC-NeRF | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | LPIPS ↓ | PSNR | SSIM | LPIPS | PSNR | SSIM | LPIPS | PSNR | SSIM | LPIPS | PSNR | SSIM | LPIPS |
| Church | **30.23** | **0.93** | **0.11** | 25.17 | 0.73 | 0.39 | 23.17 | 0.62 | 0.52 | 21.64 | 0.58 | 0.54 | 21.96 | 0.60 | 0.53 |
| Barn | **31.23** | **0.90** | **0.10** | 26.35 | 0.69 | 0.44 | 25.28 | 0.64 | 0.48 | 23.21 | 0.61 | 0.53 | 23.26 | 0.62 | 0.51 |
| Museum | **29.91** | **0.91** | **0.11** | 26.77 | 0.76 | 0.35 | 23.58 | 0.61 | 0.55 | 22.37 | 0.61 | 0.53 | 24.94 | 0.69 | 0.45 |
| Family | **31.27** | **0.94** | **0.07** | 26.01 | 0.74 | 0.41 | 23.04 | 0.61 | 0.56 | 23.04 | 0.58 | 0.56 | 22.60 | 0.63 | 0.51 |
| Horse | **33.94** | **0.96** | **0.05** | 27.64 | 0.84 | 0.26 | 24.09 | 0.72 | 0.41 | 23.12 | 0.70 | 0.43 | 25.23 | 0.76 | 0.37 |
| Ballroom | **32.47** | **0.96** | **0.07** | 25.33 | 0.72 | 0.38 | 20.66 | 0.50 | 0.60 | 20.03 | 0.48 | 0.57 | 22.64 | 0.61 | 0.48 |
| Francis | **32.72** | **0.91** | **0.14** | 29.48 | 0.80 | 0.38 | 25.85 | 0.69 | 0.57 | 25.40 | 00.69 | 0.52 | 26.46 | 0.73 | 0.49 |
| Ignatius | **28.43** | **0.90** | **0.09** | 23.96 | 0.61 | 0.47 | 21.78 | 0.47 | 0.60 | 21.16 | 0.45 | 0.60 | 23.00 | 0.55 | 0.53 |
| mean | **31.28** | **0.93** | **0.09** | 26.34 | 0.74 | 0.39 | 23.42 | 0.61 | 0.54 | 22.50 | 0.59 | 0.54 | 23.76 | 0.65 | 0.48 |

- PSNR
  - GT 영상과 픽셀 값의 차이를 정량적으로 나타내는 수치
- SSIM
  - Gaussian kernel 을 이용하여 두 이미지의 구조적 유사성을 측정하는 수치
- LPIPS
  - Pretrained CNN 모델을 활용하여 추출한 영상의 feature 들 간의 유사도를 나타내는 수치

