Learning to Render Novel Views from Wide-Baseline Stereo Pairs

Yilun Du, Cameron Smith, Ayush Tewari, Vincent Sitzmann

CVPR 2023

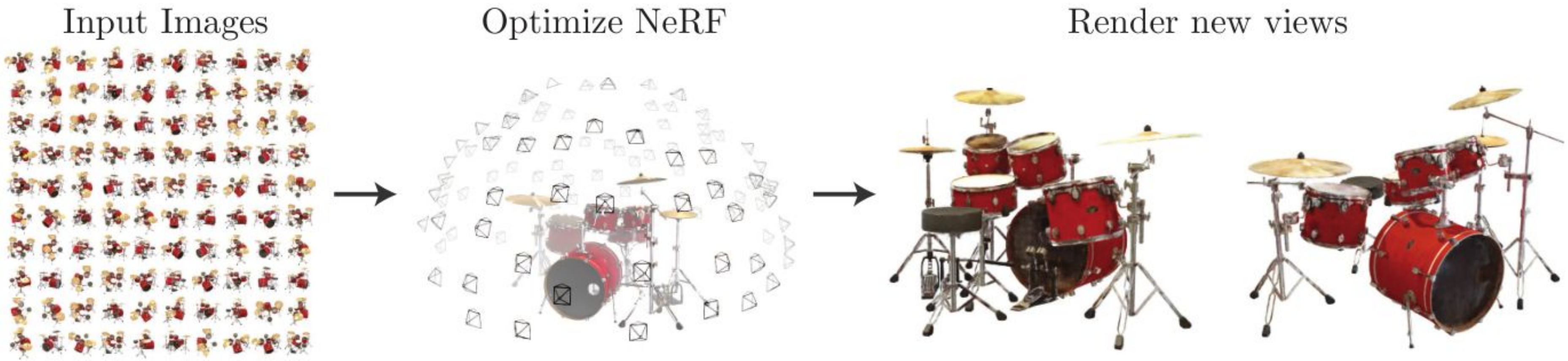
Group4: 劉冠宏 311551058、周睦鈞 311553060、黃柏叡 311553015

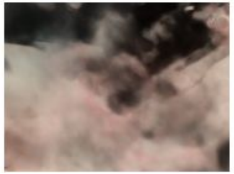
Outline

- Introduction
- Method
- Experiment
- Conclusion
- Reproduce
- Q & A

Introduction

Novel View Synthesis - NeRF





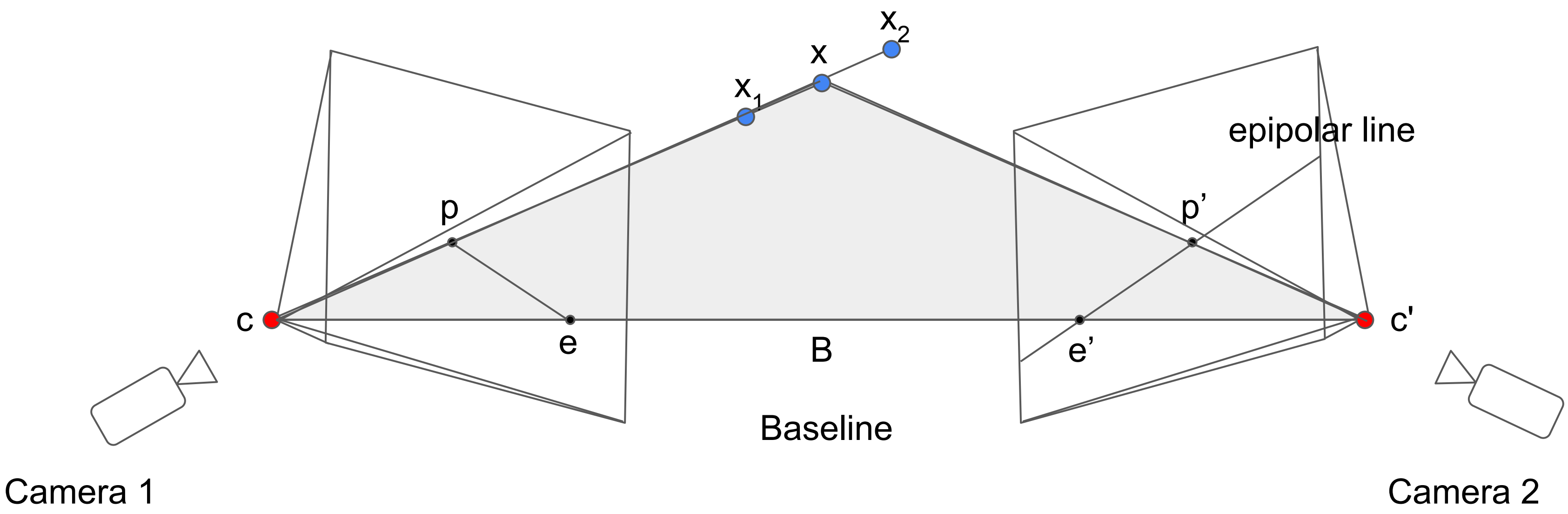
Novel View Synthesis

Novel View Synthesis - This Paper

Contributions

- Achieve state-of-the-art on challenging setting of novel view synthesis using only a wide-baseline stereo pair of images
- Propose the image-centric epipolar line sampling strategy with lightweight cross-attention based renderer instead of volume rendering
  - Image-centric smapling enable the model to fully utilize image features
  - Rendering is faster than most of the prior art, which enable them to train on large-scale real-world complex datasets

