Generalizable Neural Rendering for Novel View Synthesis

Qianqian Wang
Cornell University

Novel View Synthesis

Source views

Synthesize

Novel view

Image-Based Rendering (IBR)

Applications

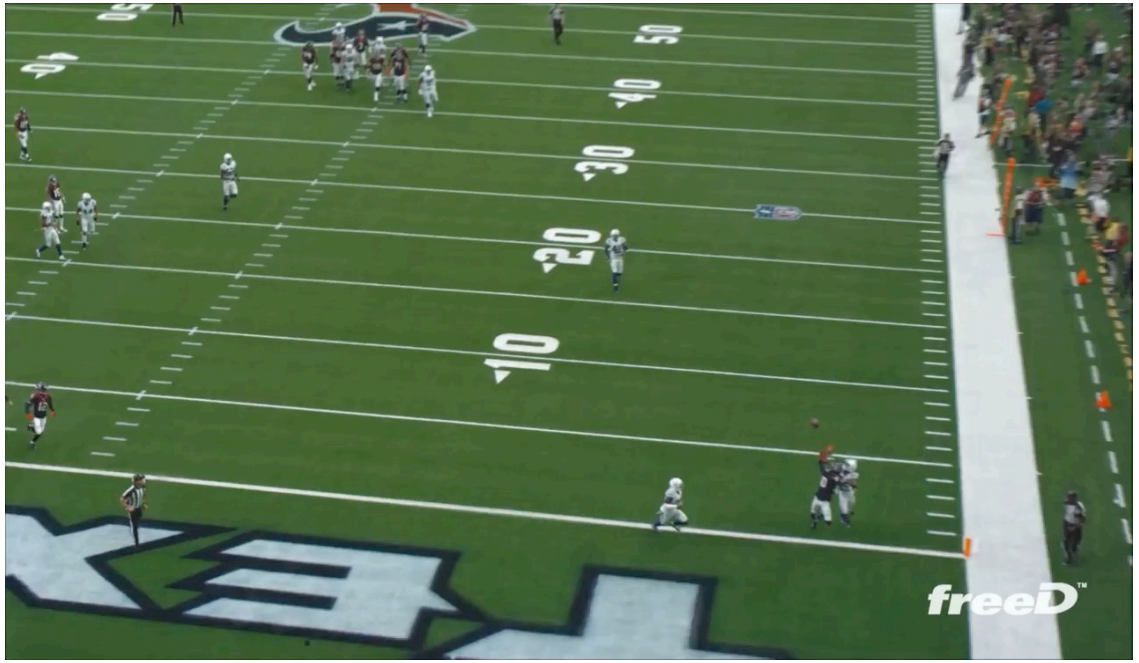
- VR Tour

Matterport

https://realsee.com/website/product/vr

Applications

- Free-viewpoint video & bullet time effect

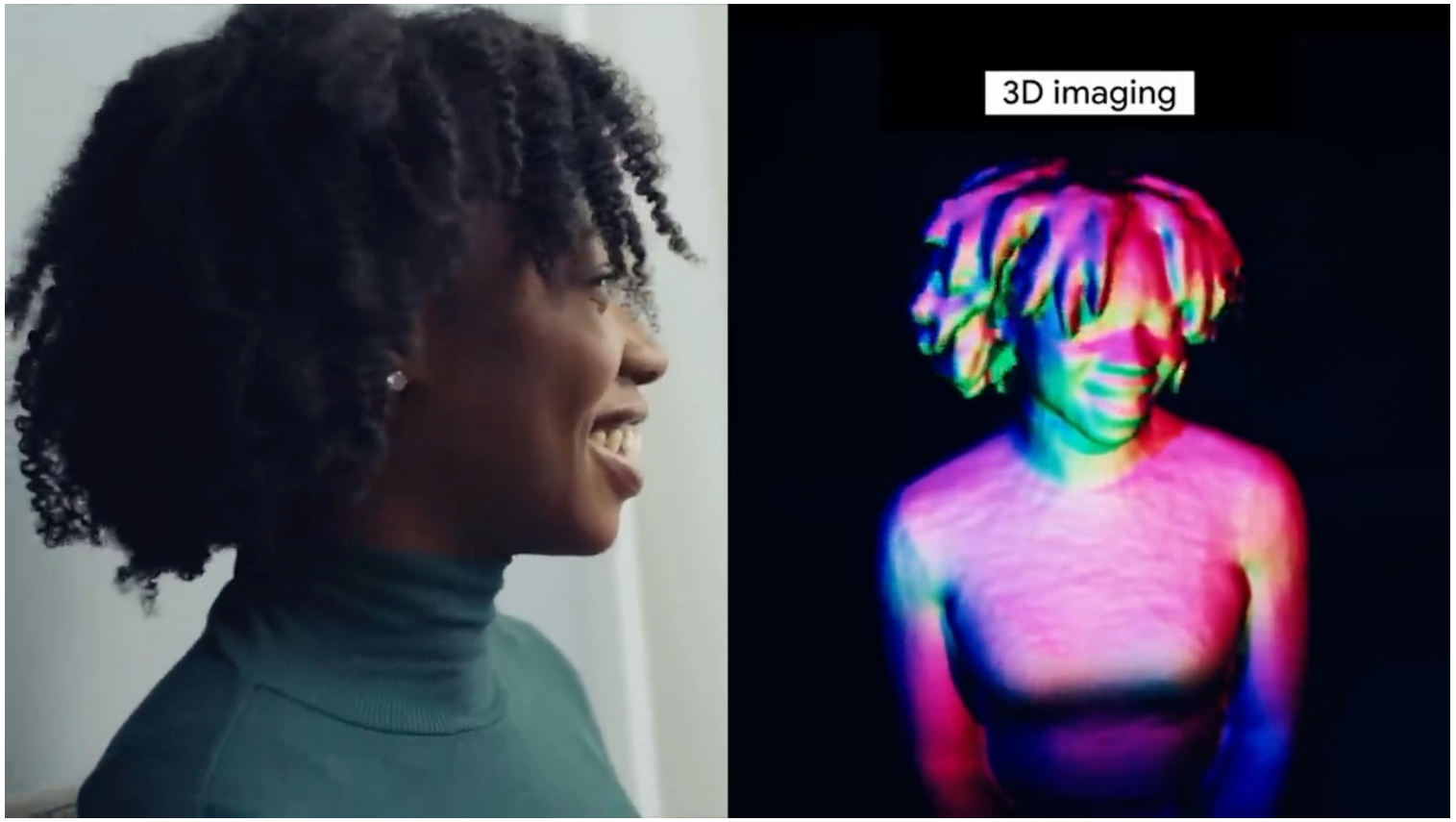


Intel True View

Dance Smash, Mango TV

Applications

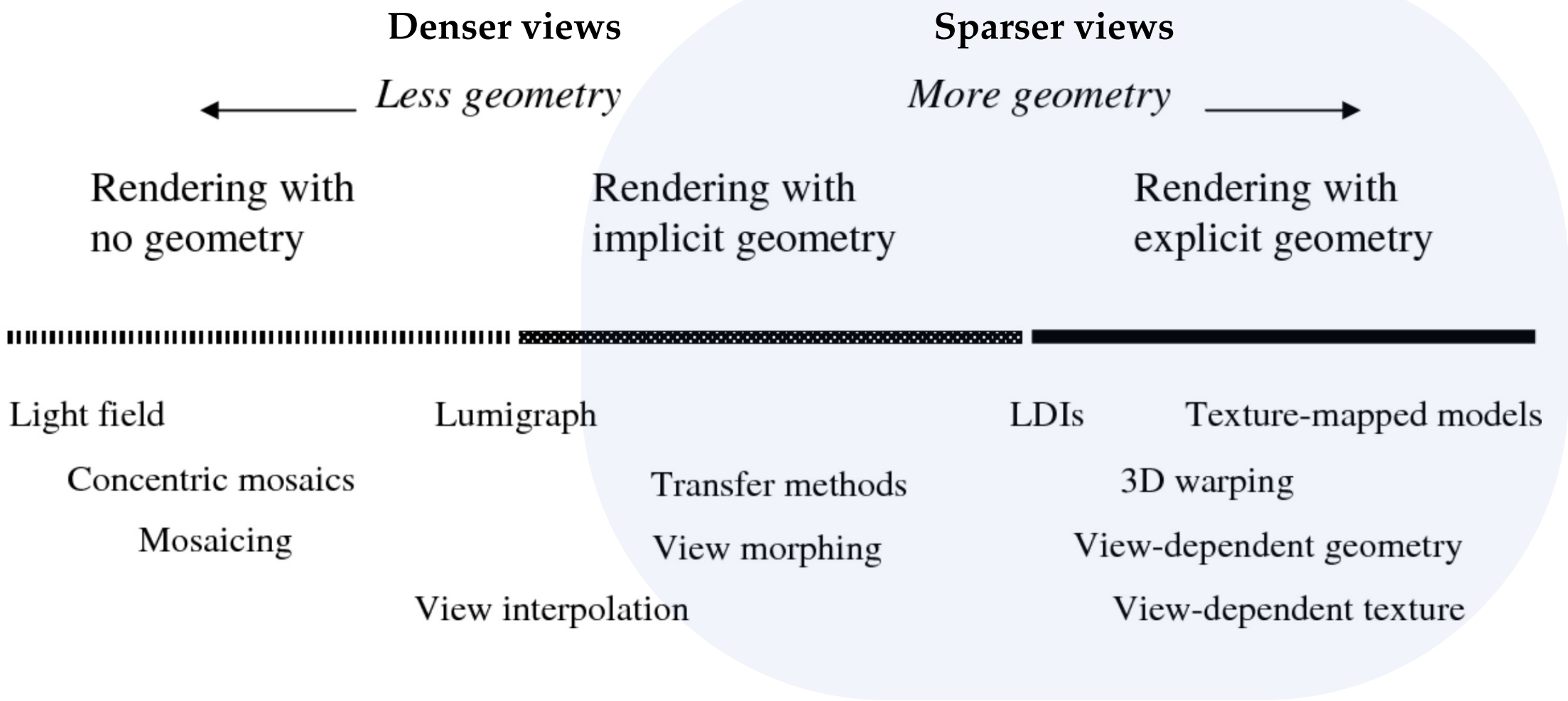
- Immersive tele-communication



Google Starline

https://blog.google/technology/research/project-starline/

IBR spectrum

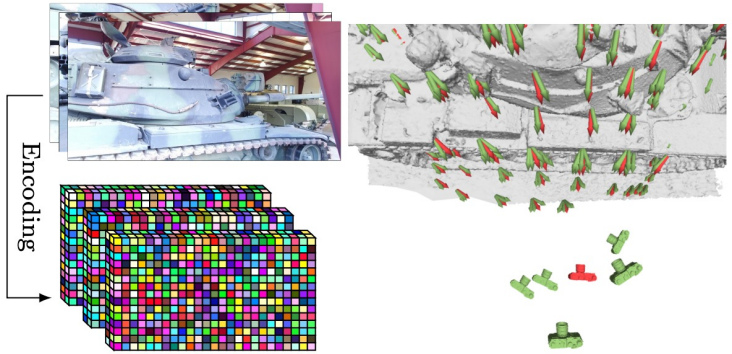


Shum, H. Y., Chan, S. C., & Kang, S. B. (2008). *Image-based rendering*. Springer Science & Business Media.

Learned texture-mapped meshes

Meshes reconstructed by
multi-view stereo (MVS) algorithms

Source views



Learned appearance by
warping/aggregating source views

Target view

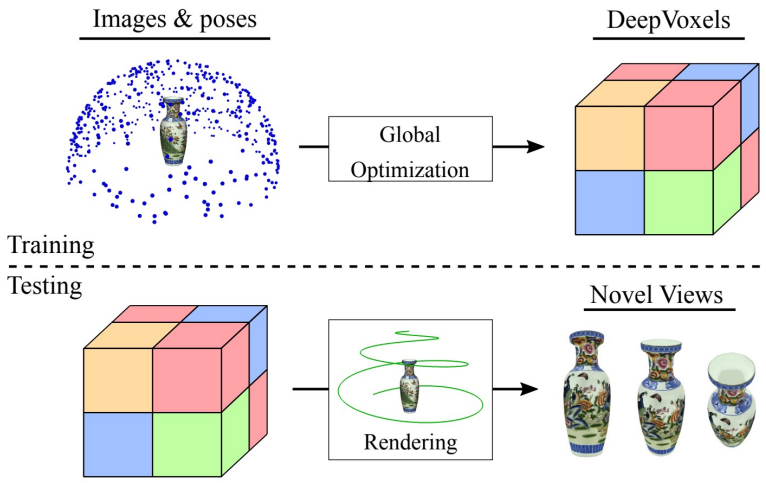
Riegler, Gernot, and Vladlen Koltun. "Stable view synthesis." CVPR. 2021.