# Paper1 - COLMAP-Free 3D Gaussian Splatting

- Result
  - 정성 결과

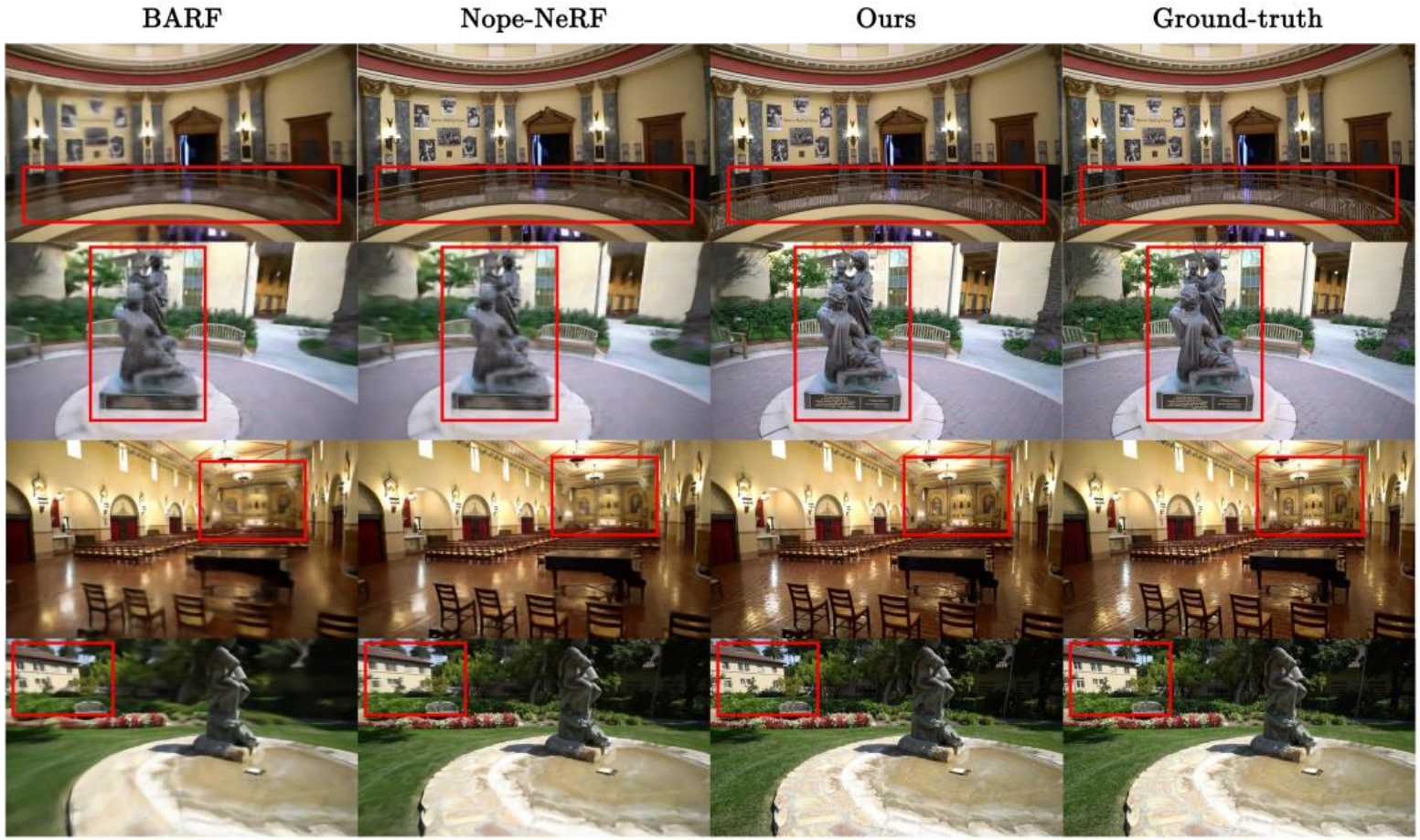



Figure 3. **Qualitative comparison for novel view synthesis on Tanks and Temples.** Our approach produces more realistic rendering results than other baselines. **Better viewed when zoomed in.**