Preliminary - Epipolar Geometry



Method

Model Architecture

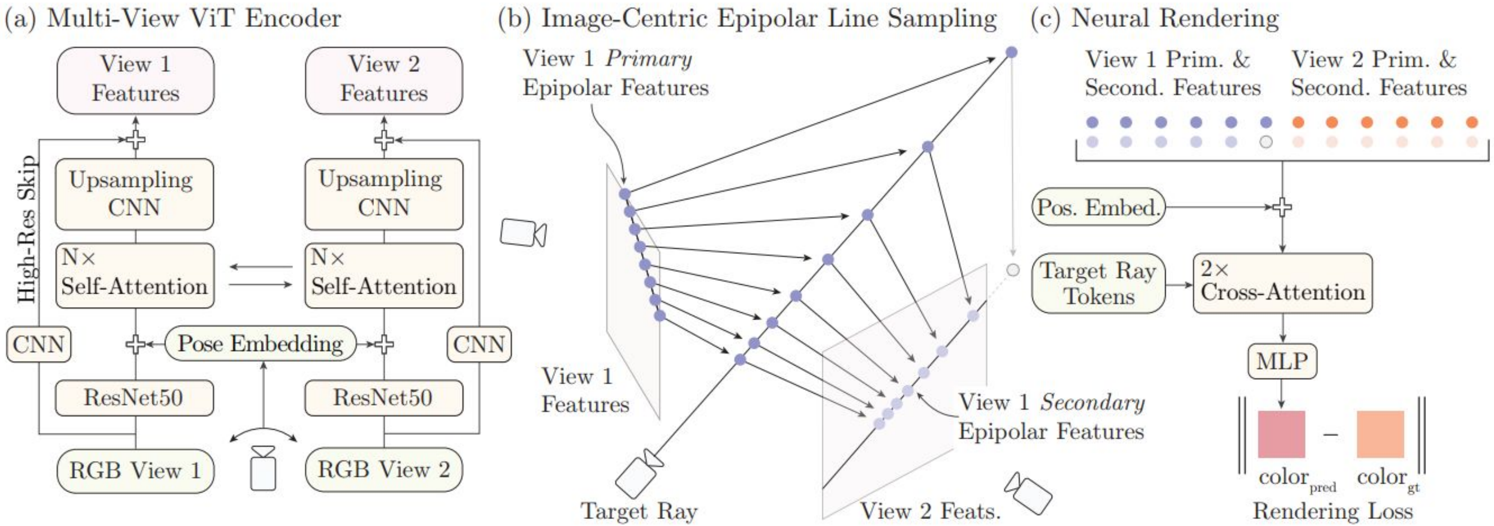


Figure 2. **Method Overview.** (a) Given context images from different viewpoints, a multi-view encoder extracts pixel-aligned features, leveraging attention across the images and their corresponding camera pose embeddings. (b) Given a target ray, in each context view, we sample *primary* features along the epipolar line equidistant in pixel space. We then project the corresponding 3D points onto the other views and sample corresponding *secondary* epipolar line features, where out-of-bounds features are set to zero. (c) We render the target ray by performing cross-attention over the set of all primary and secondary epipolar line features from all views.

Multiview Feature Encoding

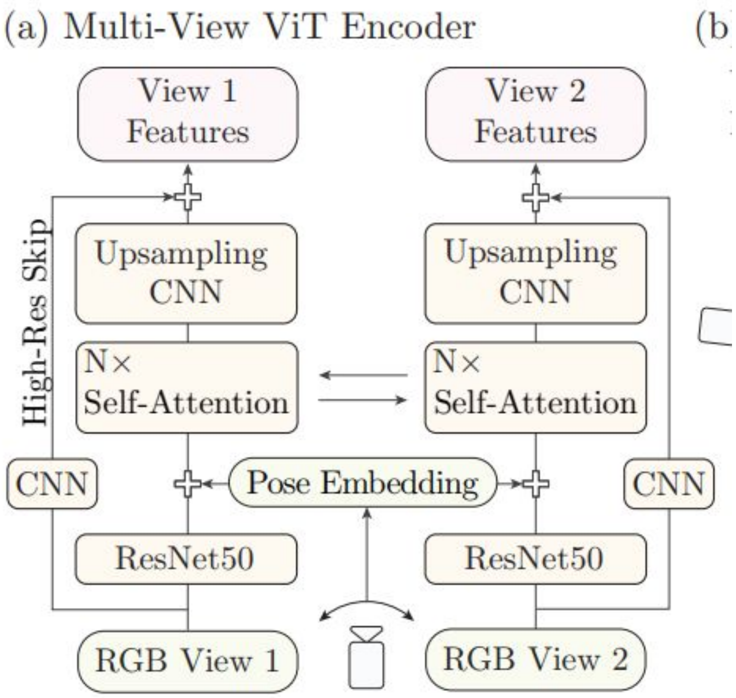
Problem:

1. Artifacts in rendering results.
2. Encoding each image separately lead to inconsistent geometry reconstruction across context images.

Solution:

1. Processing the images simultaneously.
2. But how?

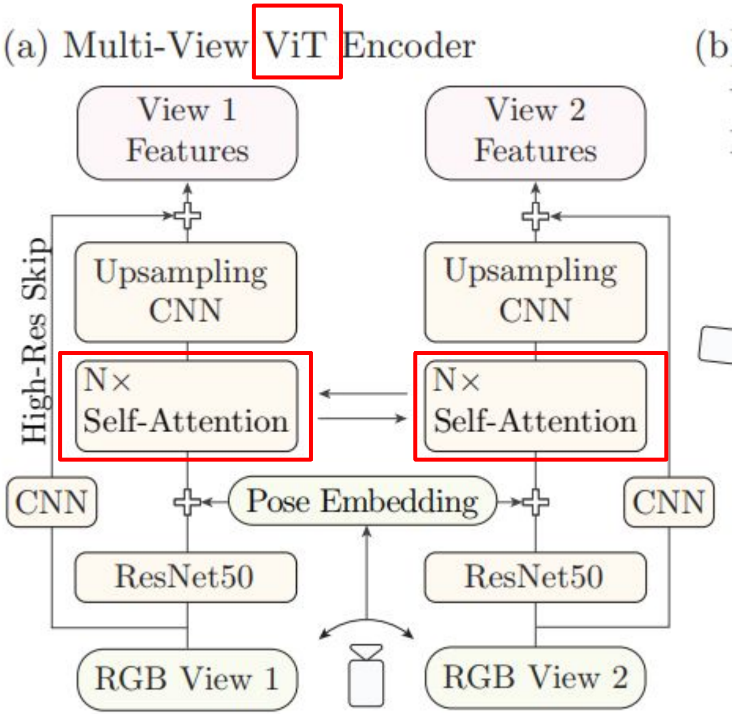
Multiview Feature Encoding



Multiview Feature Encoding

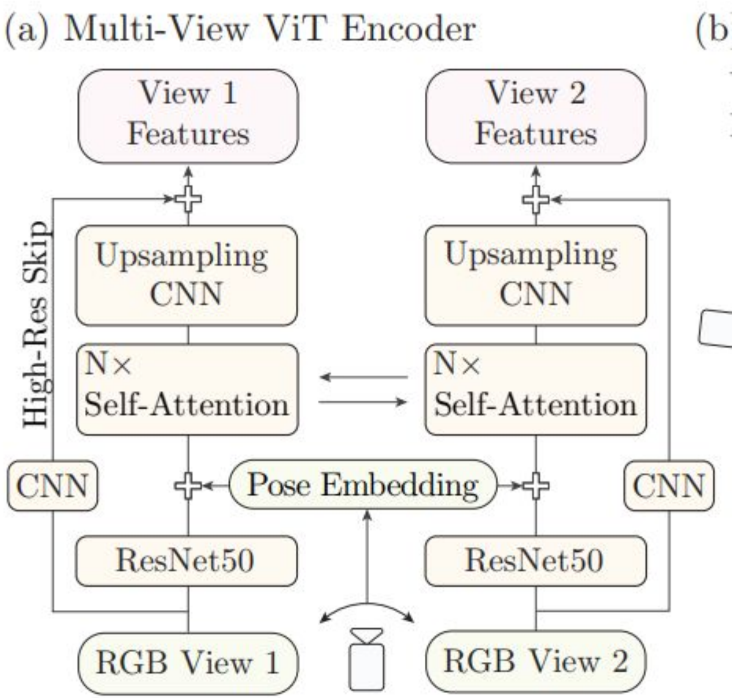
- ViT (Vision Transformer)

- Transformer
  - A better CNN
  - Mechanism: self-attention



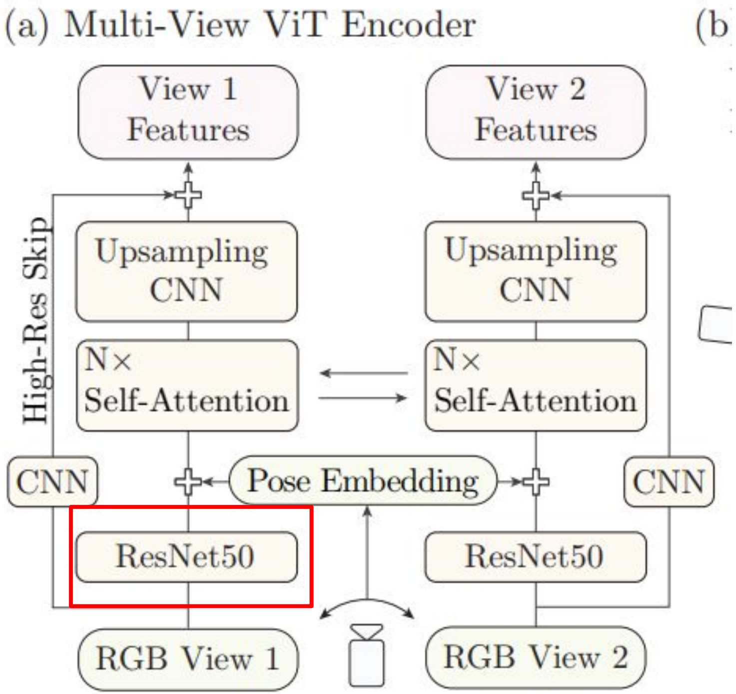
Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Upsampling
- Bypass the high resolution imformation



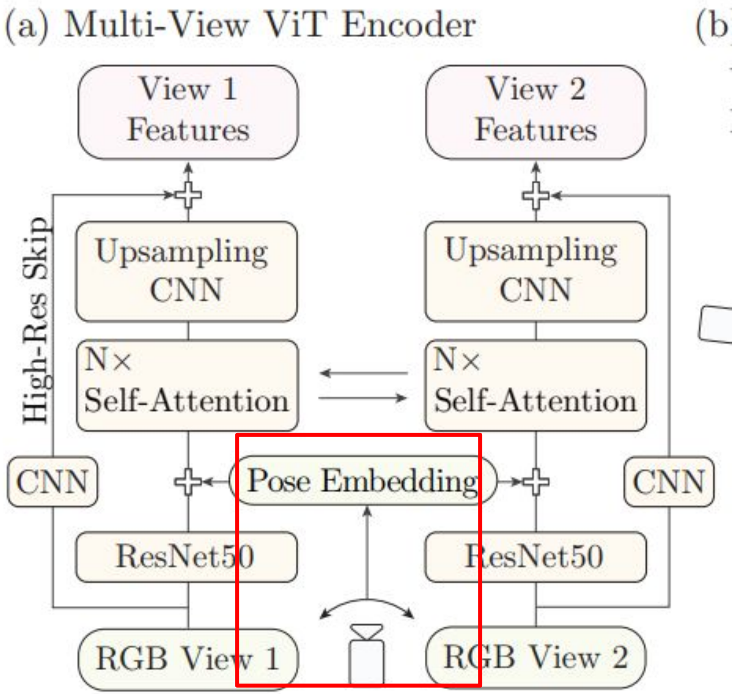
Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Upsampling
- Bypass the high resolution imformation



