Progressively Optimized Local Radiance Fields for Robust View Synthesis

Andreas Meuleman[1‡] Yu-Lun Liu[2‡] Chen Gao[3]
Jia-Bin Huang[3,4] Changil Kim[3] Min H. Kim[1] Johannes Kopf[3]
[1]KAIST [2]National Taiwan University [3]Meta [4]University of Maryland, College Park
https://localrf.github.io/

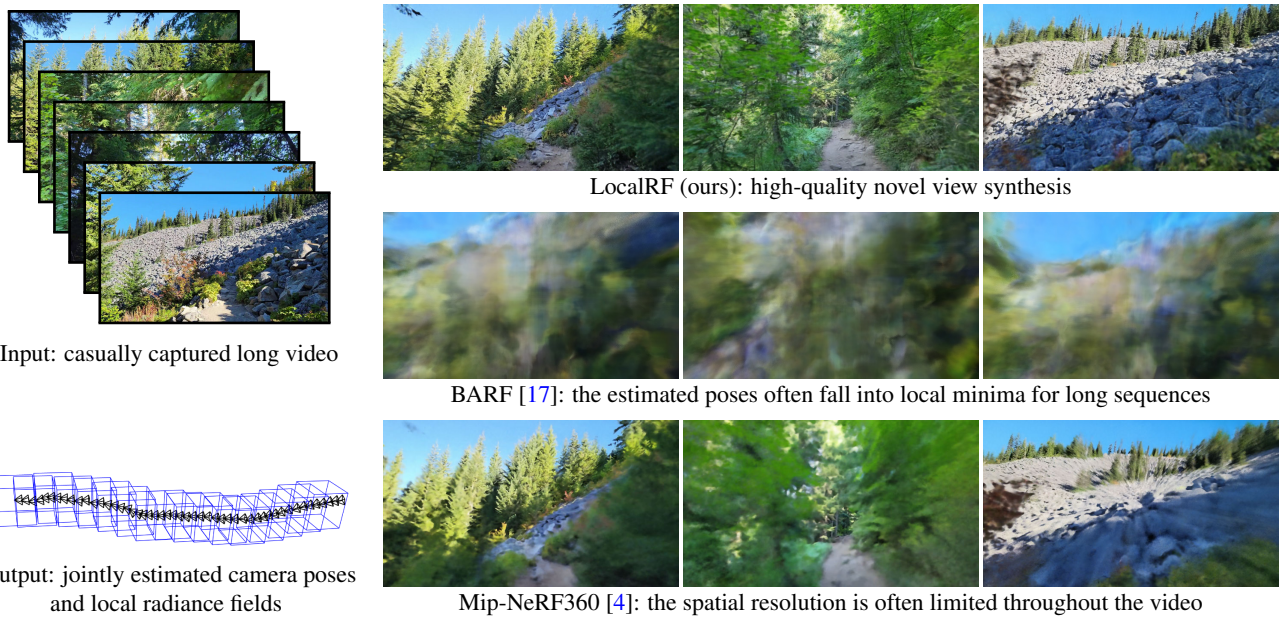


Figure 1. **High-quality novel view synthesis from a long casually captured video.** We jointly optimize camera poses and a scene representation using a progressive scheme that dynamically allocates local radiance fields (blue boxes). Our method robustly handles casual hand-held captures, scales to processing arbitrarily long videos with limited memory usage, and maintains high resolution throughout the entire video.

Abstract

We present an algorithm for reconstructing the radiance field of a large-scale scene from a single casually captured video. The task poses two core challenges. First, most existing radiance field reconstruction approaches rely on accurate pre-estimated camera poses from Structure-from-Motion algorithms, which frequently fail on in-the-wild videos. Second, using a single, global radiance field with finite representational capacity does not scale to longer trajectories in an unbounded scene. For handling unknown poses, we jointly estimate the camera poses with radiance field in a progressive *manner. We show that progressive optimization significantly improves the robustness of the reconstruction. For handling large unbounded scenes, we dynamically allocate new* local *radiance fields trained with frames within a temporal window. This further improves robustness (e.g., performs well even under moderate pose drifts) and allows us to scale to large scenes. Our extensive evaluation on the* TANKS AND TEMPLES *dataset and our collected outdoor dataset,* STATIC HIKES, *show that our approach compares favorably with the state-of-the-art.*

1. Introduction

Dense scene reconstruction for photorealistic view synthesis has many critical applications, for example, in VR/AR (virtual traveling, preserving of important cultural

‡Part of the work was done while Andreas and Yu-Lun were interns at Meta.

artifacts), video processing (stabilization and special effects), and mapping (real-estate, human-level maps). Recently, rapid progress has been made in increasing the fidelity of reconstructions using radiance fields [22]. Unlike most traditional methods, radiance fields can model common phenomena such as view-dependent appearance, semi-transparency, and intricate micro-details.