# Paper1 - COLMAP-Free 3D Gaussian Splatting

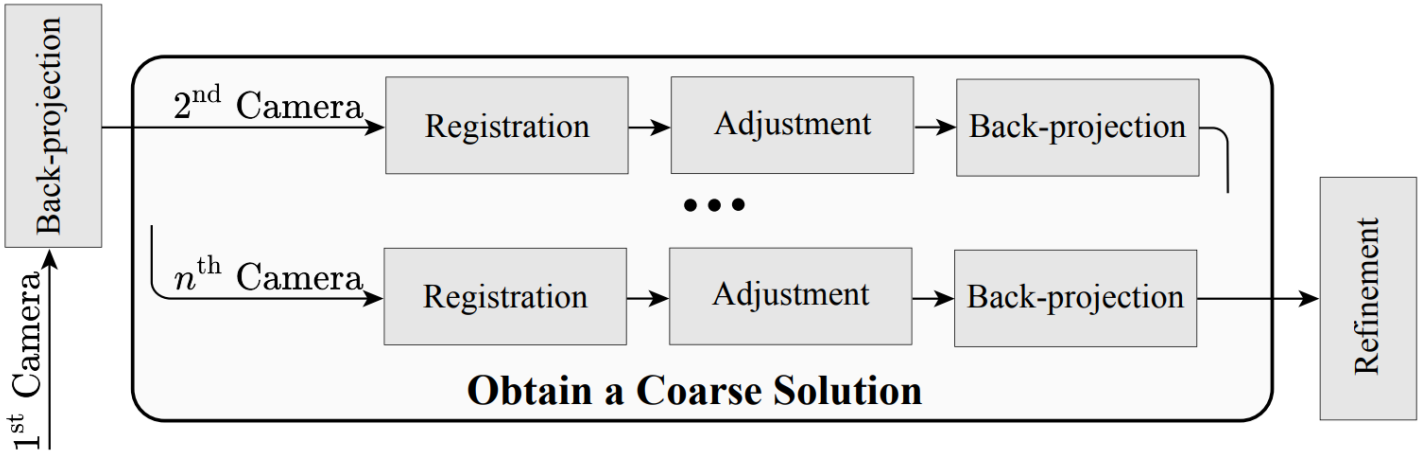
- Result
  - 정성 결과

# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Introduction
  - Sparse view 환경에선 카메라 포즈 추정이 어려움
    - 일반적으로 static scene 에 대한 카메라 포즈는 SfM 기반 방법론으로 추정되어짐
      - Feature matching, triangulation 을 활용하여 reprojection error 최소화하는 최적화 방법
    - Reprojection error 를 최적화 하기위한 3차원 정보가 충분히 만들어 지지 않기 때문
  - Sparse view 환경에선 novel view synthesis 또한 어려움
    - NeRF 와 3D Gaussian Splatting 과 같이 높은 성능을 달성하는 방법론들은 사전에 정확하게 계산되어진 카메라 포즈를 필요로 함
      - 3D Gaussian Splatting 방법은 카메라 포즈 뿐만 아니라 초기 point cloud 도 매우 중요
    - Sparse view 환경은 SfM 기반 방법론으로 카메라 포즈 추정이 어렵기 때문에 novel view synthesis 또한 어려움
    - FoV 가 크게 겹쳐서 카메라 포즈 추정이 되었다고 하더라도 3차원 정보 학습을 위한 픽셀 정보가 적기 때문에 novel view synthesis 는 여전히 어려움
  - Sparse view 환경에서 novel view synthesis 가 잘 되어질 수 있도록 논문의 최적화 방법 및 카메라 포즈 와 depth 를 같이 조정하는 방법론 제시

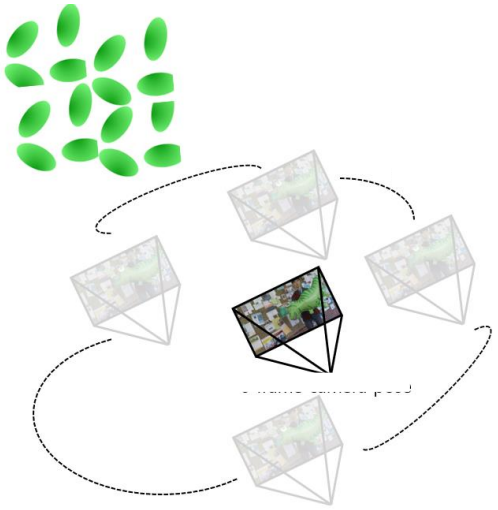
# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

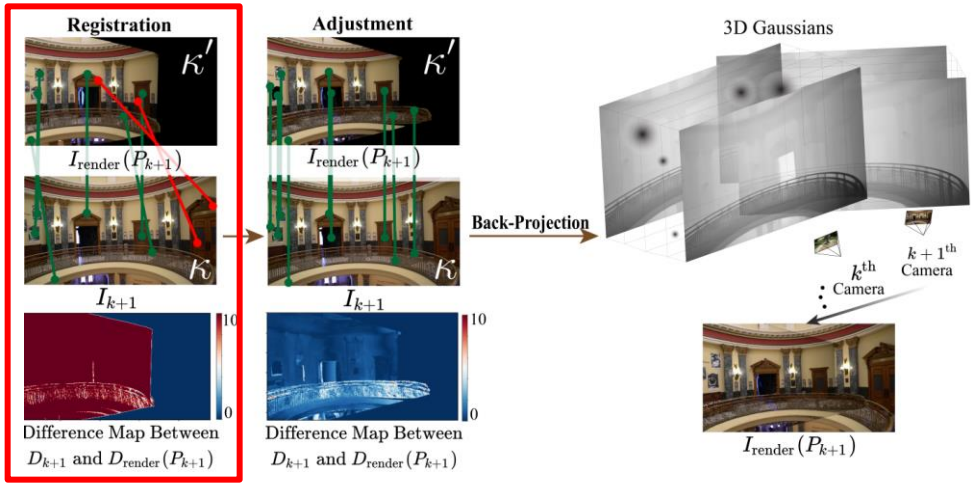
- Main Method
  - Algorithm Overview
    - 첫 번째로 입력 받은 영상 $I_1$에 대한 카메라 포즈($P_1$)를 identity 로 등록
    - $I_1$ 과 대응되는 깊이맵 $D_1$을 이용하여 back-projection (point cloud reconstruction) 수행
    - Back-projection 되어진 point cloud 를 이용하여 3D Gaussian 학습
    - 두 번째로 $I_2$와 대응되는 깊이맵 $D_2$를 입력 받음
    - $I_2$에 대한 카메라 포즈($P_2$)를 $I_1$ 과 동일한 값으로 복사
    - $I_1$으로 학습된 3D Gaussian 을 $P_2$로 렌더링하며 포즈 등록 수행
    - 해당 과정을 계속 반복하면서 전체 카메라 포즈를 추정