Learned texture-mapped meshes

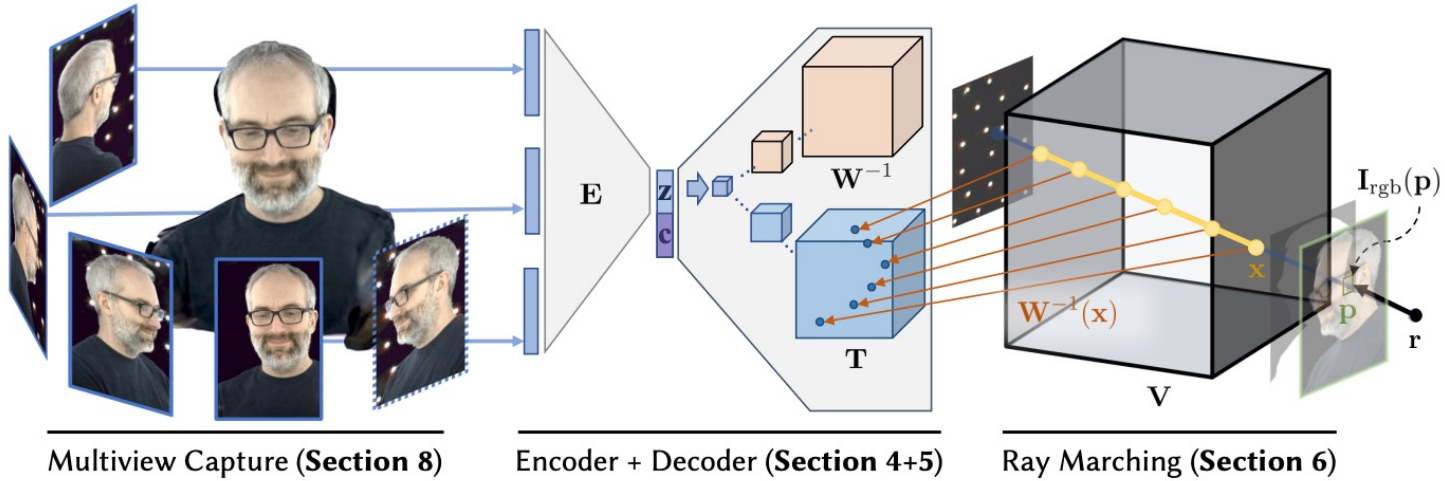
- **Advantages**
  - Relatively scalable
  - Handles sparse views well
  - The learned texture model is generalizable

- **Limitations**
  - Limited by the performance of MVS algorithms

Volumetric Representations



DeepVoxels



Neural Volumes

Construct a discretized 3D volume as the scene representation
(DeepVoxels: feature volume; Neural Volumes: RGB-α volume)

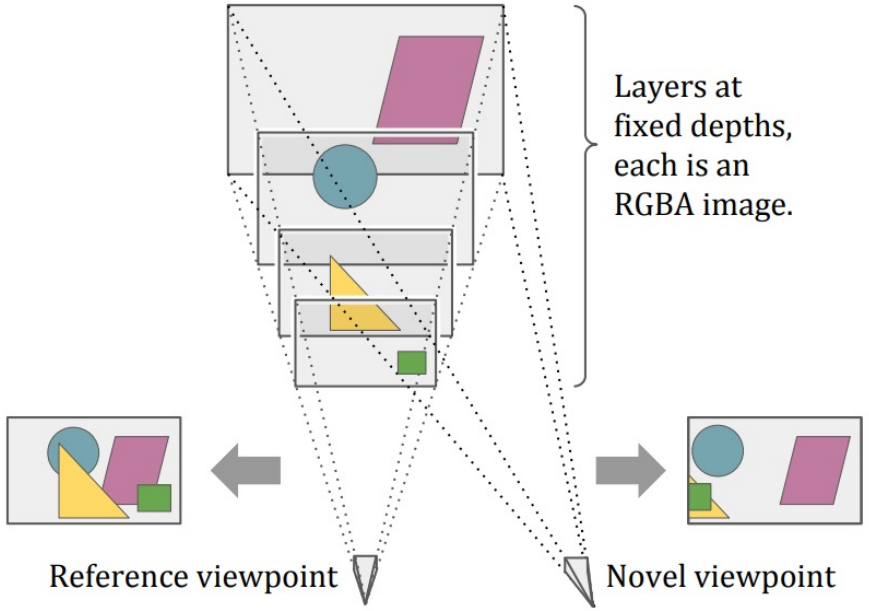
Volumetric Representations

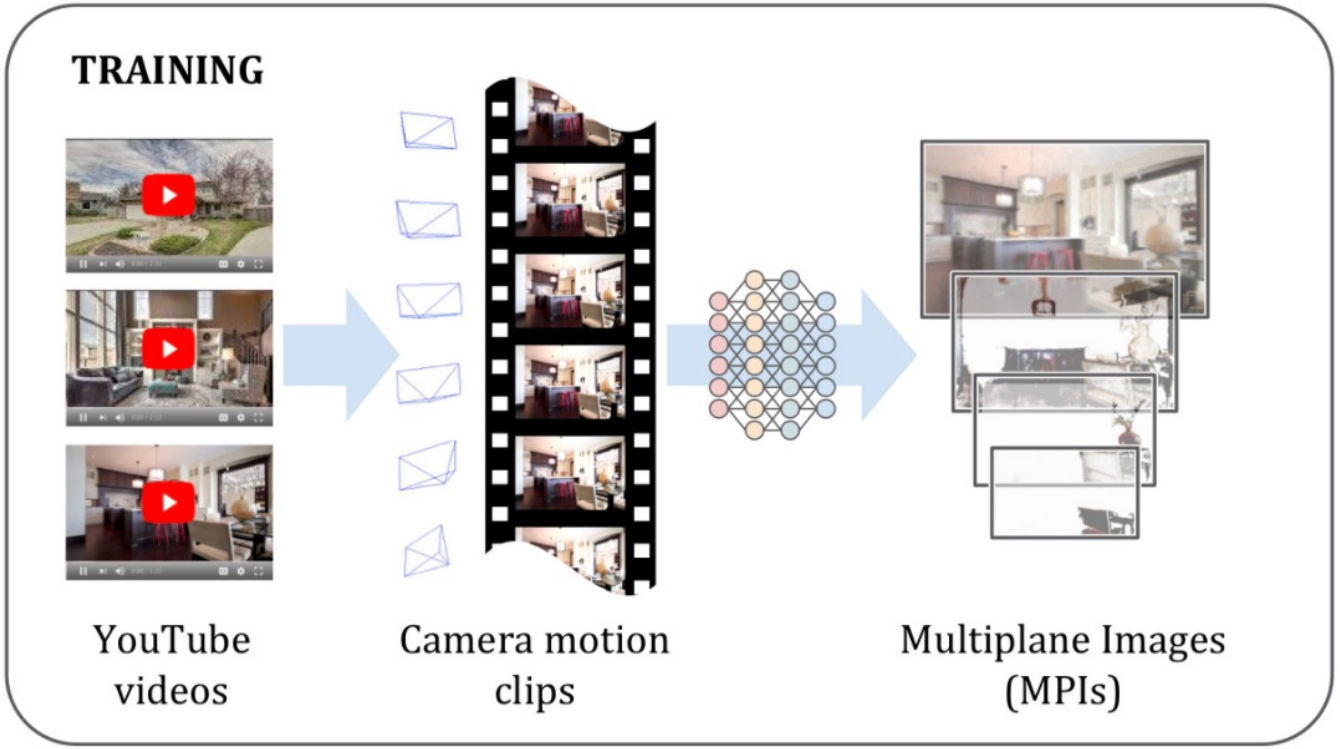
- **Advantages**
  - Can handle partially-transparent objects like smoke
  - Learned end-to-end (does not require proxy geometry as input)

- **Limitations**
  - Does not scale well to large scenes
  - Scene-specific (needs to train a model for each scene specifically)

Multi-Plane Images (MPI)

- Multi-Plane Image (MPI) is a set of front-parallel RGBA planes at fixed depths



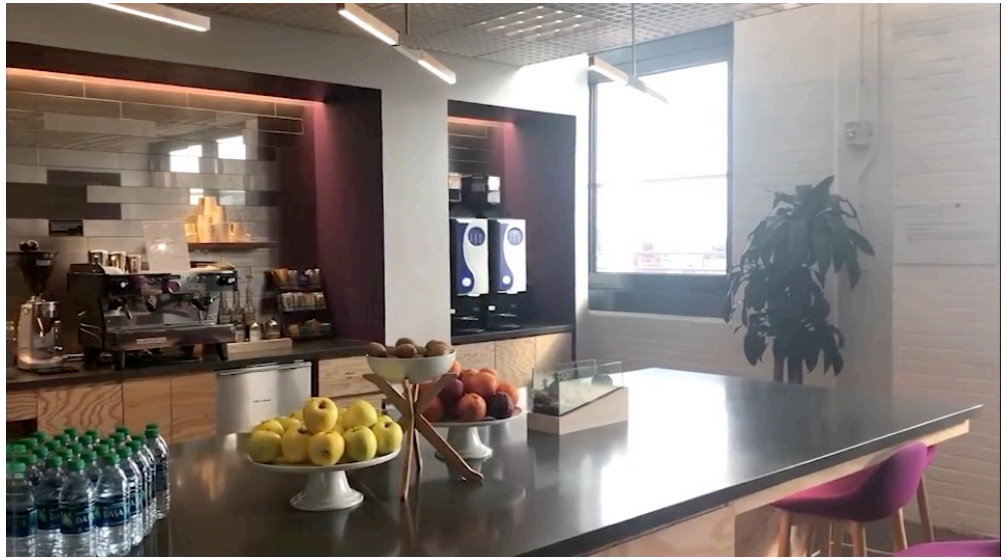


Tinghui Zhou, et al. Stereo Magnification: Learning View Synthesis using Multiplane Images, In SIGGRAPH 2018

Multi-Plane Images (MPI)

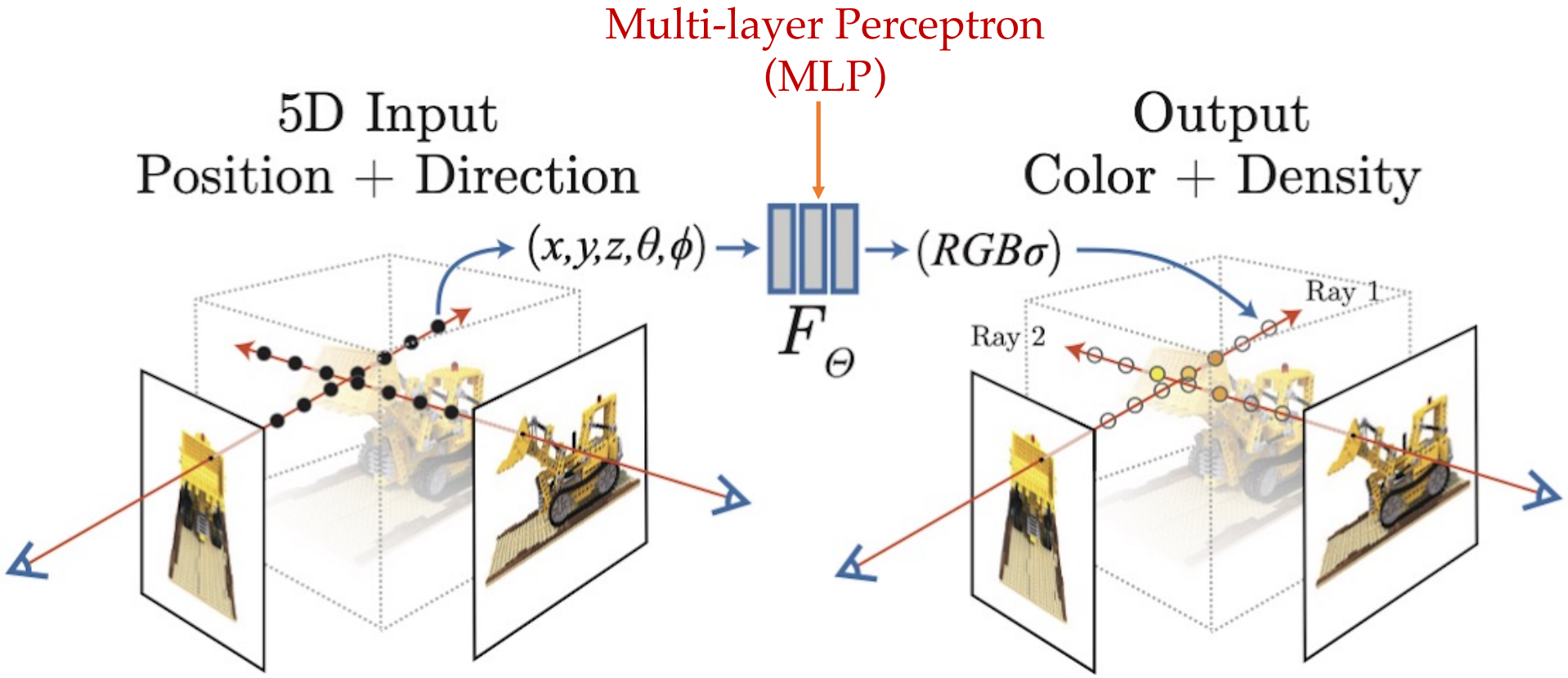
- **Advantages**
  - Learned end-to-end
  - Can handle partially-transparent and specular objects, and thin structures
  - Generalizable
  - Real-time rendering

- **Limitations**
  - Only allows small viewpoint changes
  - Memory expensive



Coordinate-based neural representations

- Neural Radiance fields (NeRF)



continuous volumetric scene representation

Neural Radiance fields (NeRF)

- Render images by volume rendering

Rendering model for ray r(t) = o + td:

$$C \approx \sum_{i=1}^{N} T_i \alpha_i c_i$$

colors

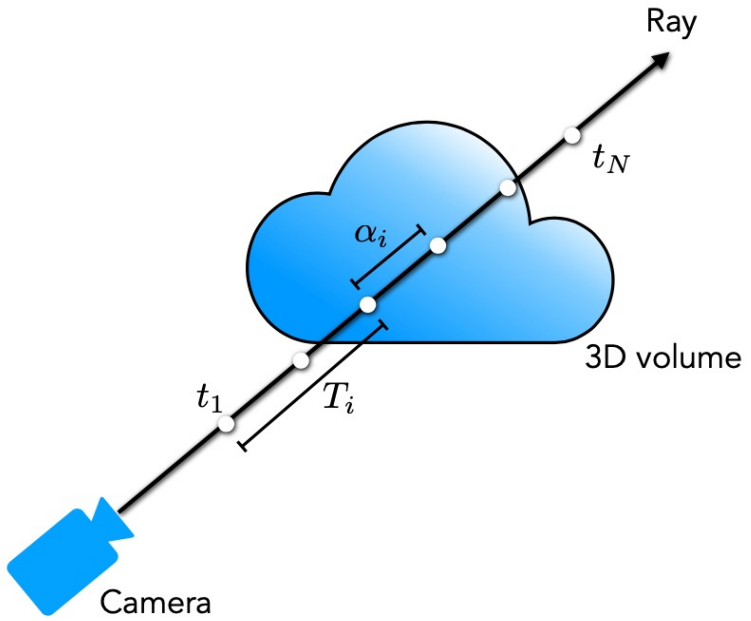
weights

How much light is blocked earlier along ray:

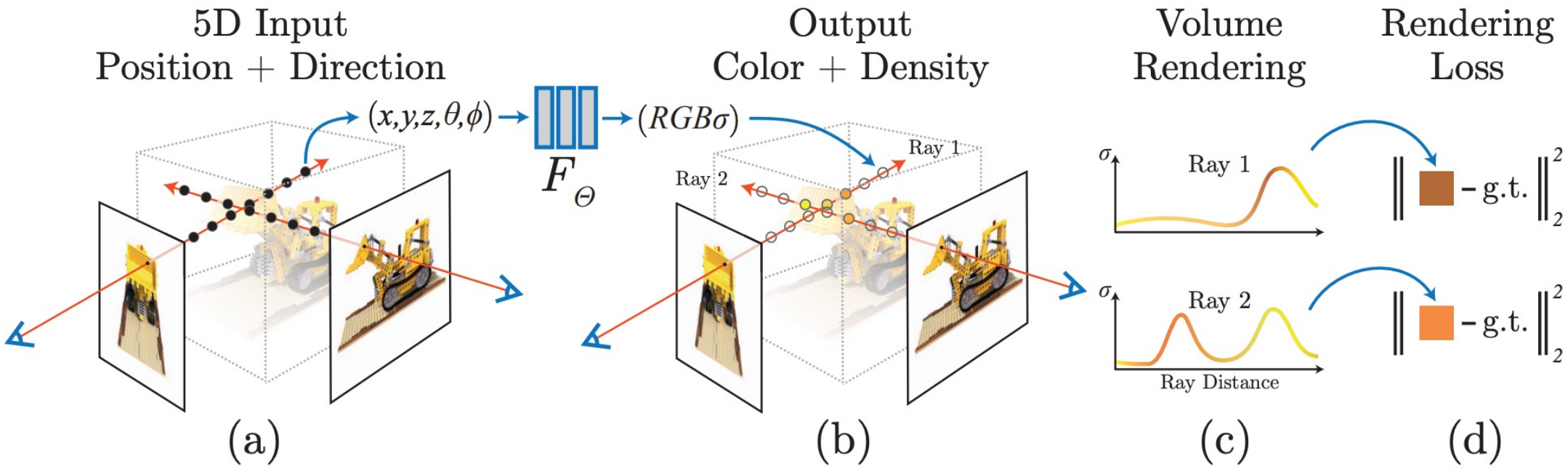
$$T_i = \prod_{j=1}^{i-1} (1 - \alpha_j)$$