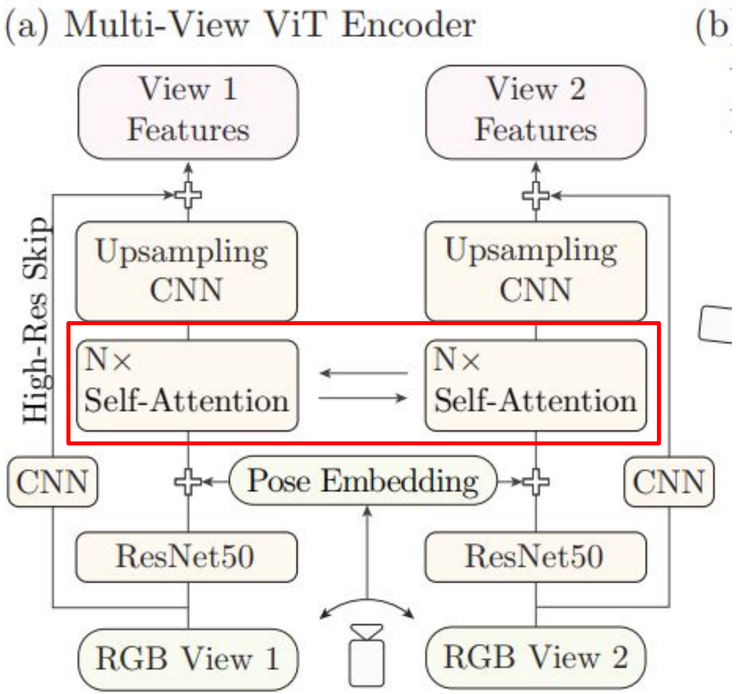
Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Upsampling
- Bypass the high resolution imformation



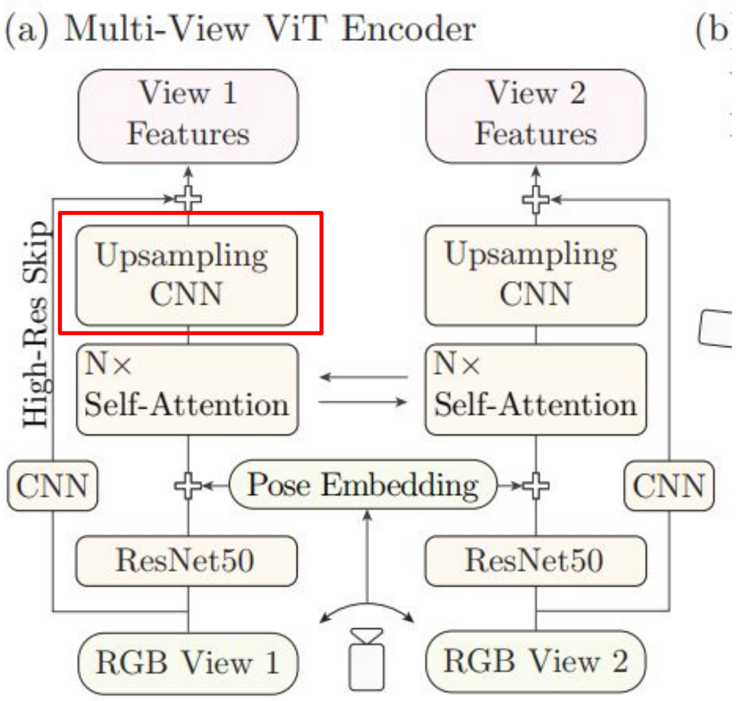
Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Upsampling
- Bypass the high resolution imformation



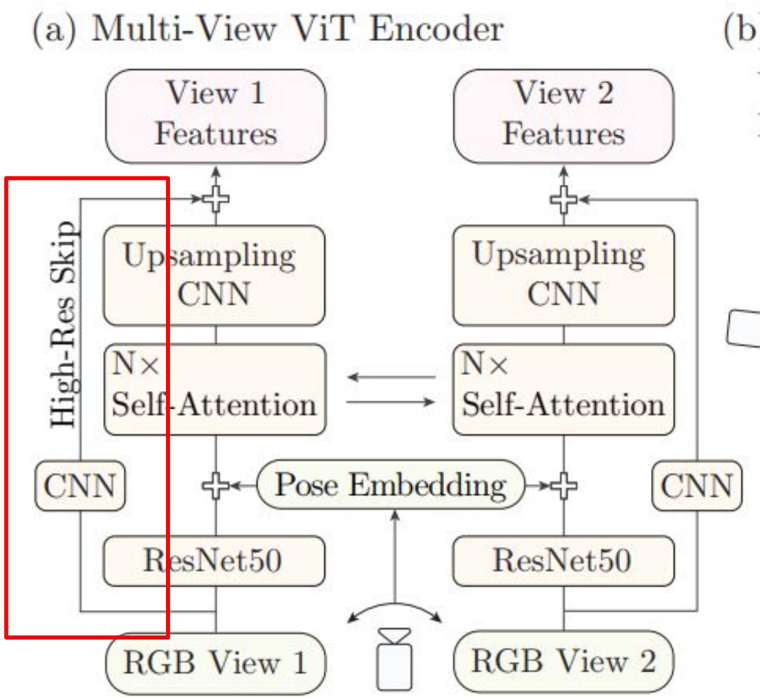
Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Sequence to spatial data
- Bypass the high resolution imformation



Multiview Feature Encoding

- Feature Extraction
- Embedding
- Transformer
- Simultaneous processing
- Sequence to spatial data
- Bypass the high resolution imformation