# Paper2- A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

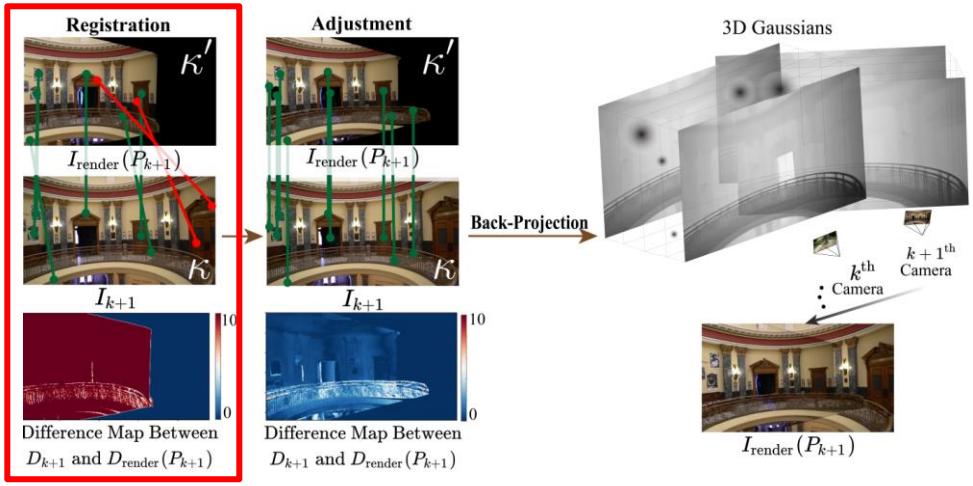
- Main Method
  - Optimization Framework
    - Registration stage
      - $k$번째 영상까지 카메라 포즈 및 3D Gaussian 에 대한 등록이 되었다고 가정
      - $k$번째 영상 카메라 포즈를 복사하여 $k+1$번 째 영상 카메라 포즈 초기값으로 등록
      - 3D Gaussian 을 $k+1$번 째 카메라로 렌더링
        - ✓렌더링 된 영상과 GT 영상 사이의 photometric loss 를 통해 카메라 포즈 최적화
          - $\mathcal{L}_{rgb} = \|I - I_{render}(P)\|_1$





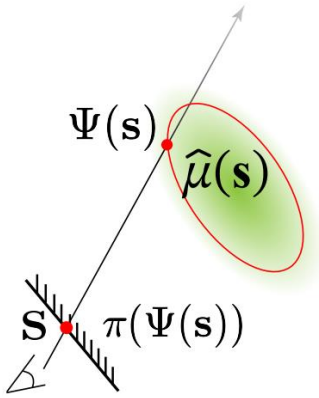
# Paper2- A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Main Method
  - Optimization Framework
    - Registration stage
      - 렌더링된 영상과 GT 영상 사이의 correspondence 획득
        - ✓Correspondence 픽셀 위치가 서로 동일하도록 매칭하여 카메라 포즈를 최적화
          - $\mathcal{L}_{corr} = \sum_{i=1}^{M} \left\| q\left(\kappa'^{(i)}\right) - \kappa^{(i)} \right\|_1$
      - 렌더링된 영상과 GT 영상 사이의 photometric loss 와 correspondence loss 를 통해 복사된 값을 초기값으로하는 포즈가 최적화 되도록 유도



# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

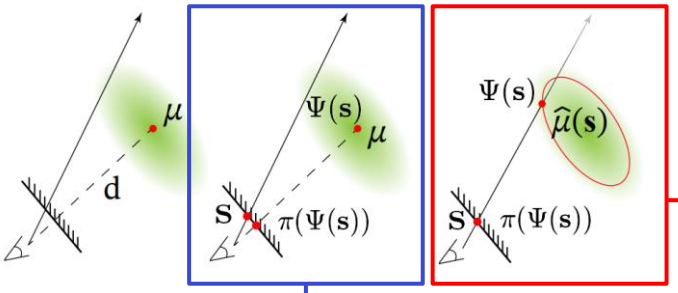
- Main Method
  - Optimization Framework
    - Differentiable Surface Rendering
      - GT 영상의 특징점과 대응되는 $k+1$ 카메라에 대한 렌더링 영상의 특징점 획득
      - 획득된 특징점과 대응되는 렌더링 영상의 픽셀 정보 저장
      - 저장된 픽셀 정보와 대응되는 3D Gaussian 에 대한 surface reconstruction 수행
        - ✓해당 픽셀에 대응되는 surface 를 3D Gaussian 정보들을 이용하여 reconstruction
          - $\Psi(s) = \sum_{i,\widehat{\mu_i}(s)\neq\emptyset} \widehat{\mu_i}(s)\alpha_i(s)\prod_{j=1,\widehat{\mu_j}(s)}^{i-1}(1-\alpha_j(s))$
          - $\widehat{\mu_i}(s)$ 는 픽셀 $s$에 해당하는 ray 와 3D surface 가 교차되는 점
            - $\widehat{\mu_i}(s) = o + l\mathrm{d}(s)$
          - $\widehat{\mu_i}(\mathrm{x}) = \mu_i + R_i f(s_i)$ ⟶ Ellipsoid shell coordinate 에서 정의
            - $\mu_i$: Gaussian center, $R_i$: Gaussian rotation, $s_i$: Gaussian scaling
        - ✓Reconstruction 된 surface 를 렌더링하여 $q(\kappa'^{(i)})$ 획득
          - $q(s) = \sum_{i,\widehat{\mu_i}(s)\neq\emptyset} \pi(\widehat{\mu_i}(s))\alpha_i(s)\prod_{j=1,\widehat{\mu_j}(s)}^{i-1}(1-\alpha_j(s))$




      - 카메라 포즈에 해당하는 ray 매개변수와 3D Gaussian 매개변수가 함께 최적화 됨

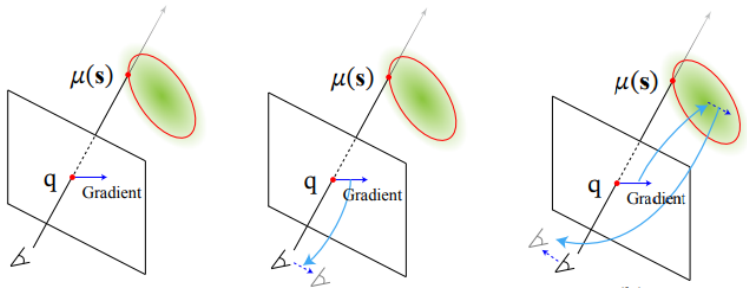
# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Main Method
  - Optimization Framework
    - Differentiable Surface Rendering
      - Surface Rendering




$$\Psi(\mathbf{s}) = \sum_i \boldsymbol{\mu}_i \alpha_i(\mathbf{s}) \prod_{j=1}^{i-1} (1 - \alpha_j(\mathbf{s}))$$

$$\Psi(\mathbf{s}) = \sum_{i, \widehat{\boldsymbol{\mu}}_i(\mathbf{s}) \neq \varnothing} \widehat{\boldsymbol{\mu}}_i(\mathbf{s}) \alpha_i(\mathbf{s}) \prod_{j=1, \widehat{\boldsymbol{\mu}}_j(\mathbf{s}) \neq \varnothing}^{i-1} (1 - \alpha_j(\mathbf{s}))$$



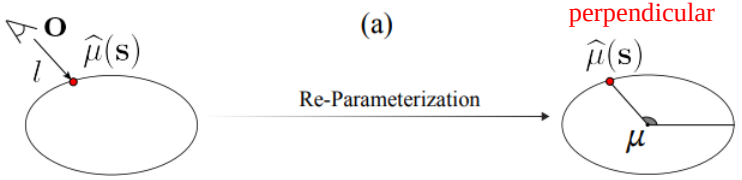

Surface rendering gradients propagtion

$$\frac{\partial \mathcal{L}}{\partial \mu_i(\mathbf{x})}$$

$$\mu_i(\mathbf{x}) = o + ld$$

$$\frac{\partial \mathcal{L}}{\partial l} = \frac{\partial \mathcal{L}}{\partial \mu_i(\mathbf{x})} \cdot \frac{\partial \mu_i(\mathbf{x})}{\partial l} = \frac{\partial \mathcal{L}}{\partial \mu_i(\mathbf{x})} \cdot \mathbf{d}$$

perpendicular




**Algorithm 2** Numerically Stable Ray-ellipsoid Intersection Test

**Input:** Ray origin $\mathbf{o} \in \mathbb{R}^3$ and ray direction $\mathbf{d} \in \mathbb{R}^3$. The properties of an ellipsoid: center $\mathbf{e}_c \in \mathbb{R}^3$, scaling $\mathbf{e}_s = (\mathbf{e}_{s_x}, \mathbf{e}_{s_y}, \mathbf{e}_{s_z})^T \in \mathbb{R}^3_+$ and rotation $\mathbf{e}_r \in \mathfrak{so}(3)$.