How much light is contributed by ray segment *i*:

$$\alpha_i = 1 - e^{-\sigma_i \delta t_i}$$



Neural Radiance fields (NeRF)



Training using L2 Loss on image colors

Neural Radiance fields (NeRF)

Neural Radiance fields (NeRF)

- **Advantages**
  - Impressive view synthesis results (handles complex geometry and view dependent effects well)
  - Very compact (5MB vs. LLFF 15GB)
  - Strong multi-view consistency

- **Limitations**
  - Scene-specific, optimizing a NeRF for a scene needs ~1 day
  - Low rendering speed

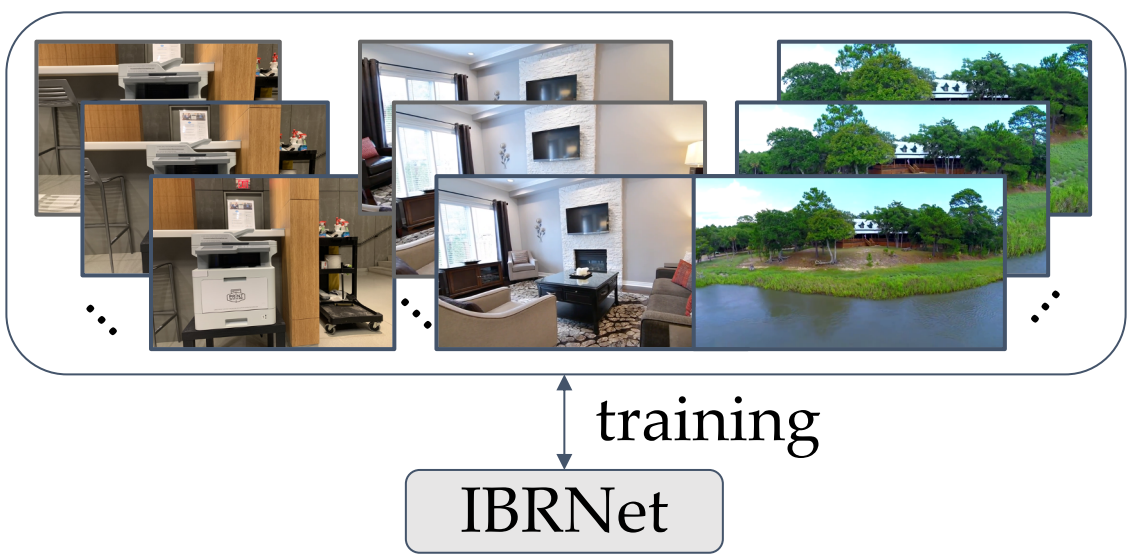
Mildenhall, Ben, et al. "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines." *TOG* 38.4 (2019): 1-14.

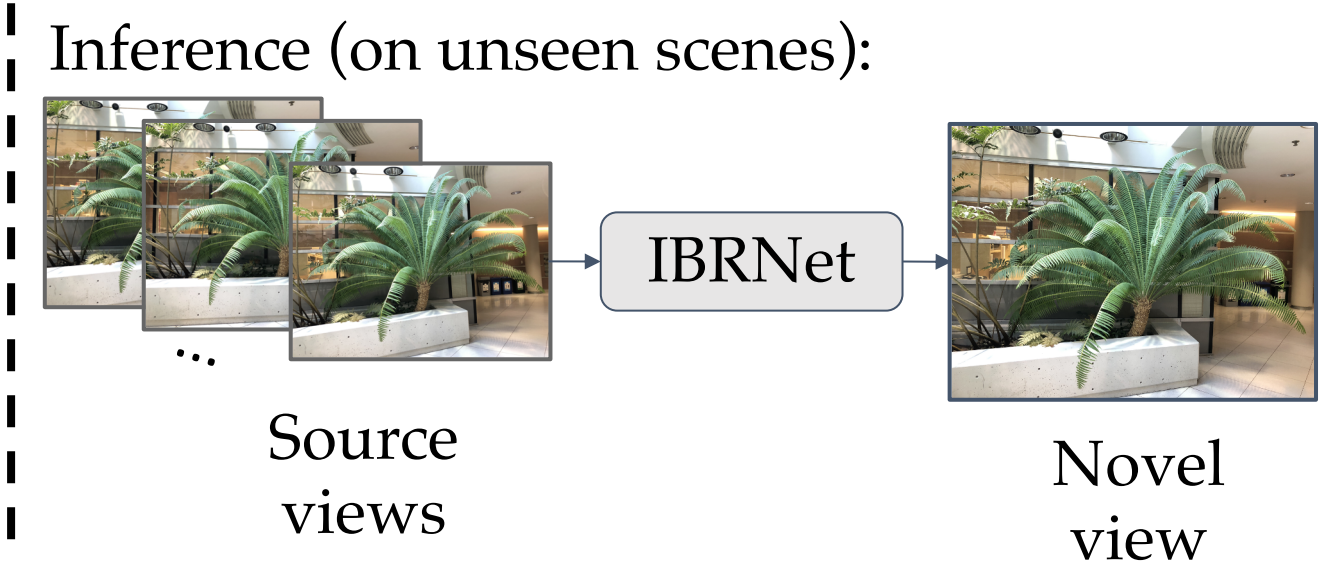
Improving & Extending NeRF

- Improving training / rendering speed
- Generalizing NeRF
- Extending NeRF to dynamic scenes
- Relighting with NeRF
- Generation with NeRF
- …

Our proposed method: IBRNet

IBRNet generates **continuous** scene radiance **on-the-fly** from source views for rendering novel views.





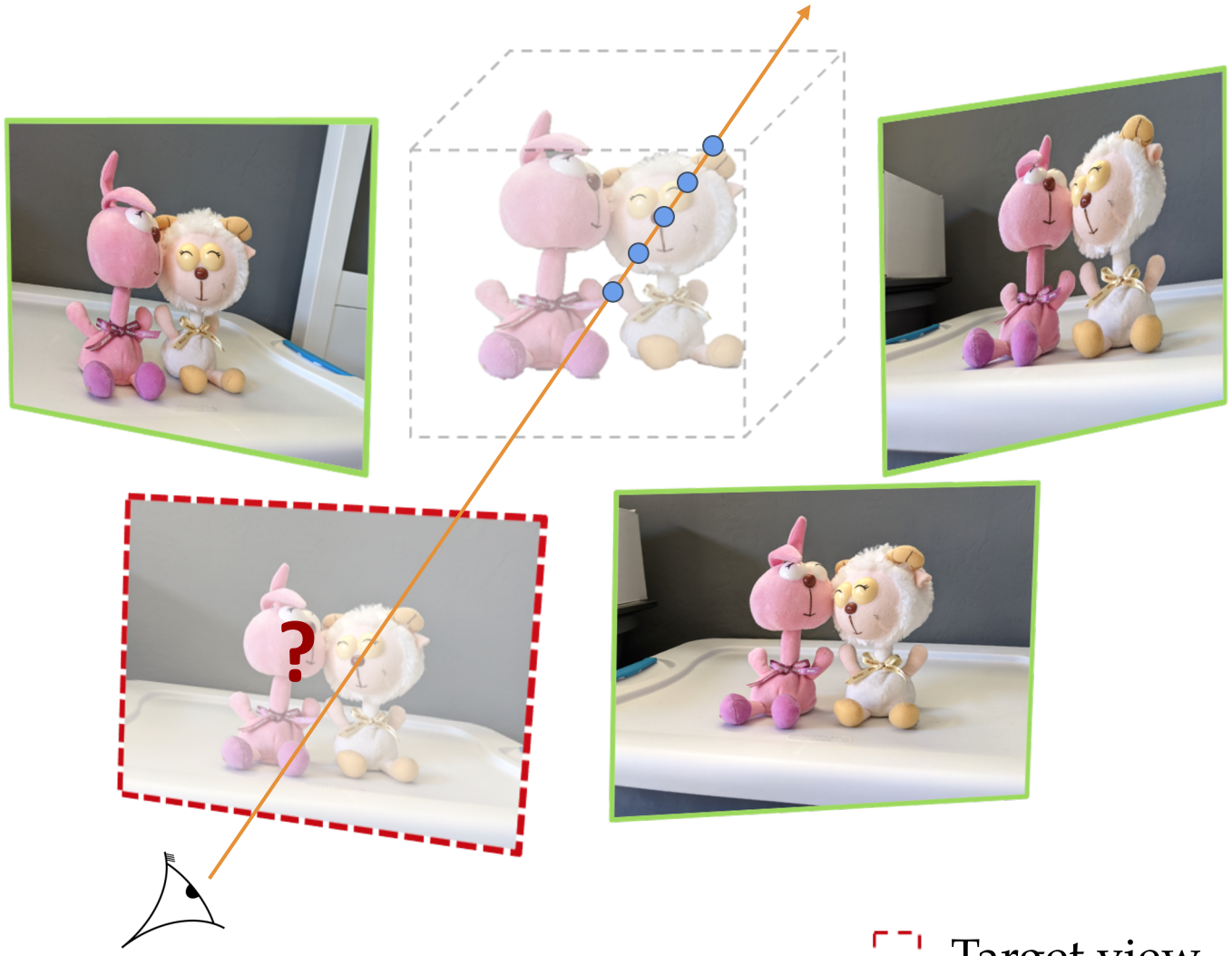
- Generalizes well to novel scenes
- Continuous representations that allow high-resolution rendering

Wang, Qianqian, et al. "Ibrnet: Learning multi-view image-based rendering." *CVPR*. 2021.

Our intuitions

- Instead of memorizing the scene, learn a general view interpolation function
- Replaces a trained per-scene MLP with on-the-fly multi-view stereo matching and image-based rendering

Pipeline Overview



To synthesize a target view:

Identify nearby source views & extract 2D features

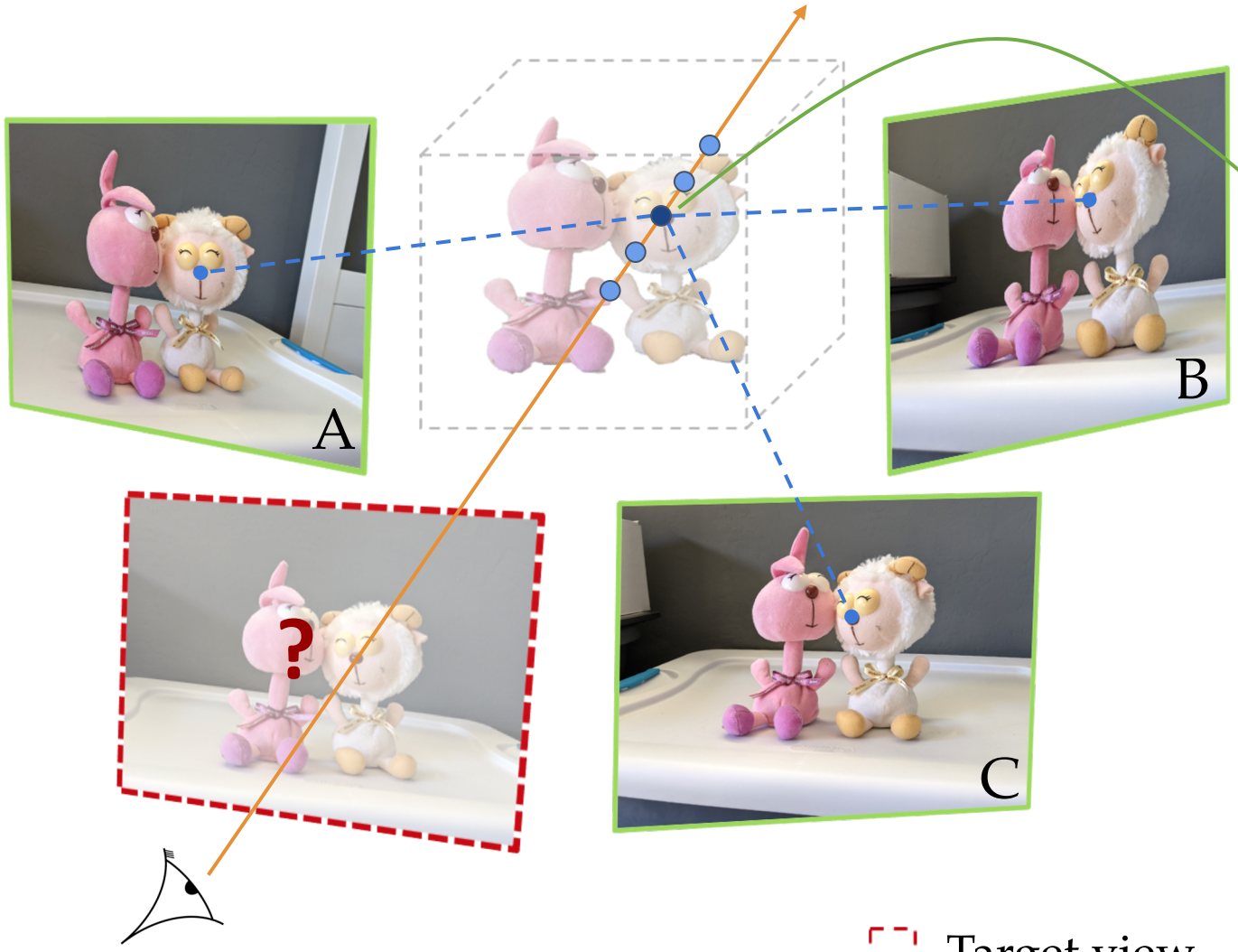
↓

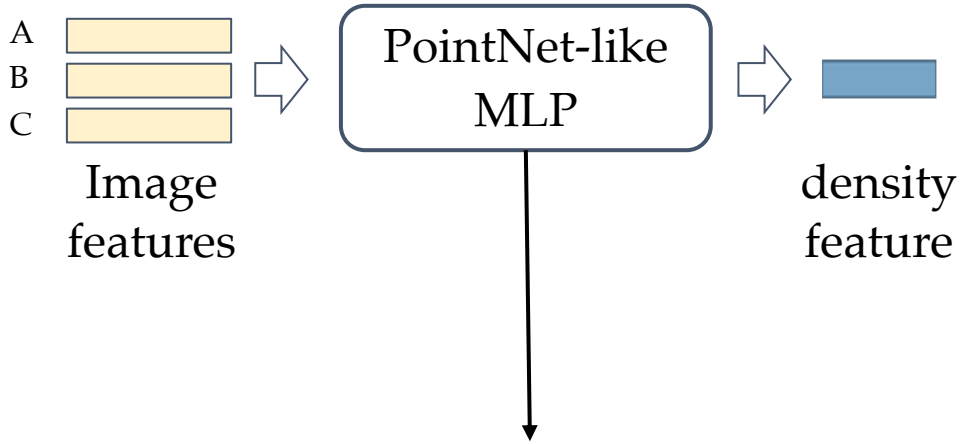
Produce volume densities and colors (**IBRNet**)