Epipolar Line Sampling and Feature Matching

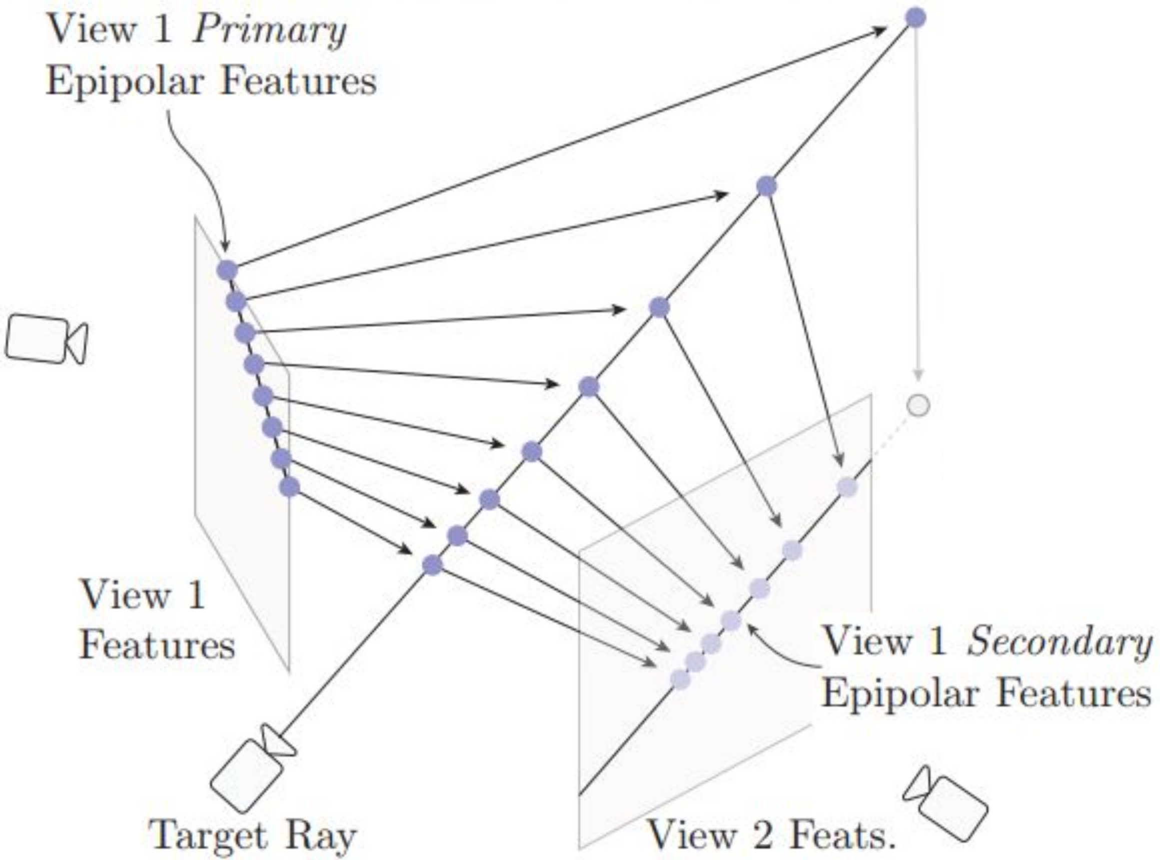
Problem:

1. Volume rendering is not suitable.
2. The number of pixels along the epipolar line should maximum effect the results.

Solution:

1. Epipolar line

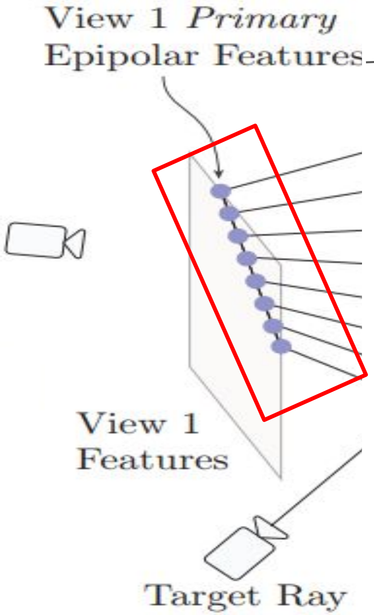
(b) Image-Centric Epipolar Line Sampling



Epipolar Line Sampling

- Uniformly sample N pixel coordinates along the line segment
- The depth value is computed via trangulation

$$\{(d, \boxed{\mathbf{f}})_k\}_{k=1}^{N}$$

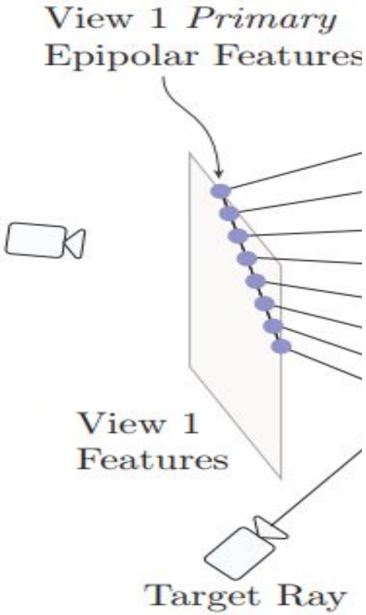


from muti-view ViT encoder

Epipolar Line Sampling

- Uniformly sample N pixel coordinates along the line segment
- The depth value is computed via trangulation

$$\{(\boxed{d,}\,\mathbf{f})_k\}_{k=1}^{N}$$



from muti-view ViT encoder

D. Triangulation

Here, we provide details on computing 3D points using triangulation. For a pixel coordinate in the context image (u', v'), we may solve for its corresponding 3D point via:

$$l^* = \arg\min_{l} \|\pi_t(\mathbf{o}_i + l \cdot \mathbf{R}_i^{-1}\mathbf{K}_i^{-1}[u', v', 1]) - \mathbf{u}_t\|_2^2, \tag{5}$$

where $\mathbf{o}_i$ is the camera origin of the respective context image, $\pi_t(\cdot)$ denotes projection onto the target camera, and $\mathbf{u}_t$ is the pixel coordinate of the target ray we aim to render. The 3D point $\mathbf{p}^*$ can then be obtained as $\mathbf{p}^* = \mathbf{o}_i + l^* \cdot \mathbf{R}_i^{-1}\mathbf{K}_i^{-1}[u', v', 1]$, and its depth in the context camera can be obtained as the z-coordinate of the point in the context camera's coordinates. Let $\mathbf{r}_i$ denotes the normalized ray direction $\mathbf{R}_i^{-1}\mathbf{K}_i^{-1}[u', v', 1]$. The closed form solution can be represented as:

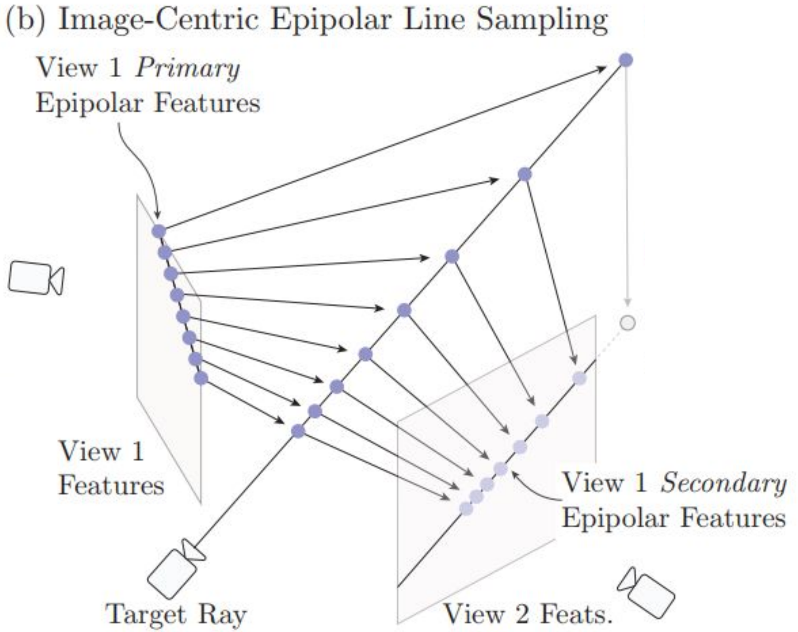
$$\begin{aligned} l^* &= \frac{u \cdot \mathbf{o}_i[z] - c_x \mathbf{o}_i[z] - f_x \mathbf{o}_i[x]}{f_x \mathbf{r}_i[x] + c_x \mathbf{r}_i[x] - u\mathbf{r}_i[z]} \\ &= \frac{v \cdot \mathbf{o}_i[z] - c_y \mathbf{o}_i[z] - f_y \mathbf{o}_i[y]}{f_y \mathbf{r}_i[y] + c_y \mathbf{r}_i[y] - u\mathbf{r}_i[z]}, \end{aligned}$$

where $\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$.

Feature Matching

- Refine the geometry information
- Primary features -> 3D points -> secondary features

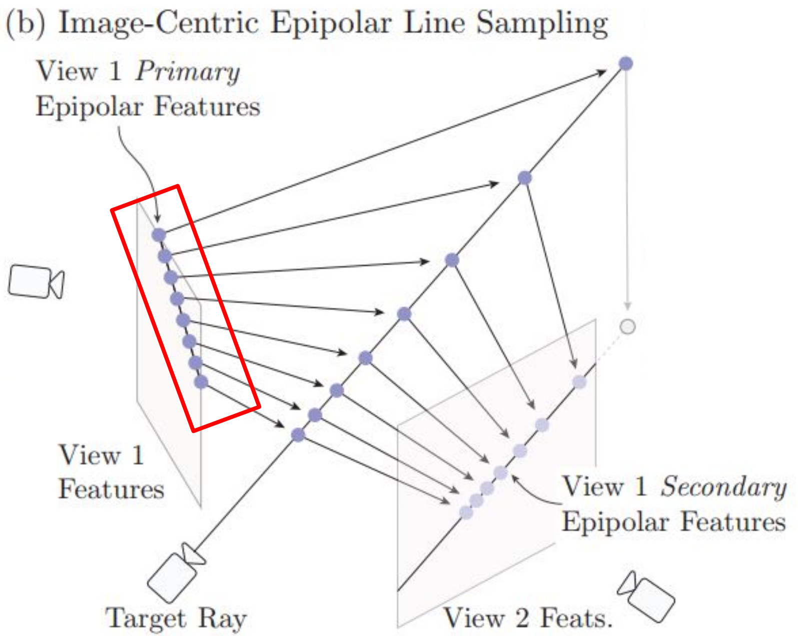
$$\{(d, \mathbf{f}, \hat{\mathbf{f}})_k\}_{k=1}^{2N}$$



Feature Matching

- Refine the geometry information
- Primary features -> 3D points -> secondary features

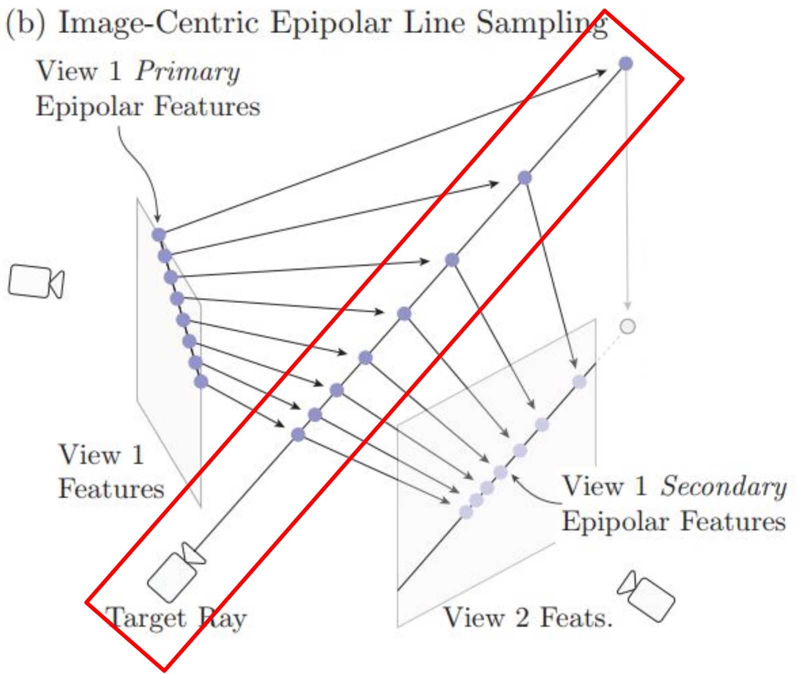
$$\{(d, \mathbf{f}, \hat{\mathbf{f}})_k\}_{k=1}^{2N}$$



Feature Matching

- Refine the geometry information
- Primary features -> 3D points -> secondary features