Challenges. In this paper, we aim to create radiance field reconstructions of *large-scale* scenes that are acquired using a single handheld camera since this is arguably the most practical way of capturing them outside the realm of professional applications. In this setting, we are faced with two main challenges: (1) estimating accurate camera trajectory of a long path and (2) reconstructing the large-scale radiance fields of scenes. Resolving them together is difficult because changes in observation can be explained by either camera motion or the radiance field's ability to model view-dependent appearance. For this reason, many radiance field estimation techniques assume that the accurate poses are known in advance (typically fixed during the radiance field optimization). However, in practice, one has to use a separate method, such as Structure-from-Motion (SfM), for estimating the camera poses in a pre-processing step. Unfortunately, SfM is not robust in the handheld *video* setting. It frequently fails because, unlike radiance fields, it does not model view-dependent appearance and struggles in the absence of highly textured features and in the presence of even slight dynamic motion (such as swaying tree branches).

To remove the dependency on known camera poses, several approaches propose jointly optimizing camera poses and radiance fields [11,17,41]. These methods perform well when dealing with a few frames and a good pose initialization. However, as shown in our experiments, they have difficulty estimating long trajectories of a video camera from scratch and often fall into local minima.

Our work. In this paper, we propose a joint pose and radiance field estimation method. We design our method by drawing inspiration from classical *incremental SfM* algorithms and *keyframe-based SLAM* systems for improving the robustness. The core of our approach is to process the video sequence *progressively* using overlapping *local* radiance fields. More specifically, we progressively estimate the poses of input frames while updating the radiance fields. To model large-scale unbounded scenes, we dynamically instantiate local radiance fields. The increased locality and progressive optimization yield several major advantages:

- Our method scales to processing arbitrarily long videos without loss of accuracy and without hitting memory limitations.
- Increased robustness because the impact of misestimations is locally bounded.
- Increased sharpness because we use multiple radiance fields to model local details of the scene (see Figure 1 and 2b).

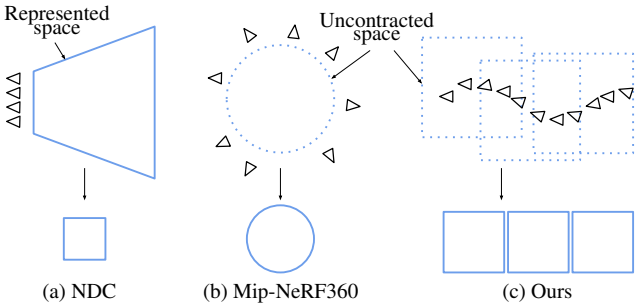


Figure 2. **Space parameterization.** (a) NDC, used by NeRF [22] for forward-facing scenes, maps a frustum to a unit cube volume. While a sensible approach for forward-facing cameras, it is only able to represent a small portion of a scene as the frustum cannot be extended beyond a field of view of 120° or so without significant distortion. (b) Mip-NeRF360's [4] space contraction squeezes the background and fits the entire space into a sphere of radius 2. It is designed for inward-facing 360 scenes and cannot scale to long trajectories. (c) Our approach allocates several radiance fields along the camera trajectory. Each radiance field maps the entire space to a $[-2, 2]$ cube (Equation (5)) and, each having its own center for contraction (Equatione (7)), the high-resolution uncontracted space follows the camera trajectory and our approach can adapt to any camera path.

We validate our method on the TANKS AND TEMPLES dataset. We also collect a new dataset STATIC HIKES of twelve outdoor scenes using four consumer cameras to evaluate our method. These sequences are challenging due to long handheld camera trajectories, motion blur, and complex appearance.