↓

Volume rendering

Volume density prediction

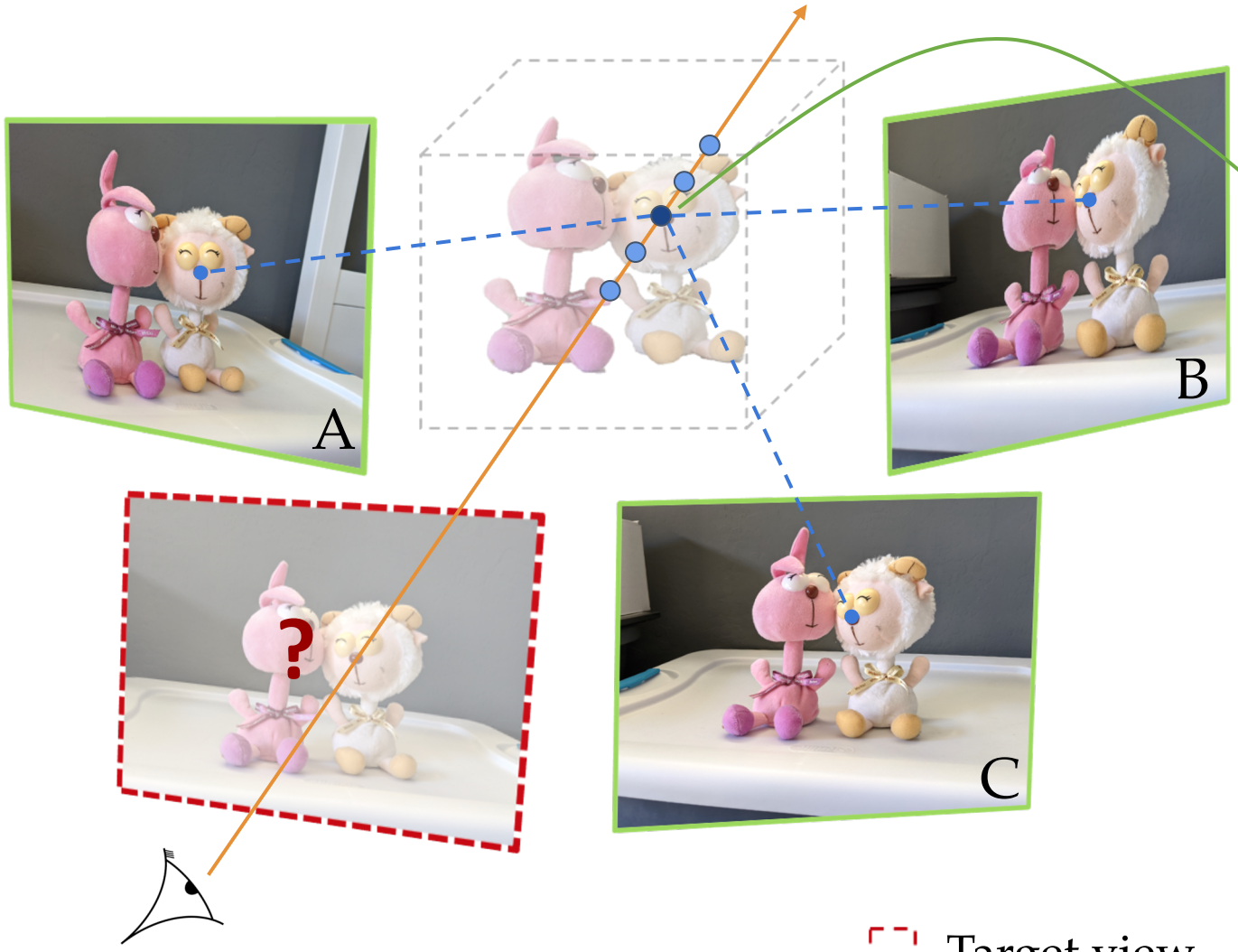


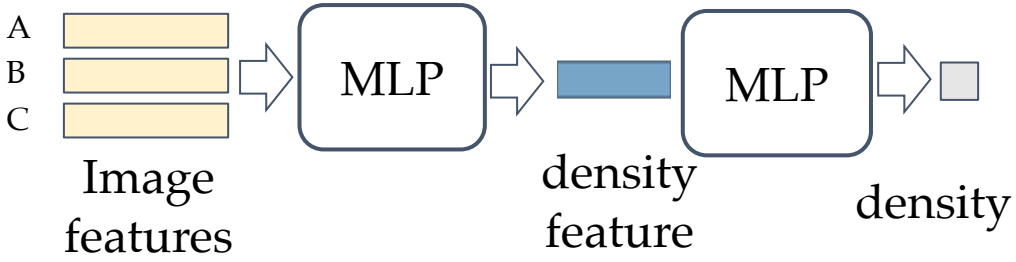


Checks photo consistency

On-surface<->high consistency<->high density
Free space <-> low consistency <-> low density

Volume density prediction





Doesn't work well!

Photo-consistency can be noisy and ambiguous

long-range contextual information?

Depth prediction in traditional MVS

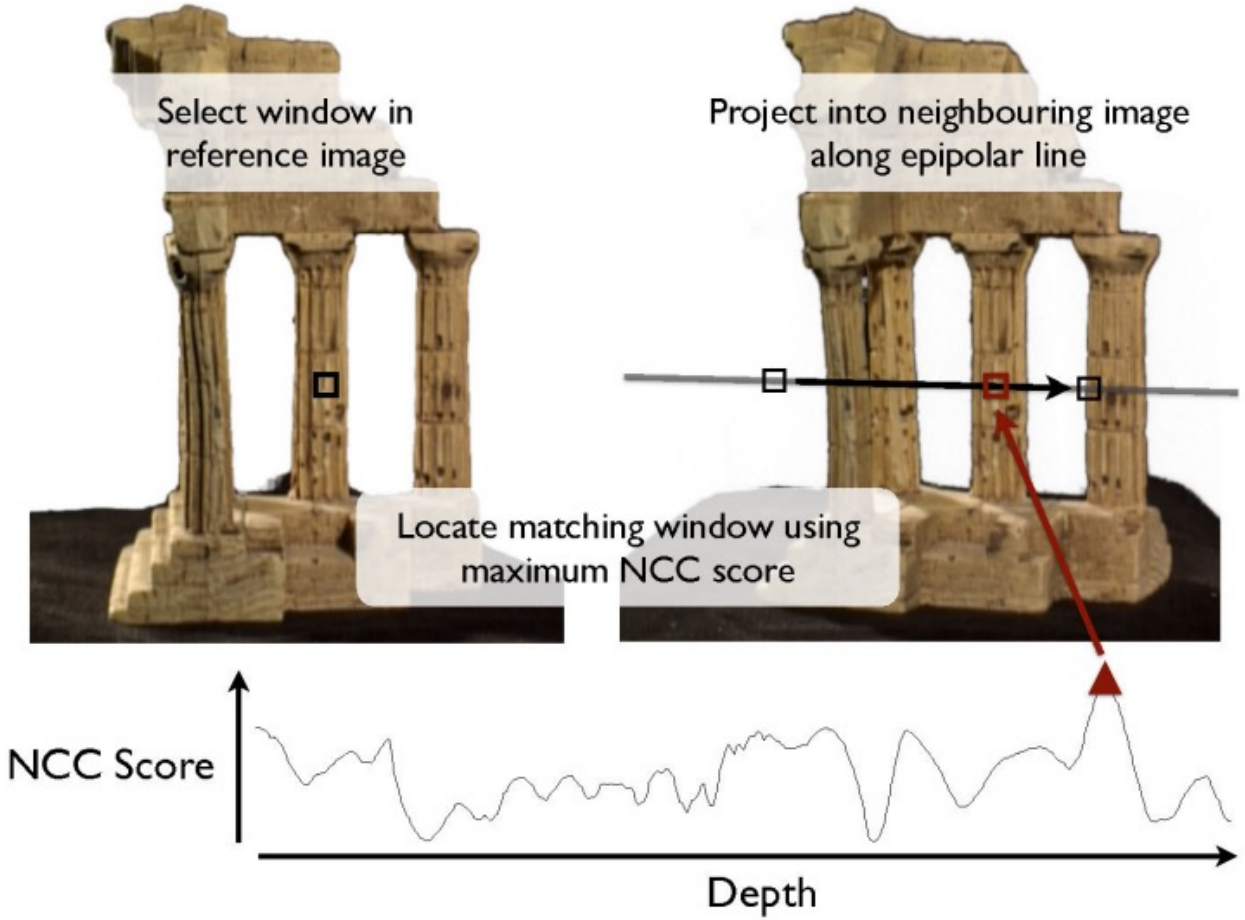
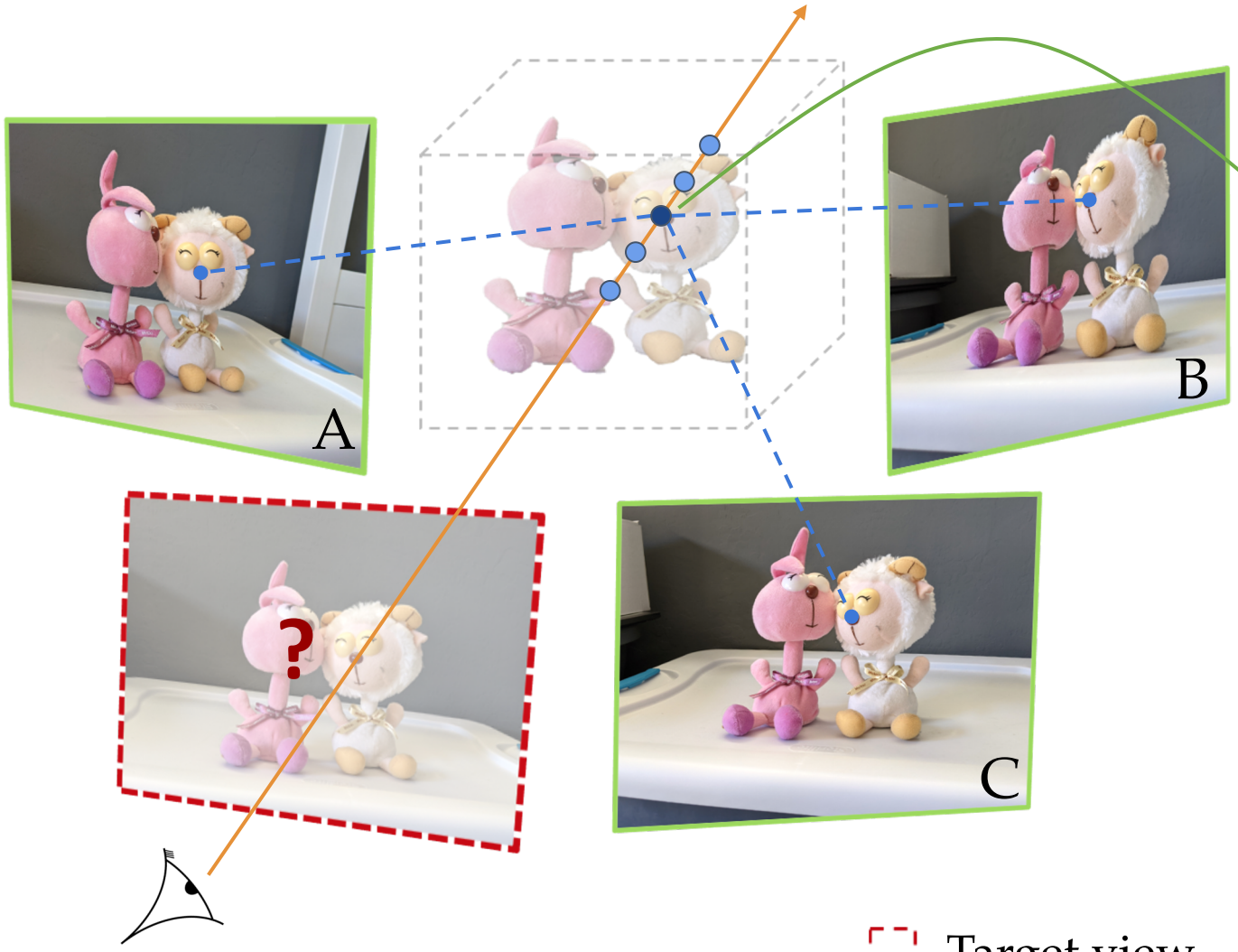
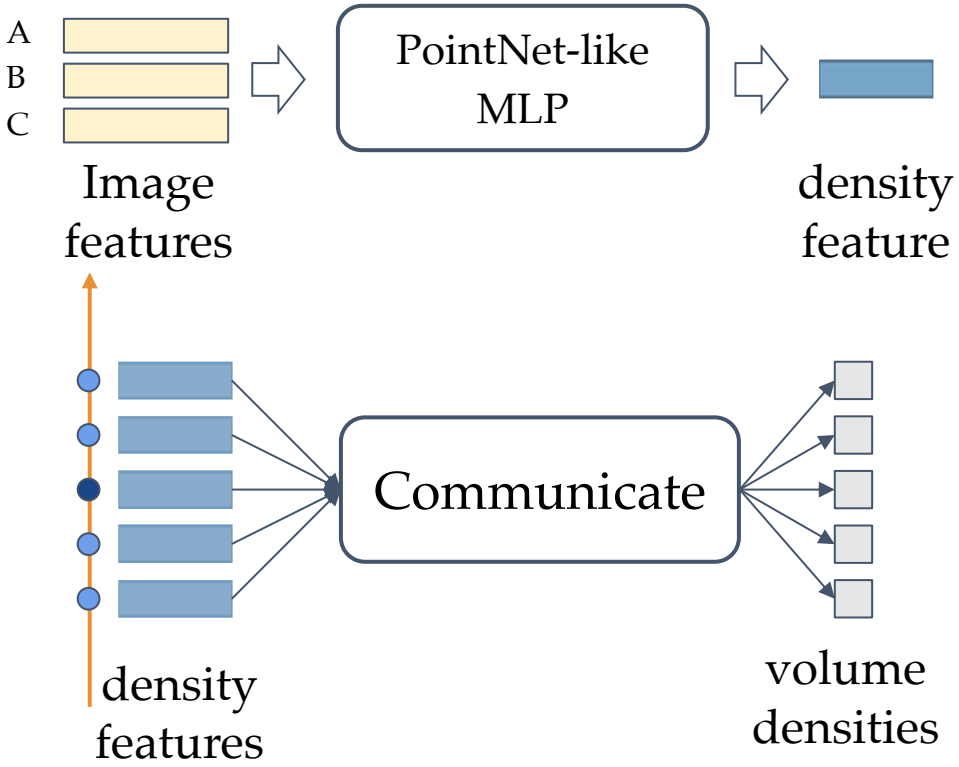


Figure 3.3: Winner-takes-all strategy for depthmap reconstruction. The figure illustrates a process to estimate a depth value for a pixel highlighted by a black rectangle in the left image. The global maximum of the photo-consistency function such as the NCC score is chosen to be the reconstructed depth for the pixel.