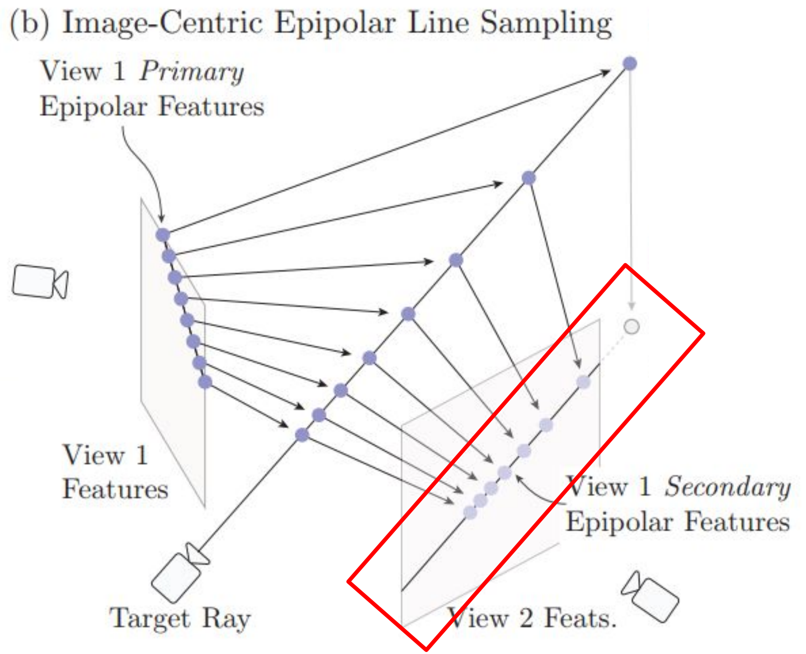
$$\{(d, \mathbf{f}, \hat{\mathbf{f}})_k\}_{k=1}^{2N}$$



Feature Matching

- Refine the geometry information
- Primary features -> 3D points -> secondary features

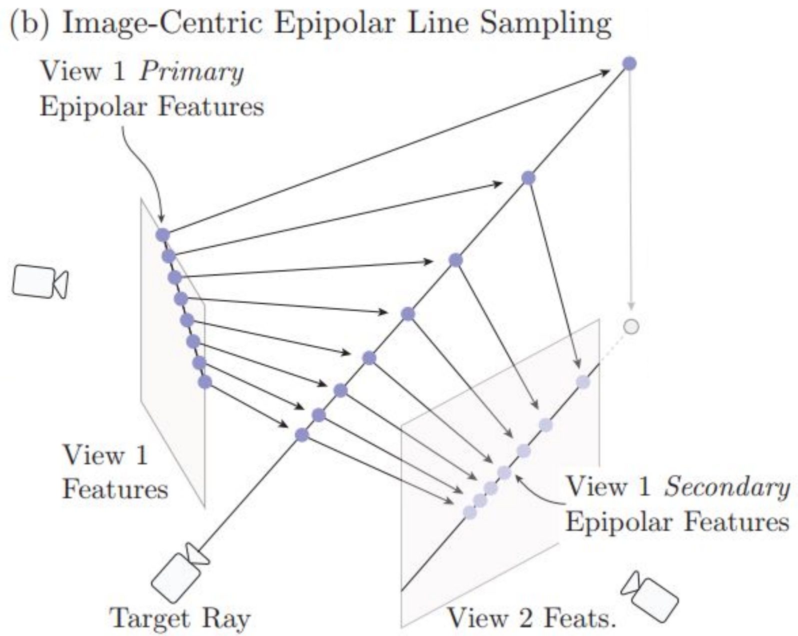
$$\{(d, \mathbf{f}, \hat{\mathbf{f}})_k\}_{k=1}^{2N}$$



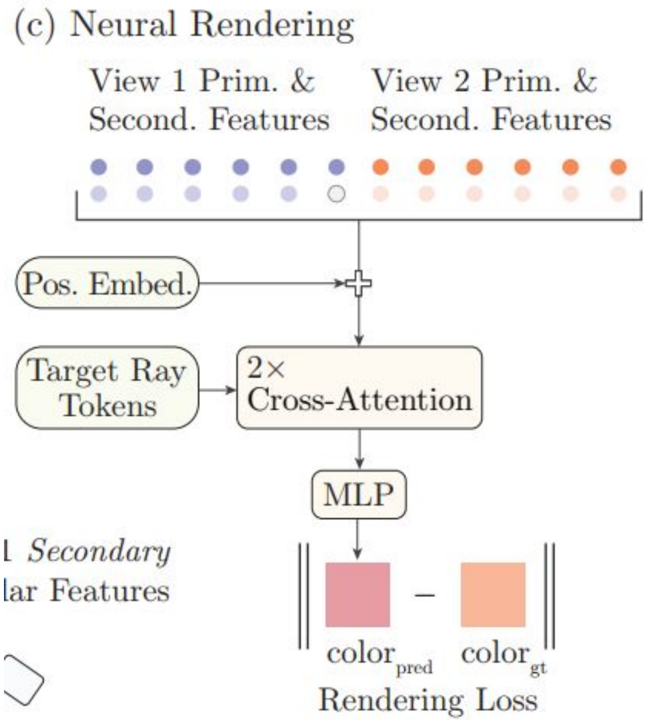
Feature Matching

- Refine the geometry information
- Primary features -> 3D points -> secondary features

$$\boxed{\{(d, \mathbf{f}, \hat{\mathbf{f}})_k)_{k=1}^{2N}}$$



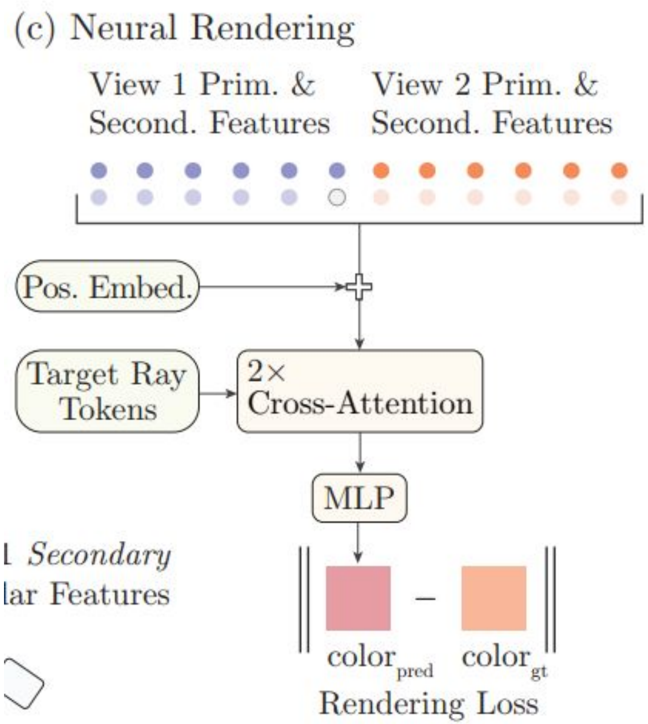
Differentiable Rendering via Cross-Attention