Our contributions. We present a new method for reconstructing the radiance field of a large-scale scene, which contains the following contributions:

- We propose to progressively estimate the camera poses and radiance fields, leading to significantly improved robustness.
- We show that multiple overlapping local radiance fields improve visual quality and support modeling large-scale unbounded scenes.
- We contribute a newly collected video dataset that presents new challenges not covered by existing view synthesis datasets.

Limitations. Our work aims to synthesize novel views from the reconstructed radiance fields. While we jointly estimate the poses in the pipeline, we do not perform global bundle adjustment and loop closure (i.e., not a complete SLAM system). We leave this important direction for future work.

2. Related Work

Novel view synthesis. Novel view synthesis aims to synthesize new views from multiple posed images. Recently, neural implicit representation has shown promising novel view synthesis results [22]. However, achieving high-quality artifact-free rendering results is still a challenging task. Recent works further improve the visual quality by addressing inconsistent camera exposure or illumination [20, 32, 36], handling dynamic elements [8, 16, 18, 28, 29, 43], anti-aliasing [3], high noise [21] or optimization from a reduced number of frames [27]. While these implicit representation-based methods yield high-quality results, they take days to train. To improve the training efficiency, some works also explore more explicit representations with voxel-like structures [33, 35], tensor factorization [6], light field representation [1, 2], or hashed voxel/MLP hybrid [24]. Our work also leverages the recent advantage of TensoRF [6].

Scalable view synthesis. Several systems have been proposed to support unbounded scenes [4, 47]. However, these methods require either omnidirectional inputs [10], proxy geometry [42], specialized drone shots [40], or satellite shots [44] and struggle with monocular videos captured at ground level. Recently, Mip-NeRF 360 [4] contracts background into a contracted space, and NeRF++ [47] optimizes an environment map to represent the background. BlockNeRF [36] is scalable but requires multiview inputs and several observations. NeRFusion [49] constructs per frame local feature volumes using a pretrained 2D CNN followed by a sparse 3D CNN. It is scalable and has demonstrated good accuracy on large indoor scenes, but it does not tackle camera pose estimation or unbounded outdoor scenes. Since the representation is mostly reconstructed before an optional per-scene optimization, it is not trivial to optimize poses simultaneously.

In contrast to all these constraints, our method is robust, works with arbitrary long camera trajectories, and only takes casually captured monocular first-person videos as input.

Camera pose estimation. Visual odometry estimates camera poses from videos. They can either rely directly on the color by maximizing the photoconsistency [46, 51] or on extracted hand-crafted features [25, 26, 34]. Recently, learning-based methods [9, 15, 39, 50, 51] learn to optimize the camera trajectories in a self-supervised way and show strong results. Similarly, many methods extend NeRF to optimize the camera poses jointly with radiance fields from photometric loss [11, 17, 41]. However, these methods struggle to reconstruct and synthesize faithful images for large scenes and often fail for monocular first-person videos with long camera trajectories. Vox-Fusion [45] and Nice-SLAM [53] achieve good pose estimation but are designed for RGB-D inputs and require accurate depth: Vox-Fusion to allocate a sparse voxel grid and Nice-SLAM to determine where to sample along the ray. Note that our goal does *not* lie in estimating camera poses. Instead, focus on reconstructing overlapping local radiance fields that enable photorealistic view synthesis. We believe integrating advanced techniques such as global bundle adjustment can improve our results.

3. Method

Our method takes a potentially very long monocular video of a large-scale scene as input. Our goal is to reconstruct the radiance field of the scene along with the camera trajectory to enable free-viewpoint novel view synthesis.

We choose TensoRF [6] as our base representation for its quality, reasonable training speed and model size. TensoRF models the scene with a factorized 4D tensor that maps a 3D position $\mathbf{x}$ to the corresponding volume density σ and view-dependent color $\mathbf{c}$. High-quality novel view synthesis results for small-scale scenes have been demonstrated using this representation. However, it has only been achieved with accurate pre-known camera poses, and TensoRF's representation power needs to be increased for capturing the details from long trajectories of unbounded scenes.

In this work, we resolve the need for pre-known camera poses by improving the *robustness* of joint camera pose and radiance field estimation, and we *scale the method* to handle arbitrarily long input sequences. To this end, we propose a progressive optimization scheme that processes the input video with a sweeping temporal window and incrementally updates the radiance fields and the camera poses. This process ensures that new frames are added to a well-converged solution for camera poses and radiance field representation of the previous structures, effectively preventing getting stuck in poor local minima. In addition, we dynamically allocate new *local* radiance fields throughout the optimization that are supervised by a limited number of input frames (within a temporal window). This further improves robustness while processing arbitrarily long videos with fixed memory.