Volume density prediction



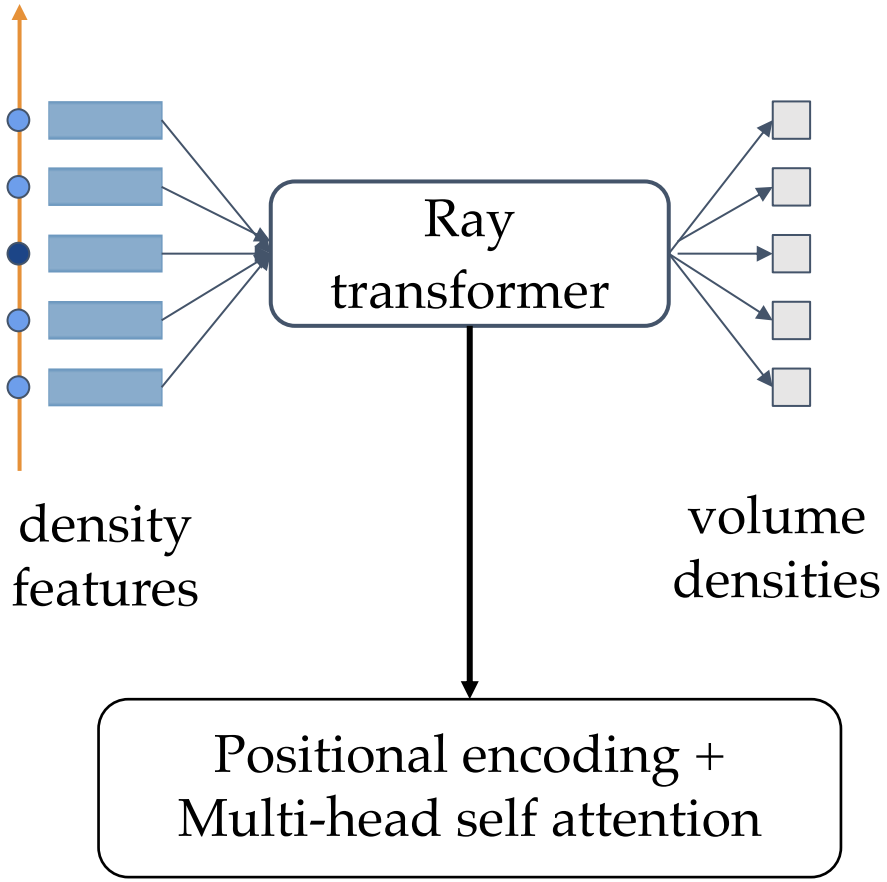


Target view

source view

Aggregate information along the entire ray

Ray Transformer



allows arbitrary number of input samples

Ray transformer comparison

w/o ray transformer

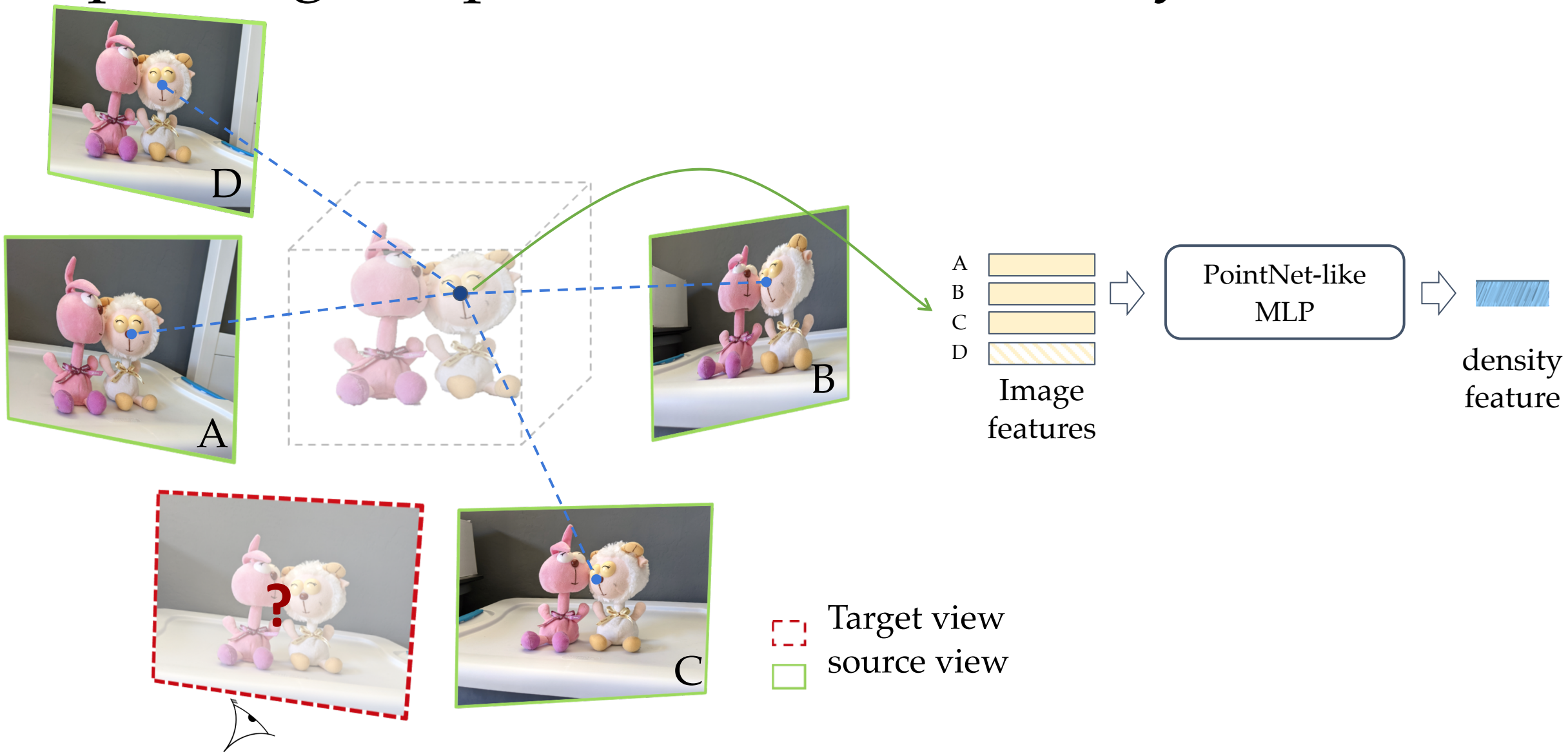
Mean PSNR: 21. 31

w/ ray transformer

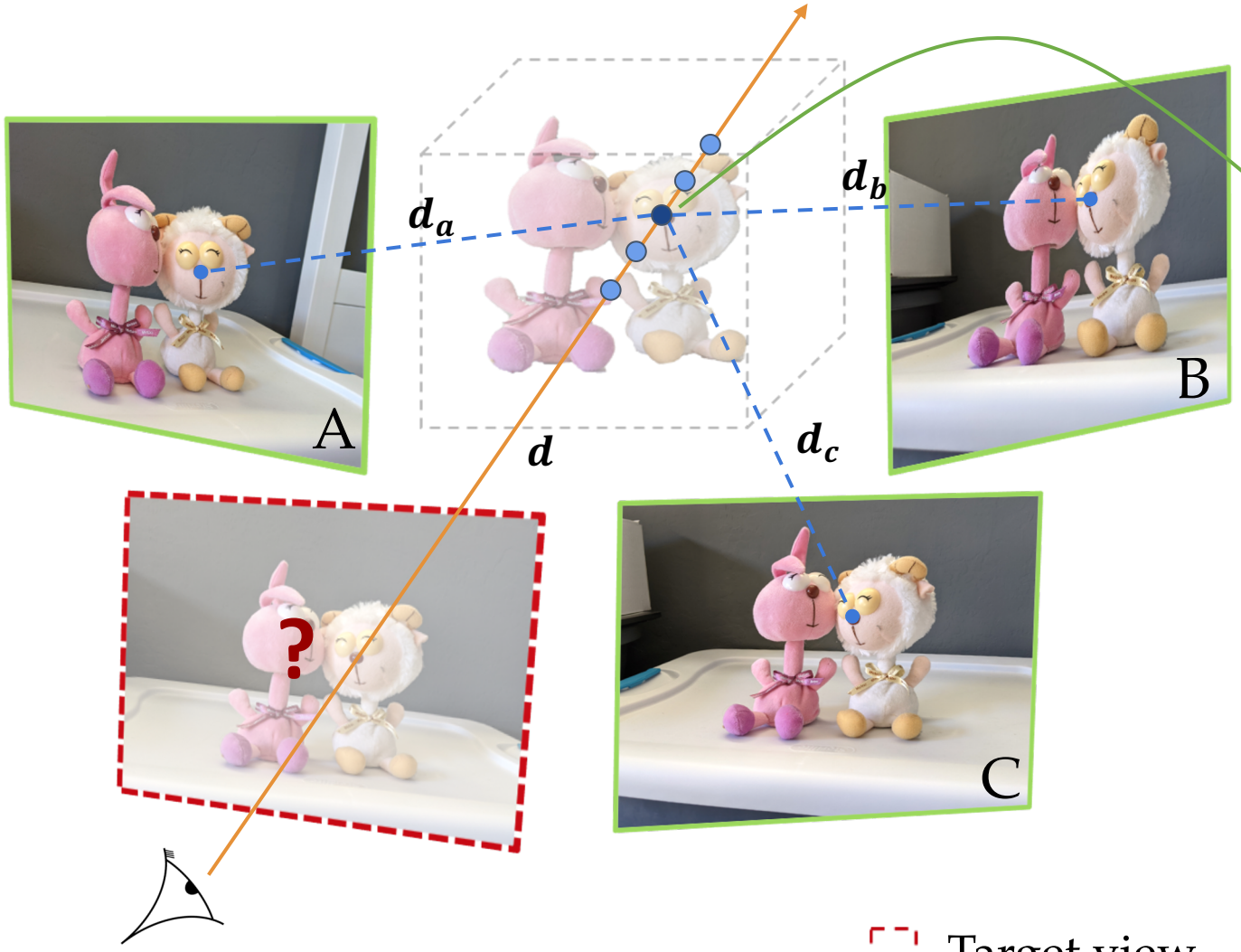
Mean PSNR: 25.13

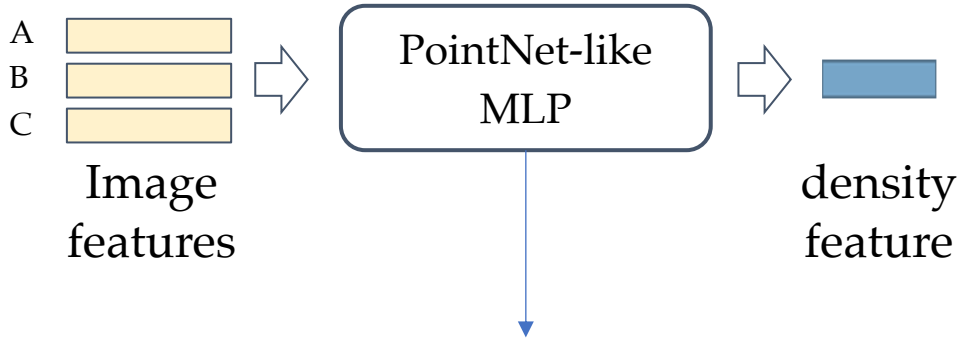
groundtruth

Improving temporal visual consistency



Improving temporal visual consistency

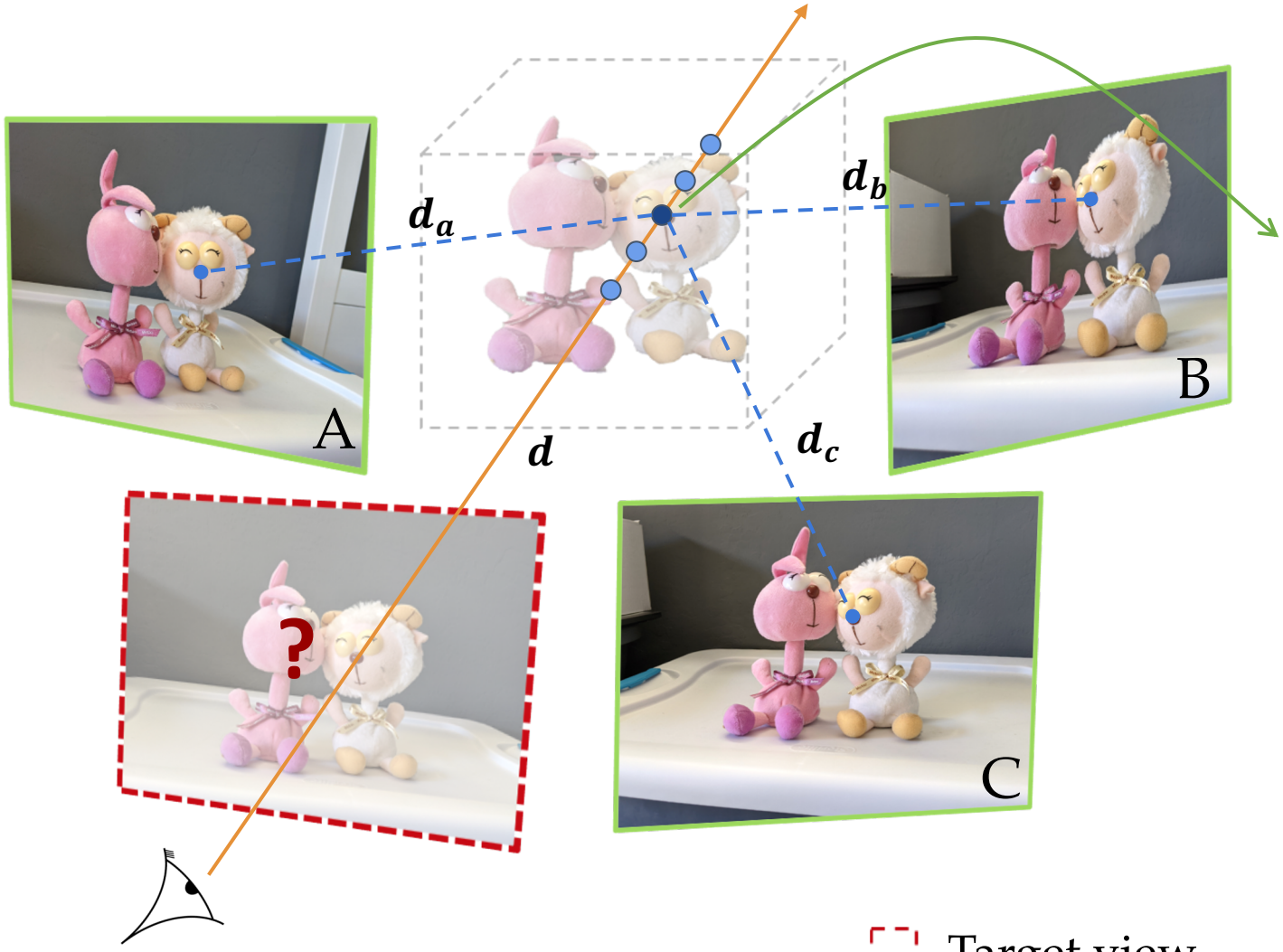


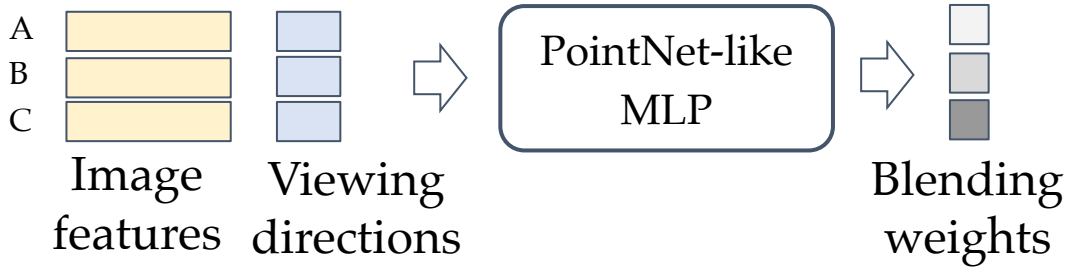


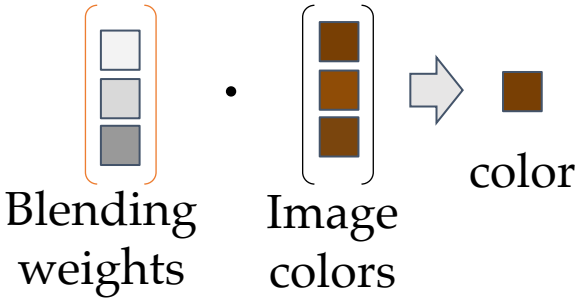
$$\tilde{w}_i^{\mathbf{f}}(\mathbf{d}, \mathbf{d}_i) = \max\left(0, e^{s(\mathbf{d}\cdot\mathbf{d}_i - 1)} - \min_{j=1\ldots N} e^{s(\mathbf{d}\cdot\mathbf{d}_j - 1)}\right)$$