Differentiable Rendering via Cross-Attention

- Input: $\{(d, \mathbf{f}, \hat{\mathbf{f}})_k\}_{k=1}^{2N}$
- Output: color

- Target ray origin, target ray direction, depth, context camera ray direction.
- Final feature embedding
- Decoded into color



Training and Losses

- LPIPS perceptual loss
- Regularization loss

$$\mathcal{L} = \mathcal{L}_{\text{img}} + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}$$

$$\mathcal{L}_{\text{img}} = ||R - G||_1 + \lambda_{\text{LPIPS}} \mathcal{L}_{\text{LPIPS}}(R, G)$$

- Data augmentation
    - crop
    - scale
    - filp

$$\mathcal{L}_{\text{reg}} = \sum_{(u,v)} \sum_{(u'v') \in \mathcal{N}(u,v)} ((e(u,v) - e(u',v'))^2$$

Neighbor

Experiment

Dataset

- RealEstate10k

  A large dataset of indoor and outdoor scenes

- ACID

  A large dataset outdoor scenes

Qualitative Results (Indoor)

Qualitative Results (Outdoor)

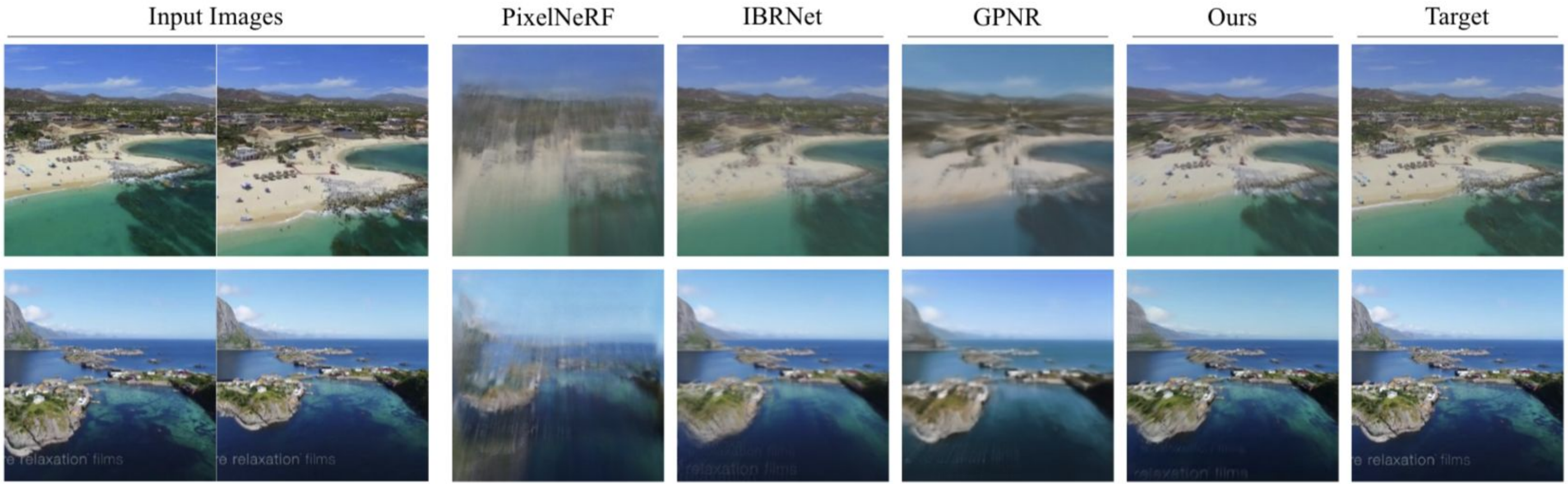


Figure 5. **Comparative Results on ACID.** Our approach is able to render novels views with higher quality than all baselines.

Quantitative Results

Method	LPIPS ↓	SSIM ↑	PSNR ↑	MSE ↓
pixelNeRF [59]	0.591	0.460	13.91	0.0440
IBRNet [54]	0.532	0.484	15.99	0.0280
GPNR [48]	0.459	0.748	18.55	0.0165
Ours	**0.262**	**0.839**	**21.38**	**0.0110**

Table 1. **Novel view rendering performance on RealEstate10K.** Our method outperforms all baselines on all metrics.

Method	LPIPS ↓	SSIM ↑	PSNR ↑	MSE ↓
pixelNeRF [59]	0.628	0.464	16.48	0.0275
IBRNet [54]	0.385	0.513	19.24	0.0167
GPNR [48]	0.558	0.719	17.57	0.0218
Ours	**0.364**	**0.781**	**23.63**	**0.0074**

Table 2. **Novel view rendering performance on ACID.** Our method outperforms all baselines on all metrics.

Ablation study

Models	LPIPS↓	SSIM↑	PSNR↑	MSE↓
Base Model	0.452	0.735	18.11	0.0201
+ 2D Sampling	0.428	0.762	19.02	0.0159
+ Cross Correspondence	0.415	0.766	19.52	0.0142
+ Multiview Encoder	0.361	0.794	20.43	0.0132
+ Regularization Loss	0.358	0.808	19.84	0.0139
+ Data Aug	0.262	0.839	21.38	0.0110

Conclusion

Conclusion

- Introduce a method for implicit 3D reconstruction and novel view synthesis from a single, wide-baseline stereo pair

- Our method surpasses the quality of prior art on datasets of challenging scenes

Reproduce

Reproduce results

Method	LPIPS ↓	SSIM ↑	PSNR ↑	MSE ↓
GPNR	0.459	0.748	18.55	0.0165
Paper	0.262	0.839	21.38	0.0110
Pretrained	0.317	0.809	21.11	0.0112
Reproduce	0.355	0.773	19.50	0.0161

First Experiment

Second Experiment

Discussion

- Two input images should not be too far from each other to get better qualitative results
- Relies on learned priors(e.g. pose embedding), it does not generalize well to new scene with very different appearances compared to training scenes

Q & A