3.1. Formulation and Preliminaries

During our optimization procedure, we estimate P camera poses $[R|t]_k, k \in [1..P]$[†] as well as the parameters of a series of M local radiance fields $\Theta_j, j \in [1..M]$.

Given a pixel, we use the camera parameters and the poses to generate a ray $\mathbf{r}$. Along this ray, we sample 3D positions $\{\mathbf{x}_i\}$ and query a radiance field that provides color and density:

$$(\mathbf{c}_i, \sigma_i) = \mathrm{RF}_{\Theta_j}(\mathbf{x}_i). \quad (1)$$

[†]We use the continuous 6D representation [52] for representing camera rotations.

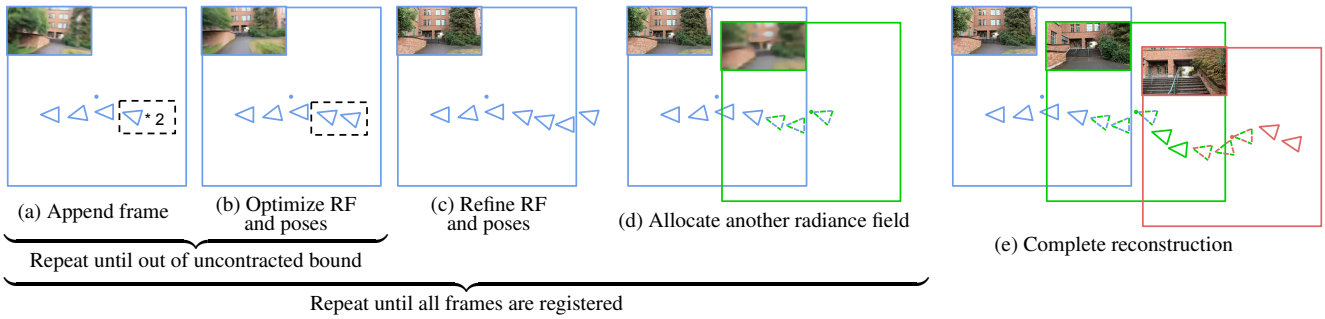


Figure 3. **Method overview.** The squares represent the uncontracted space of each local radiance field and the triangles are camera poses. The color of each camera pose indicates to which radiance fields it is linked and serves as supervision. We show as insert the renders intermediate at the intermediate optimization step for each local radiance field. (a) We add a frame at the end of the trajectory before (b) jointly estimating poses and the corresponding local radiance field. After the pose reaches the boundary of the high-resolution uncontracted space, (c) we run the optimization without adding frames to refine the poses and the radiance field. Then, to dynamically extend the representation, (d) we allocate a new radiance field. We repeat this process until we cover the full trajectory to produce (e) a complete reconstruction.

Via volume rendering [7, 12], we can render the ray using this representation:

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^{N} T_i (1 - \exp(-\sigma_i \delta_i)) \mathbf{c}_i, \tag{2}$$

$$T_i = \exp\left(- \sum_{i-1}^{j} \sigma_j \delta_j \right), \tag{3}$$

where δ_i is the distance between two consecutive sample points and N is the number of samples along the ray, and T_i indicates the accumulated transmittance along the ray. We optimize the radiance field parameters Θ_j and the camera pose used to generate the ray using the input frame's color $\mathbf{C}$ as supervision:

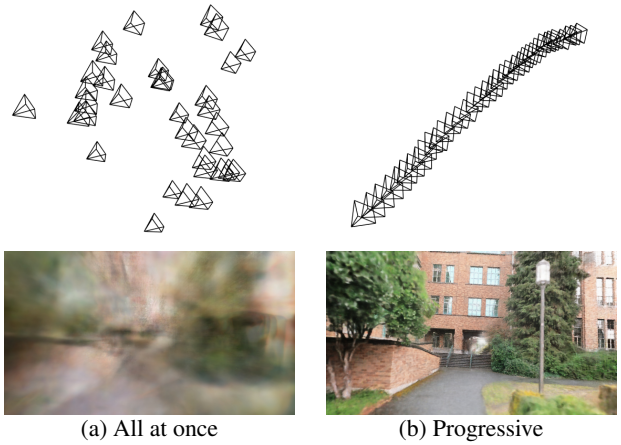
$$\mathcal{L} = \left\| \hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r}) \right\|_2^2. \tag{4}$$

Using TensoRF [6] in this context is particularly suitable as it features an explicit coarse-to-fine optimization analogous to BARF's [17] and reduces the likelihood of converging to a local minimum for pose estimation.