$$w_i^{\mathbf{f}}(\mathbf{d}, \mathbf{d}_i) = \frac{\tilde{w}_i^{\mathbf{f}}(\mathbf{d}, \mathbf{d}_i)}{\sum_j \tilde{w}_j^{\mathbf{f}}(\mathbf{d}, \mathbf{d}_j)},$$

Color prediction

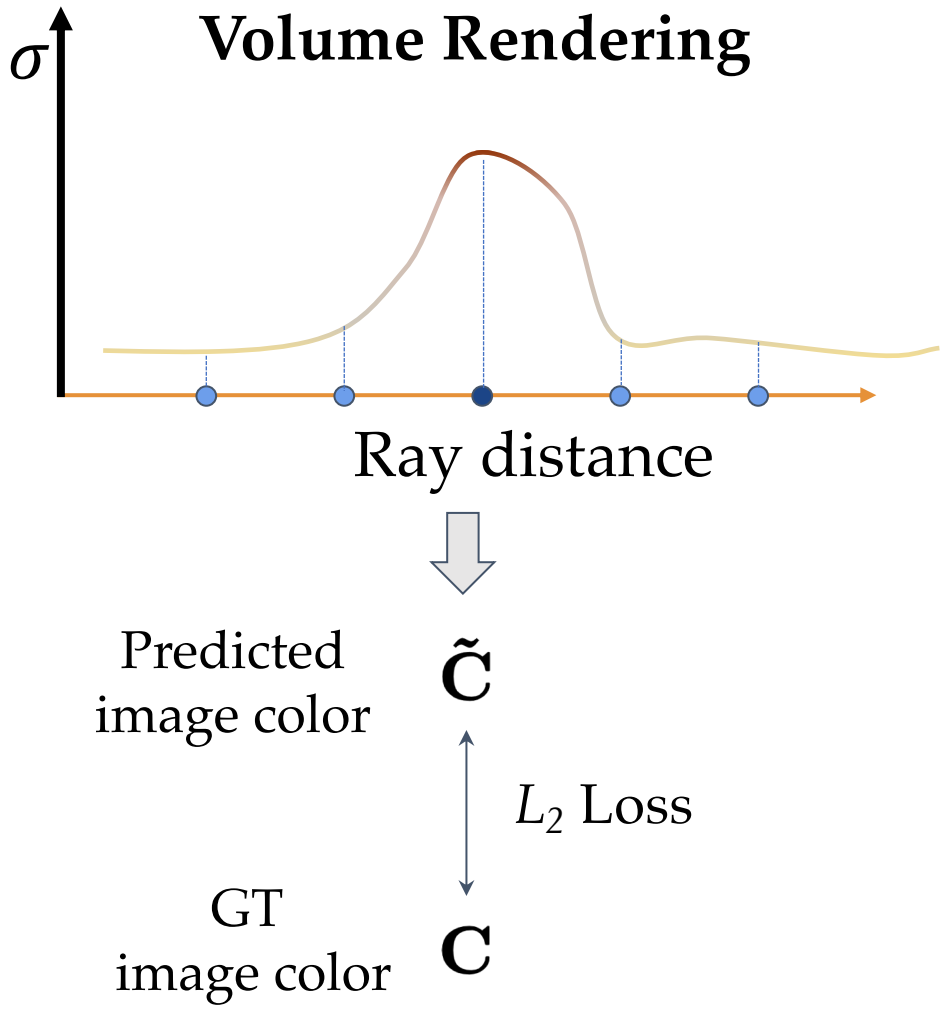


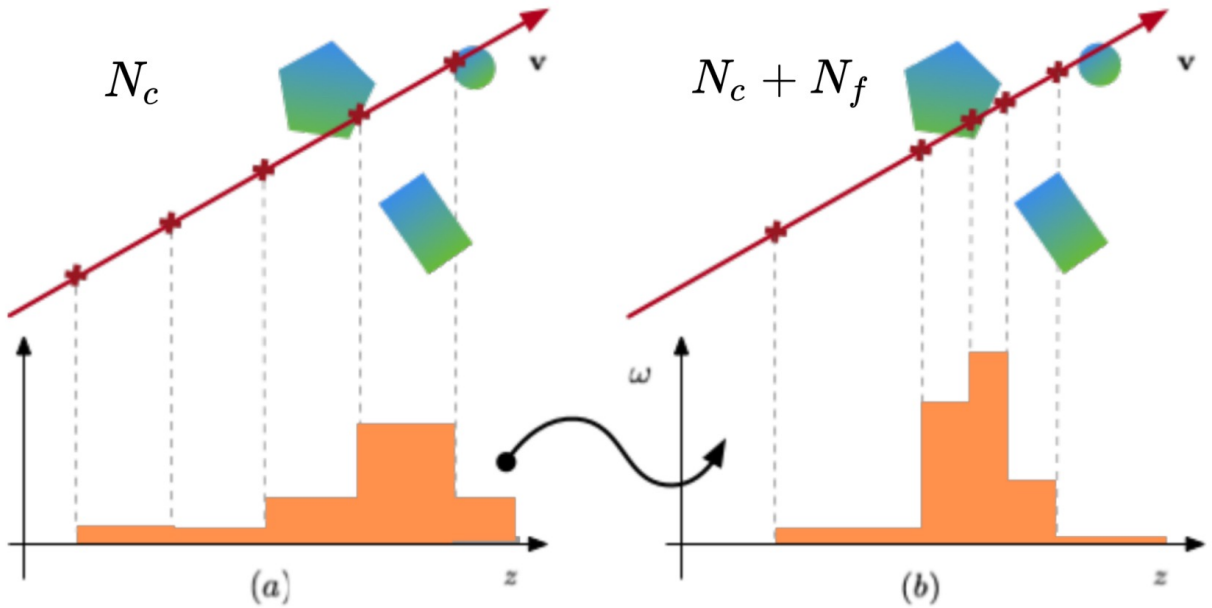




Blending image colors in source views

Volume Rendering and Training

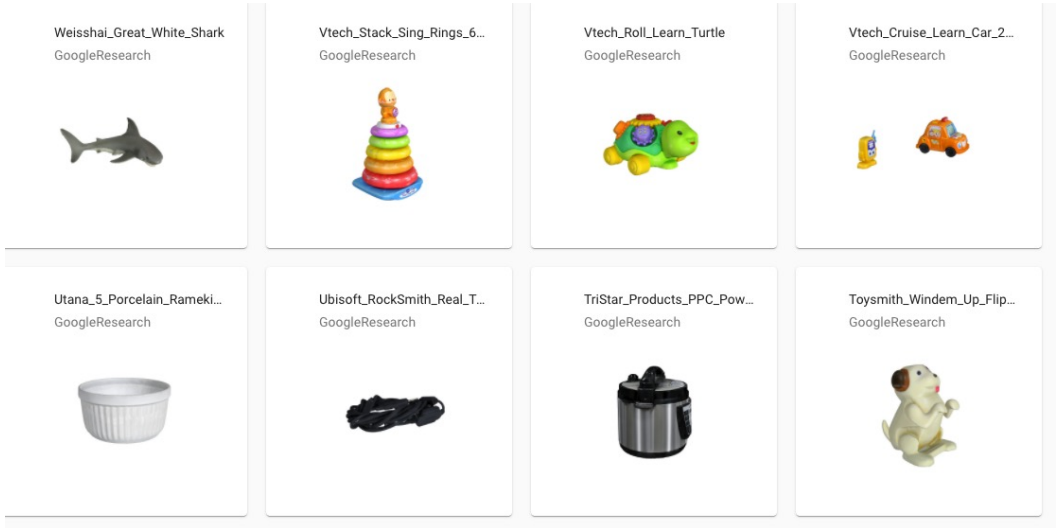




Hierarchical volume sampling

Training datasets

- Multi-view posed images



Google scanned objects

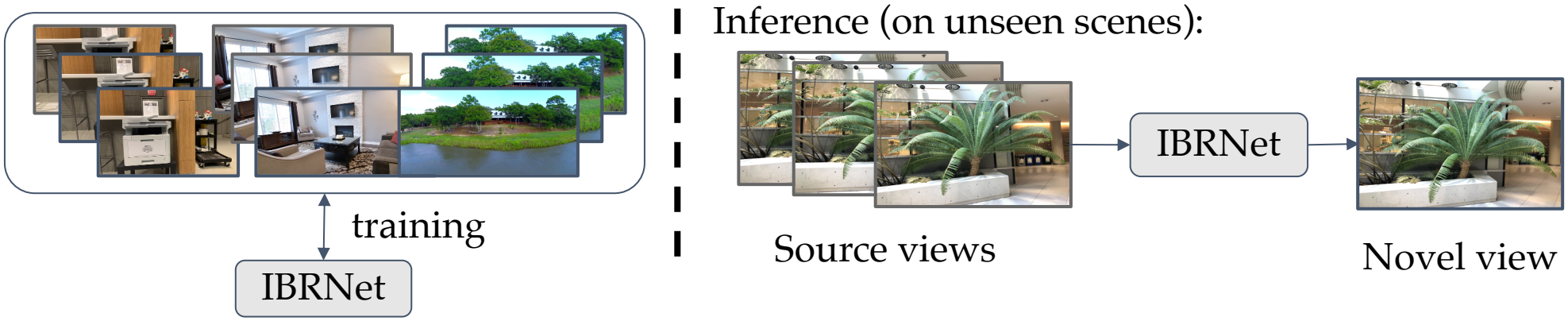
RealEstate10K

...

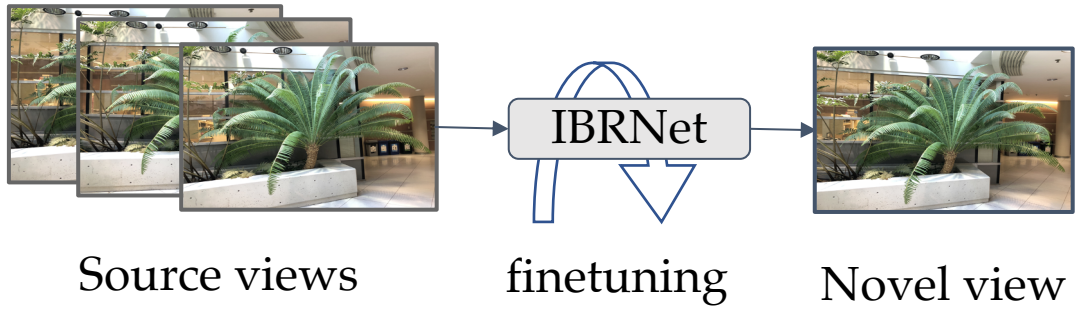
Forward facing scenes
(Collected by LLFF authors and ourselves)

Evaluation

- Directly apply the pretrained model (no per-scene optimization)

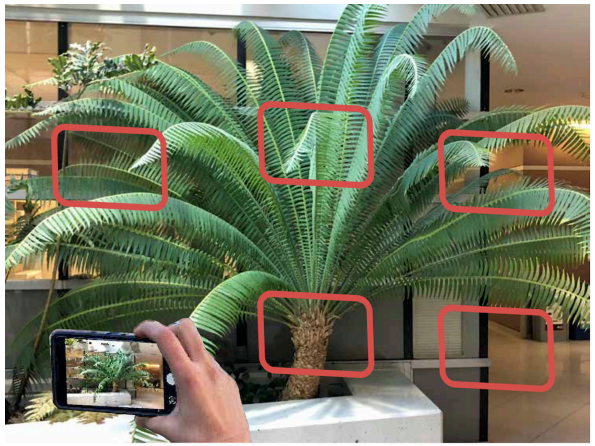


- Finetuning (per-scene optimization)