**Output:** Whether the ray intersect the ellipsoid. If yes, return the world space coordinates of the intersection point.

$\mathbf{o}' \leftarrow \mathbf{e}_r^T(\mathbf{o} - \mathbf{e}_c)$

$\mathbf{d}' \leftarrow \frac{\mathbf{e}_r^T \mathbf{d}}{||\mathbf{e}_r^T \mathbf{d}||_2}$ → Ray 의 coordinate 를 타원체의 local coordinate 로 변환

$\mathbf{t}_0 \leftarrow \mathbf{d}' \oslash \mathbf{e}_s$

$\mathbf{t}_1 \leftarrow \mathbf{o}' \oslash \mathbf{e}_s$ → Normalize 된 ray 를 타원체의 scale 에 맞게 조정

$\mathbf{t}_2 \leftarrow \frac{1}{\mathbf{e}_{s_x}\mathbf{e}_{s_y}\mathbf{e}_{s_z}}(\mathbf{d}' \times \mathbf{o}') \odot \mathbf{e}_s$

**if** $||\mathbf{t}_0||_2 < ||\mathbf{t}_2||_2$ **then**

  **return** No Intersection, $\varnothing$.

**else**

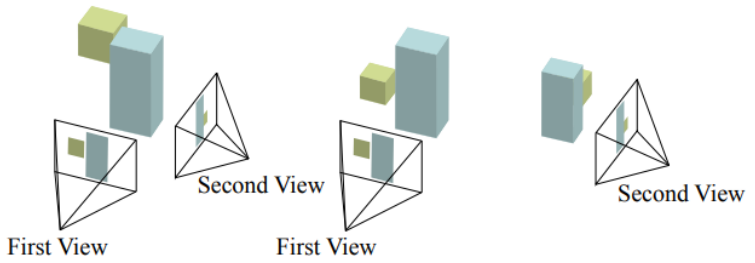
  $t \leftarrow -(\frac{\mathbf{t}_0}{||\mathbf{t}_0||_2} \cdot \frac{\mathbf{t}_1}{||\mathbf{t}_0||_2}) - \frac{1}{||\mathbf{t}_0||_2}\sqrt{1 - \frac{||\mathbf{t}_2||_2}{||\mathbf{t}_0||_2}}\sqrt{1 + \frac{||\mathbf{t}_2||_2}{||\mathbf{t}_0||_2}}$.

  **return** Intersected, $\mathbf{o} + t\mathbf{d}$.

외적을 통해 ray 와 타원체 사이의 각을 계산하고, 이를 기반으로 intersection 여부 결정

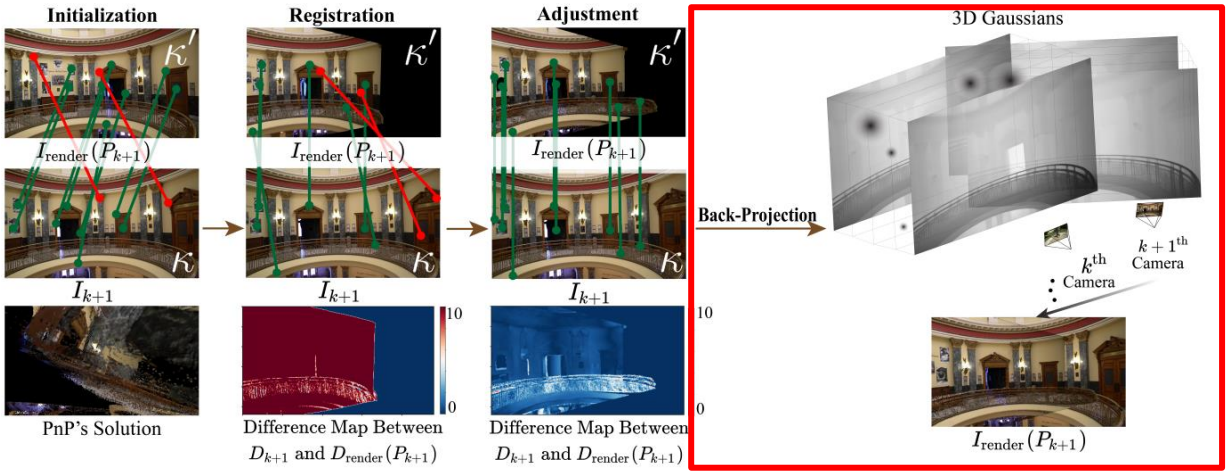
# Paper2- A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Main Method
  - Optimization Framework
    - Adjustment stage
      - 렌더링된 depth 를 사전 학습된 monocular depth 모델로 supervision 수행
        - ✓카메라 포즈를 정확하게 추정하기 위해서는 depth 또한 정렬되어야 함
          - Depth 와 같은 기하학 정보없이 카메라 포즈를 추정할 경우 물체의 위치가 맞지 않음에 따라 렌더링 성능이 크게 저할될 수 있음
      - Adjustment 단계에서는 $k+1$번째 까지 수행된 모든 카메라와 depth 가 함께 학습되어짐
        - ✓$D_{rendered}(s) = \sum_{i,\widehat{\mu_i}(s)\neq\emptyset} z_i(s)\alpha_i(s)\prod_{j=1,\widehat{\mu_j}(s)}^{i-1}(1-\alpha_j(s))$
        - ✓$\mathcal{L}_{depth} = \sum_{i=1}^{M}\left\| sg\left[b\left(\kappa'^{(i)}\right)\right] - d\left(\kappa^{(i)}\right)\right\|_1$
          - Depth 와 관련된 매개변수만을 학습