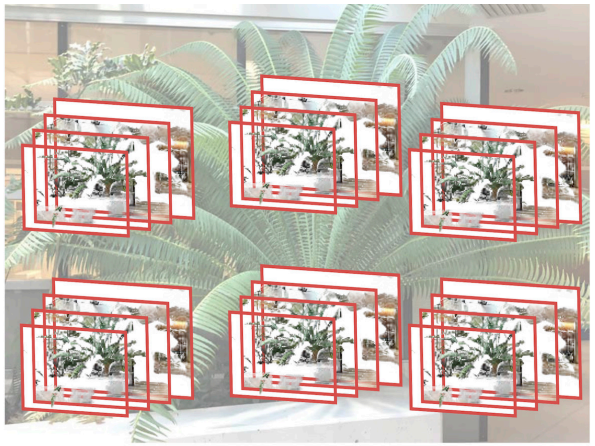


Baselines

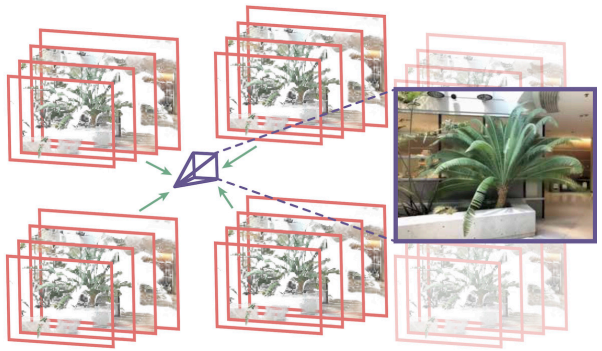
- No per-scene optimization: LLFF



Fast and easy handheld capture with guideline:
closest object moves at most D pixels between views



Promote sampled views to local light field
via layered scene representation



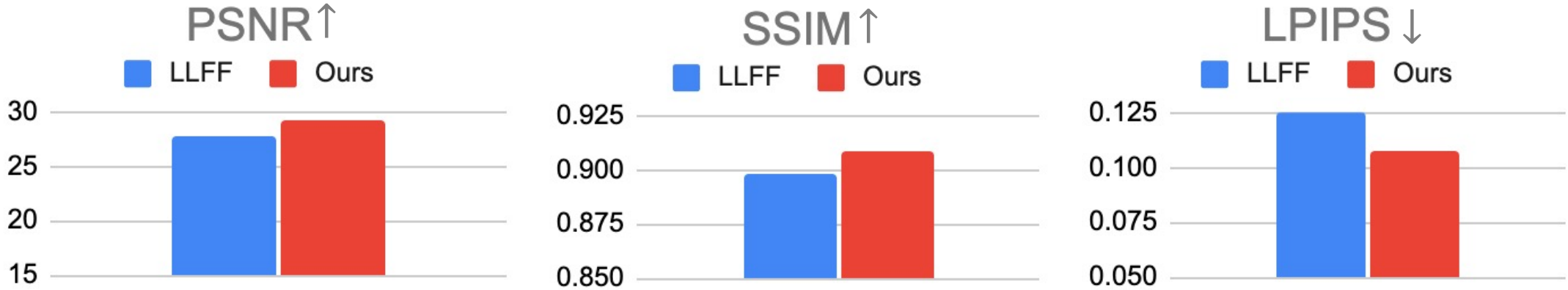
Blend neighboring local light fields
to render novel views

- Per-scene optimization: SRN, Neural Volumes, NeRF

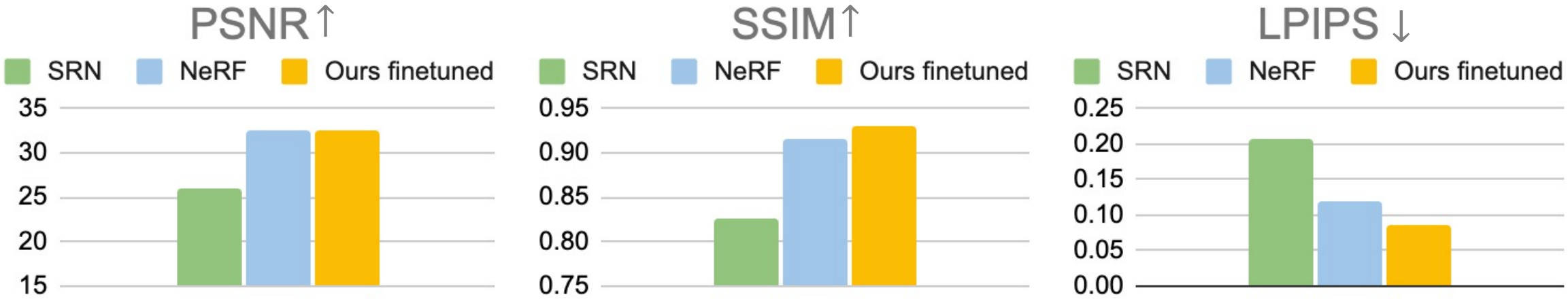
Mildenhall, Ben, et al. "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines." *TOG* 38.4 (2019): 1-14.

Evaluation

- No per-scene optimization: Ours outperforms prior state-of-the-art method LLFF



- Per-scene optimization: Ours finetuned is competitive to NeRF



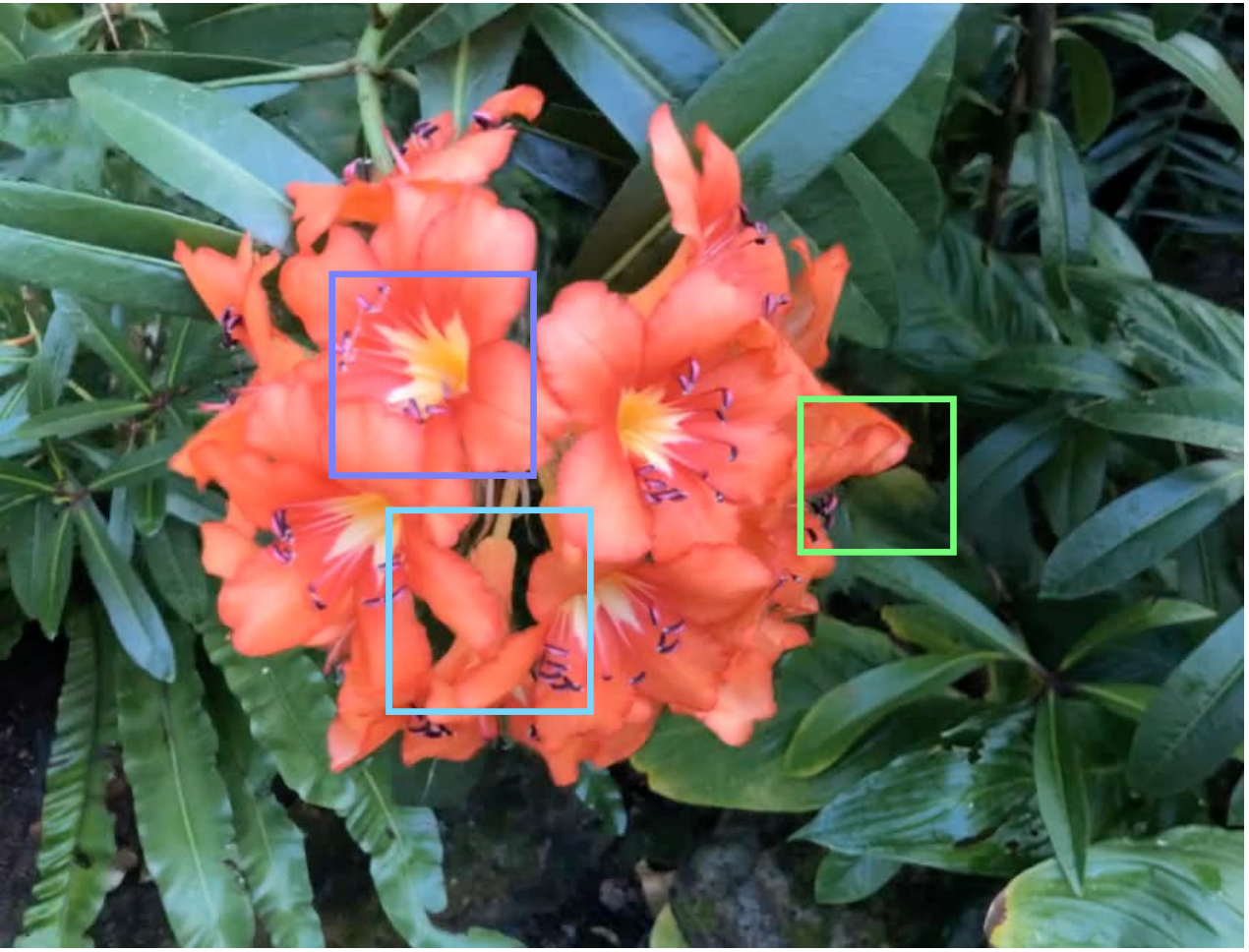
Mildenhall, Ben, et al. "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines." *TOG* 38.4 (2019): 1-14.

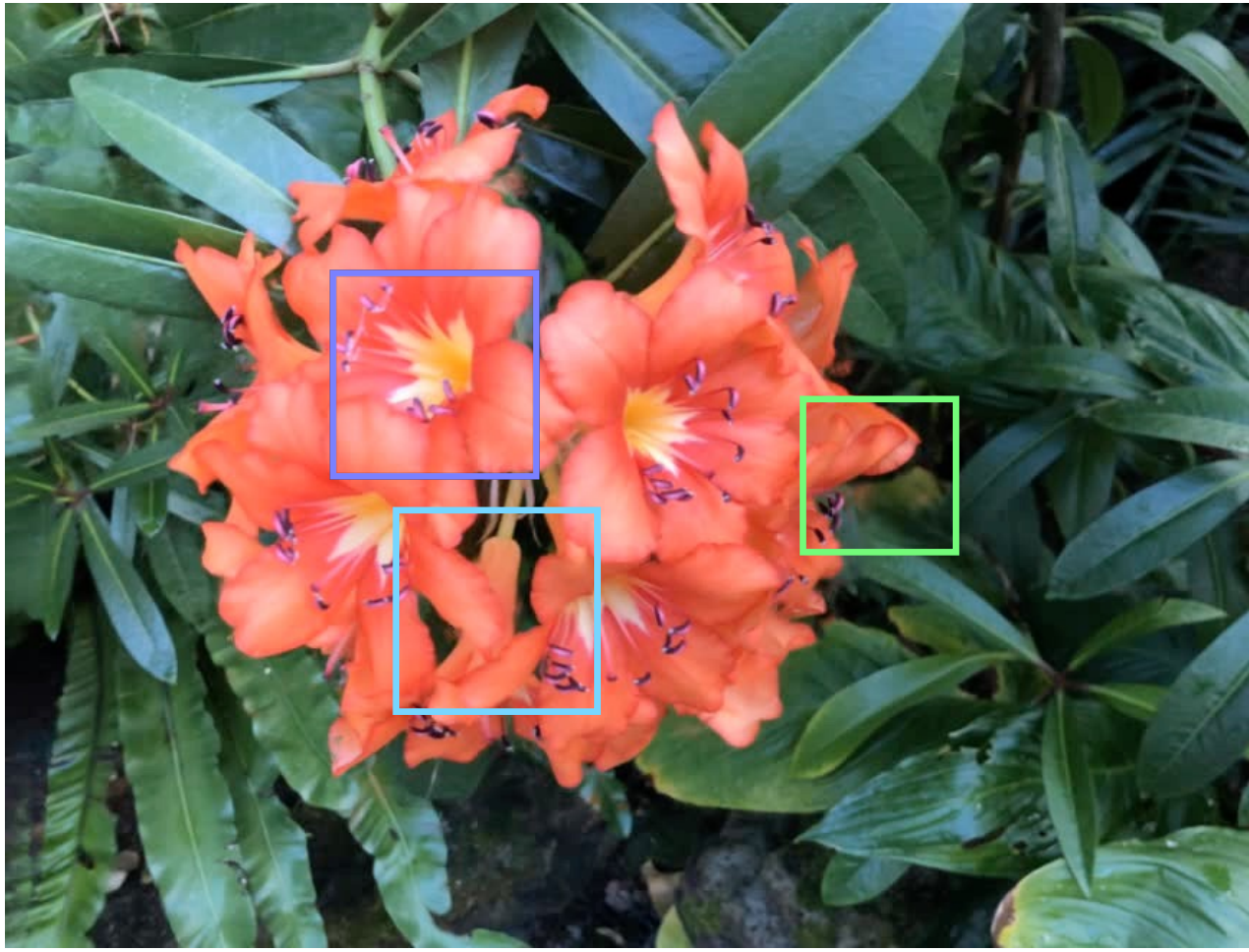
Video Comparison of **LLFF** and **Ours**


LLFF
Ours

LLFF

Ours

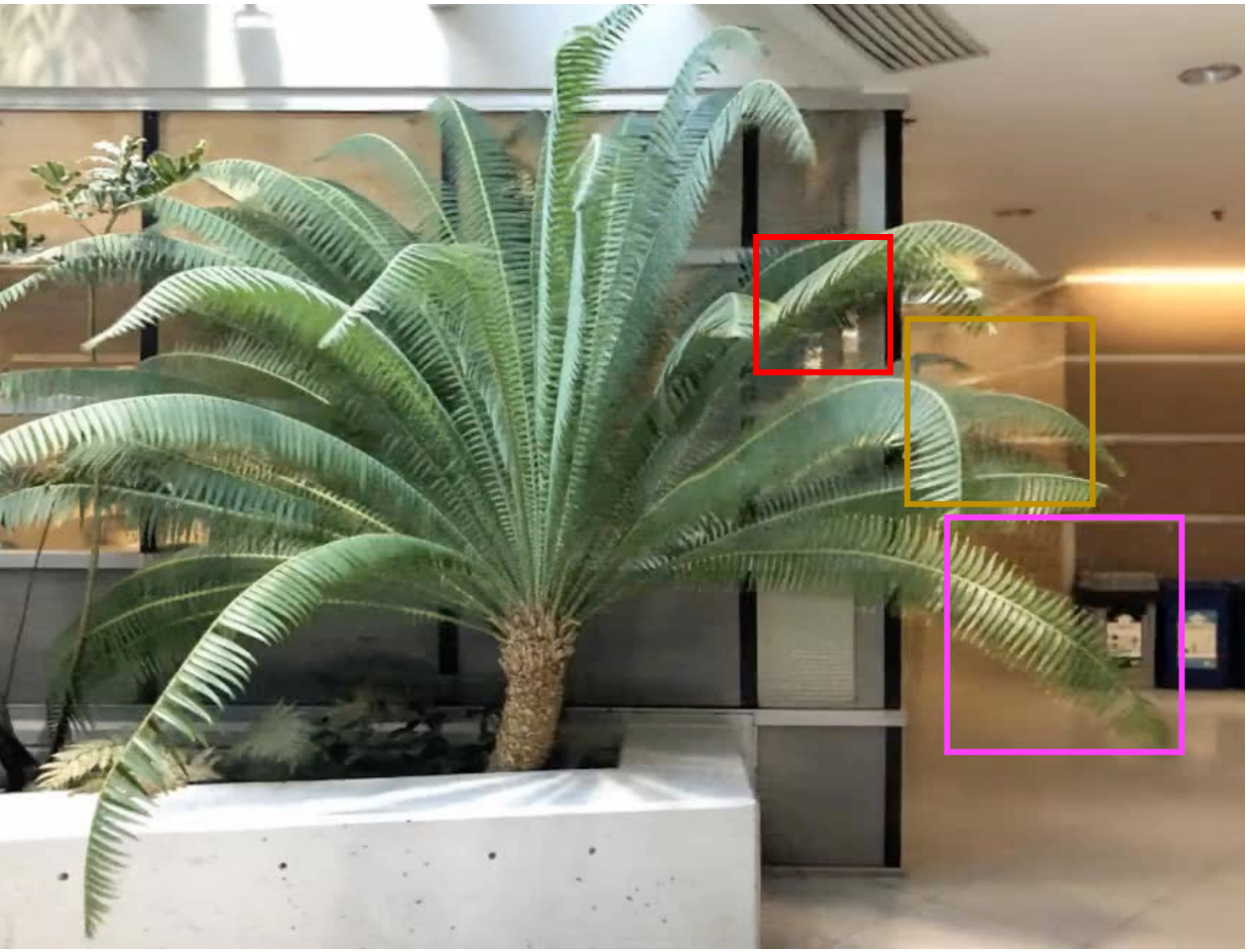


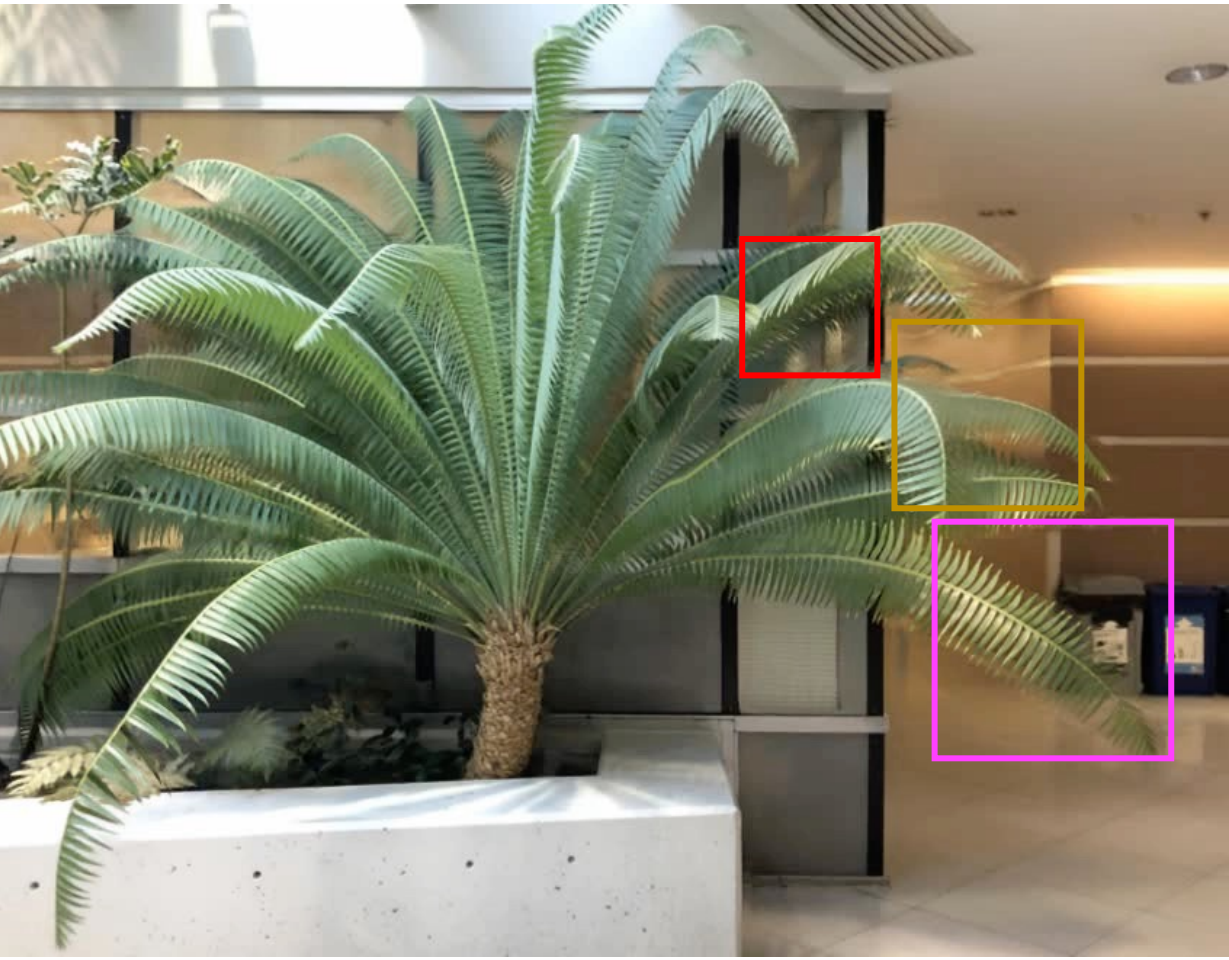


Video Comparison of **NeRF** and **Ours Finetuned**

NeRF

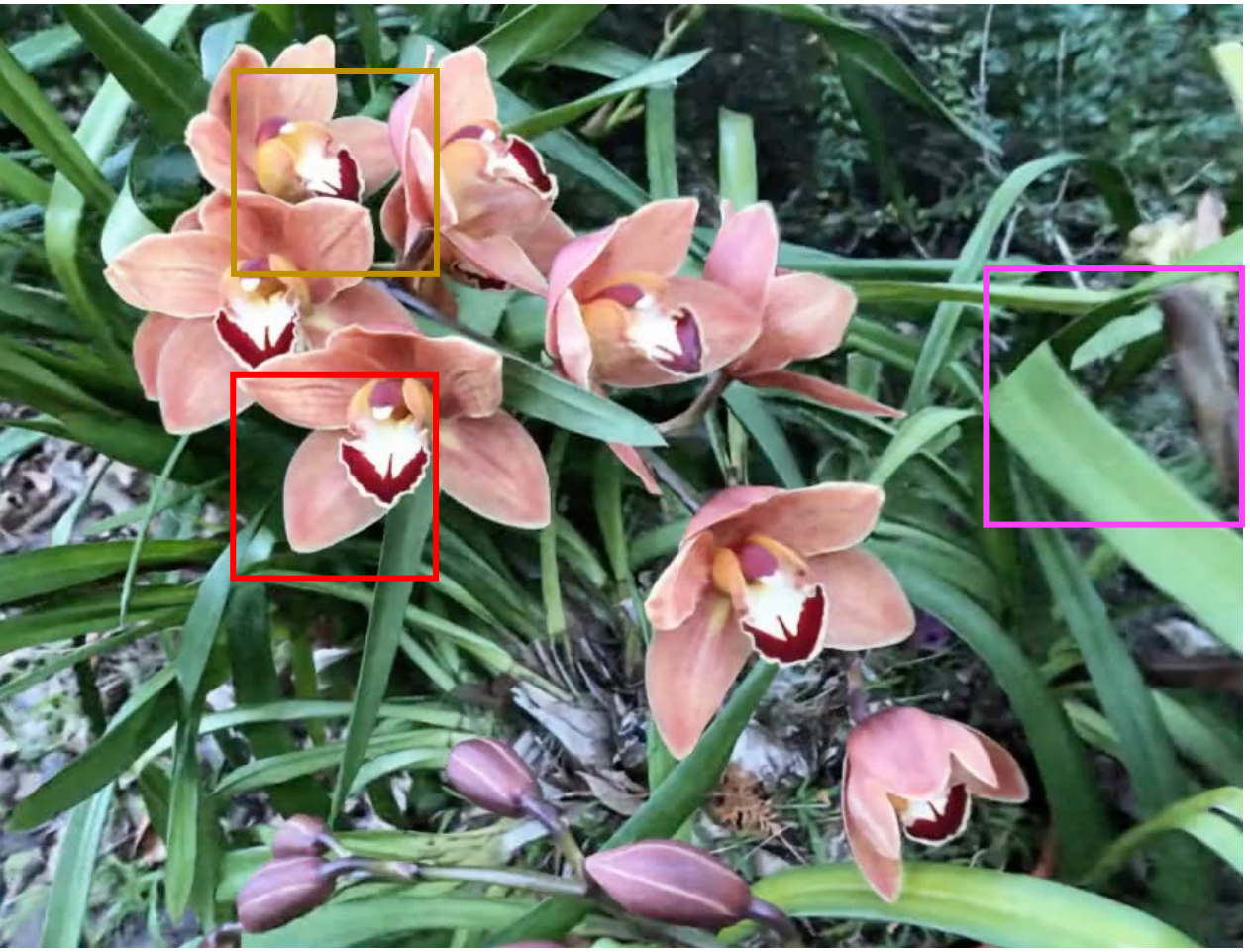
Ours Finetuned





NeRF

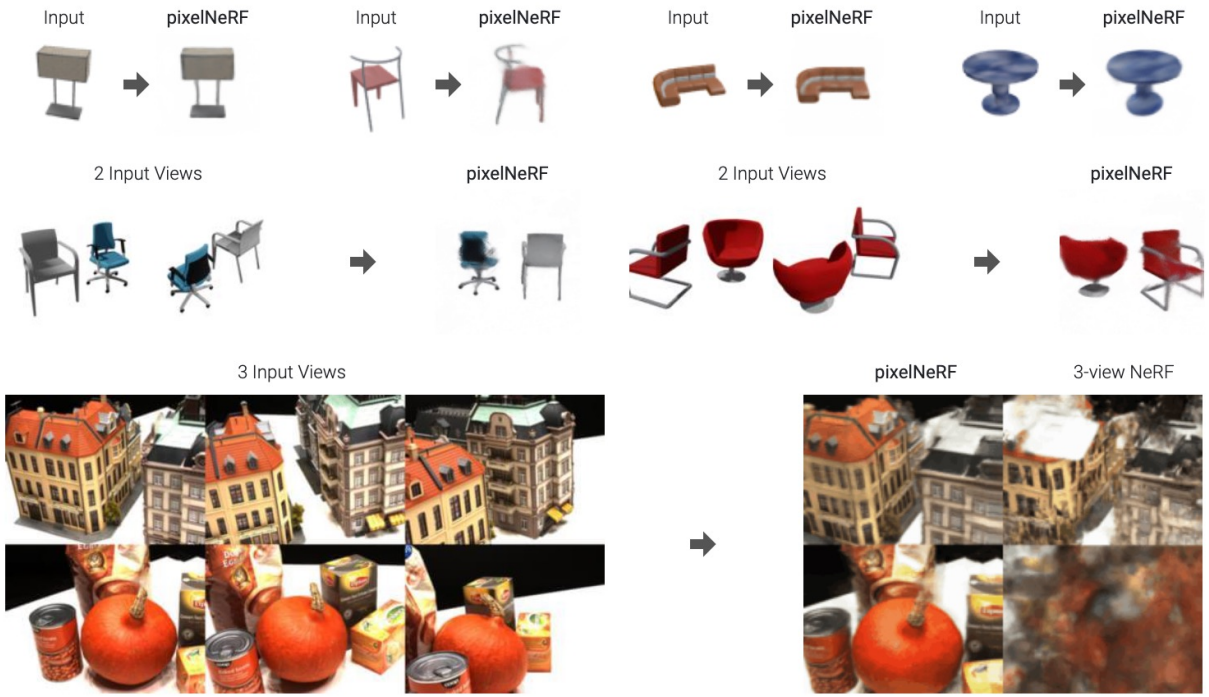
Ours Finetuned



IBRNet Summary & Analysis

- Renders target views by generating colors and densities in the space on the fly using nearby source views
- **Compared to NeRF:**
  - Generalizes to novel scenes (no per scene optimization required)
  - More scalable (local working set)
  - View-dependent geometry (vs. NeRF view independent geometry)
- **Compared to LLFF:**
  - Allows continuous sampling in the scene space
  - Better multi-view consistency
  - Low rendering speed

Comparison with concurrent work



PixelNeRF

1. Still uses coordinate-based networks
2. Uses category-level (object-level) priors (does not generalize to arbitrary scenes)
3. Focus on sparse views

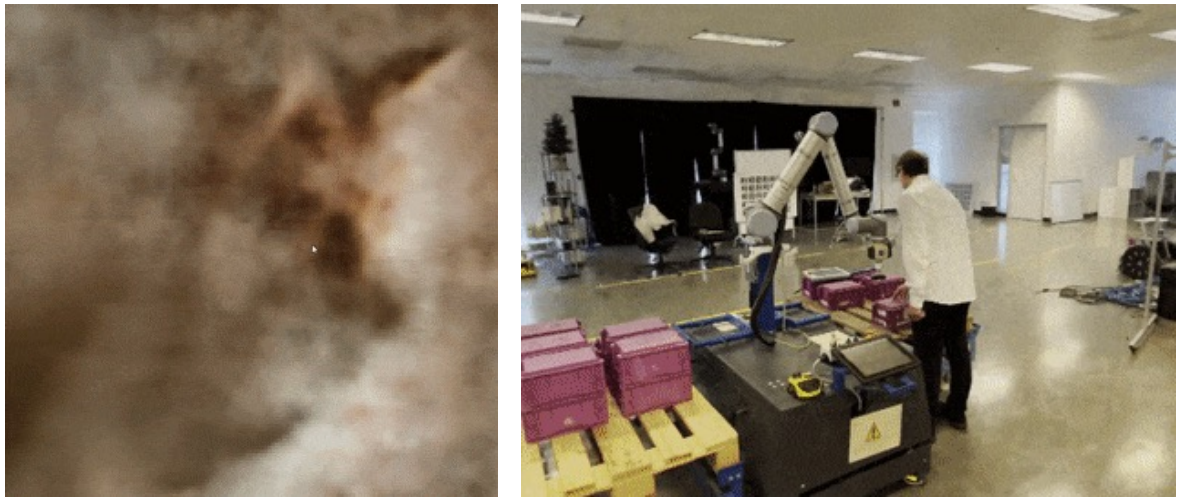
Yu, Alex, et al. "pixelnerf: Neural radiance fields from one or few images." *CVPR*. 2021.

Trevithick, Alex, and Bo Yang. "Grf: Learning a general radiance field for 3d representation and rendering." *ICCV*. 2021.

Jang, Wonbong, and Lourdes Agapito. "Codenerf: Disentangled neural radiance fields for object categories."*ICCV*. 2021.

Future directions

- Improving rendering speed
- Extending to dynamic scenes



Instant Neural Graphics Primitives: training NeRF in 5 seconds!

Müller, Thomas, et al. “Instant Neural Graphics Primitives with a Multiresolution Hash Encoding”

Thank you!