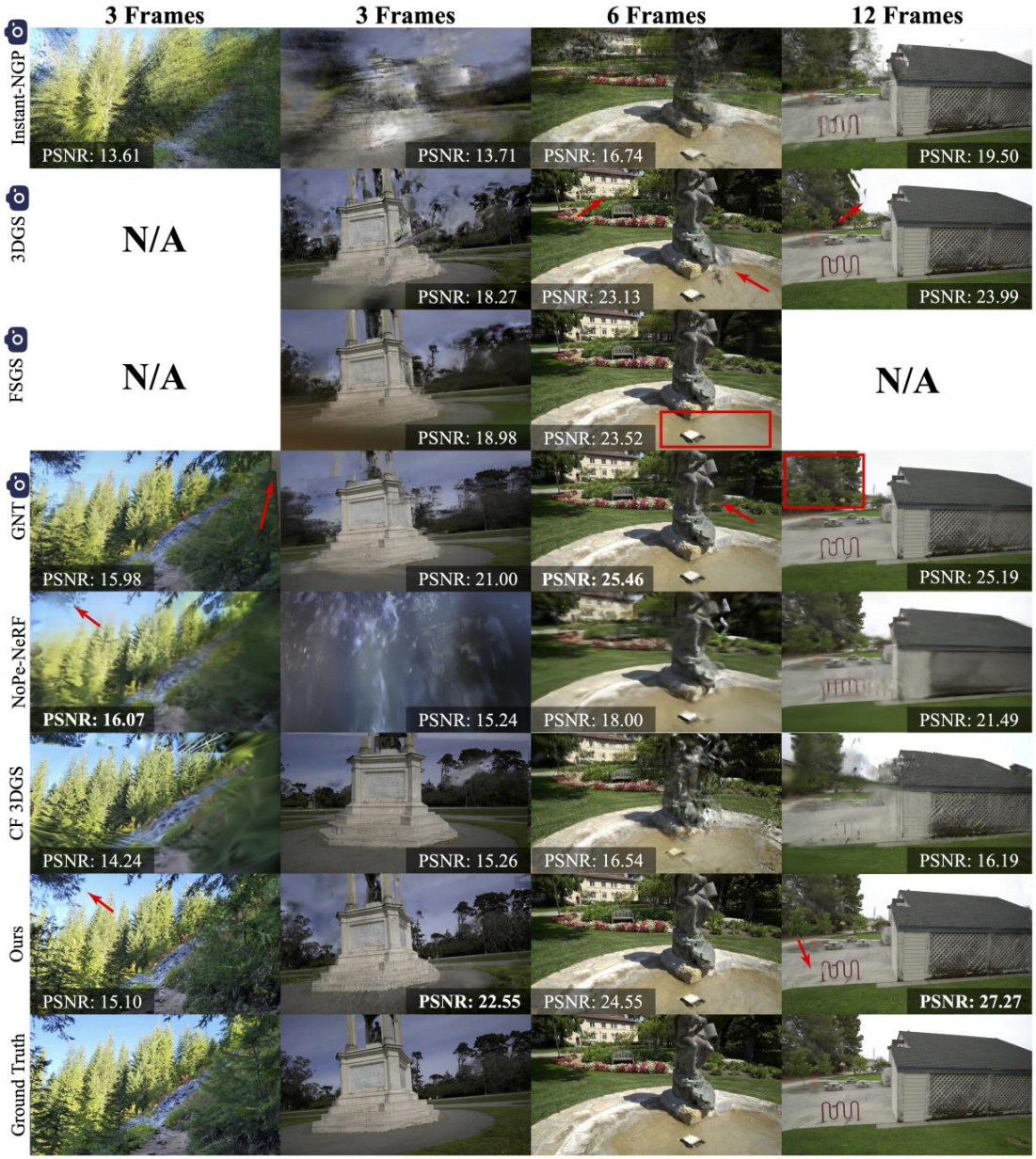
# Paper2- A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Main Method
  - Optimization Framework
    - Refinement
      - Registration 과 Adjustment 단계 이후 depth 기반 point cloud reconstruction 수행
        - ✓$k+1$번 째 카메라로 렌더링된 영역 중 빈 영역에 해당하는 3차원 정보 보완
        - ✓$k$ 까지 학습된 3D point 와 $k+1$에서 보완되는 3D point 중 겹치는 구간은 정보가 너무 과다해짐에 따라 noise 로 작용되어짐
          - 겹치는 구간에 한해 보완되는 3D point 개수를 10%로 제한
      - 업데이트된 3D 정보와 카메라 포즈를 기반으로 original 3D GS 모델 학습



# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Result
  - 정성 결과
    - 포즈 추정 모델
      - NoPe-NeRF
      - CF 3DGS
    - Only rendering 모델
      - Instant-NGP
      - 3DGS
      - FSGS
      - GNT
    - Sparse view 도 잘 수행 되어짐
      - 포즈 추정 및 렌더링 성능



# Paper2 - A Construct-Optimize Approach to Sparse View Synthesis without Camera Pose

- Result
  - 정량 결과

| Methods | 3 Views | | | 6 Views | | | 12 Views | | |
|---|---|---|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | LPIPS ↓ | PSNR ↑ | SSIM ↑ | LPIPS ↓ | PSNR ↑ | SSIM ↑ | LPIPS ↓ |
| Instant-NGP 📷 | 15.31 | 0.42 | 0.56 | 17.52 | 0.56 | 0.47 | 20.21 | 0.69 | 0.35 |
| 3DGS 📷 | 15.21 | 0.46 | 0.43 | 20.17 | 0.71 | 0.24 | 23.60 | 0.81 | 0.17 |
| FSGS† 📷 | 19.23 | 0.58 | 0.37 | 23.55 | 0.74 | 0.28 | 26.81 | 0.83 | 0.22 |
| GNT 📷 | 17.80 | 0.57 | 0.29 | 22.52 | 0.77 | 0.18 | 24.56 | 0.82 | 0.14 |
| LocalRF | 16.06 | 0.49 | 0.70 | 16.31 | 0.50 | 0.67 | 18.68 | 0.54 | 0.61 |
| NoPe-NeRF | 12.05 | 0.35 | 0.76 | 15.64 | 0.45 | 0.65 | 18.12 | 0.49 | 0.60 |
| CF 3DGS | 14.91 | 0.43 | 0.43 | 16.71 | 0.50 | 0.41 | 18.62 | 0.59 | 0.36 |
| Ours | **20.37** | **0.66** | **0.26** | **25.18** | **0.81** | **0.16** | **28.65** | **0.88** | **0.10** |

- 촬영 영역이 비교적 좁고 수렴형태로 촬영한 데이터셋에 대해 SOTA 성능 달성

| Methods | 3 Views | | | 6 Views | | | 12 Views | | |
|---|---|---|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | LPIPS ↓ | PSNR ↑ | SSIM ↑ | LPIPS ↓ | PSNR ↑ | SSIM ↑ | LPIPS ↓ |
| LocalRF | 15.97 | 0.33 | 0.47 | 18.32 | 0.47 | 0.43 | **20.13** | **0.54** | 0.41 |
| NoPe-NeRF | 14.85 | 0.25 | 0.67 | 18.59 | 0.34 | 0.57 | 18.19 | 0.34 | 0.59 |
| CF 3DGS | 15.45 | 0.28 | 0.60 | 17.02 | 0.35 | 0.52 | 17.65 | 0.39 | 0.46 |
| Ours | **16.35** | **0.38** | **0.37** | **18.96** | **0.50** | **0.31** | 19.70 | 0.53 | **0.29** |

- 촬영 영역이 넓고(camera trajectory 가 매우 큼) 긴 장면의 경우, view 가 많을 때를 제외하고는 SOTA 성능 달성

# 감사합니다