To handle unbounded scenes, we leverage a scene parameterization similar to Mip-NeRF360 [4]'s contraction, mapping every point to a $[-2, 2]$ space before querying our radiance field model:

$$\text{contract}(\mathbf{x}) = \begin{cases} \mathbf{x} & \text{if } \|\mathbf{x}\|_\infty \leq 1 \\ \left(2 - \frac{1}{\|\mathbf{x}\|_\infty}\right) \left(\frac{\mathbf{x}}{\|\mathbf{x}\|_\infty}\right) & \text{otherwise.} \end{cases} \tag{5}$$

Here we use the L_∞ norm to fully utilize TensoRF's square bounding boxes. While Mip-NeRF360 scale camera poses to keep the uncontracted space around the area of interest, we cannot adopt this strategy since we jointly estimate poses and radiance fields (the poses are unknown a priori). We achieve appropriate scaling by dynamically creating new radiance fields (see Figure 2 and Section 3.3).



(a) All at once (b) Progressive

Figure 4. **Importance of progressive optimization.** (a) When estimating all camera poses at once and from scratch, the relationship between the poses is lost and we obtain this disjointed camera trajectory. In this scene, the camera follows a forward trajectory, coherent with (b) the path estimated with progressive optimization.

3.2. Progressive Joint Camera Pose and Radiance Field Optimization

Existing pose-calibrating methods [11, 17, 41] have demonstrated that jointly optimizing a radiance field and camera poses can achieve satisfactory results in small-scale scenes. However, when dealing with longer sequences, joint optimization fails as estimated poses get stuck in local minima (see Figure 4). To improve the robustness, we start the optimization process with only a small number of frames (the first five frames in our experiments), and from there we *progressively* introduce subsequent frames to the optimization. To this end, we initialize the new pose (index $p + 1$) using

Global Local

Figure 5. **Importance of locality.** When using a single global radiance field, the whole scene reconstruction fails if a few camera poses are not estimated correctly (first row). Local radiance fields also allow us to dynamically allocate the representation's capacity around the camera trajectory, producing sharper results (second row).

the current frame at the end of the trajectory:

$$[R|t]_{p+1} \leftarrow [R|t]_p. \tag{6}$$

We then add $[R|t]_{p+1}$ to the trainable parameters and we add the frame's color as supervision of the radiance fields. In this scheme, the convergence of the parameters for the new frame benefits from the initialization of the radiance field and the currently estimated poses, making it less prone to get stuck in local minima. Since we add the camera pose at the end of the trajectory, it also introduces a locality prior that enforces each pose to be close to the former one without an explicit constraint, as it is common for videos.

To further constrain this complex joint optimization problem, we introduce additional losses described in Section 3.4.

3.3. Local Radiance Fields

The progressive scheme proposed in the previous section provides more robust pose estimation, but it still relies on a single global representation of the scene, which causes problems when modeling long videos: (1) any misestimation (e.g., outlier pose) has a global impact and might cause the entire reconstruction to break down. (2) a single model with fixed capacity cannot represent arbitrarily long videos with an arbitrary amount of detail, leading to blurry renderings (Figure 5b). A natural solution to these problems would be to pre-partition the space using radiance field tiling similar to Mega-NeRF [40]. However, this approach is not applicable in our setting because the camera poses are unknown before the optimization. To resolve this issue, we dynamically create a new radiance field, whenever the estimated camera pose trajectory leaves the uncontracted space of the current radiance field. We centered the new radiance field at the location $\mathbf{t}_j$ of the last estimated camera pose: $\mathbf{t}_j \leftarrow t_p$ (Figure 3d). When sampling a ray, we use this translation to center the radiance fields:

$$(\mathbf{c}_i, \sigma_i) = \mathrm{RF}_{\Theta_j}(\mathbf{x}_i - \mathbf{t}_j). \tag{7}$$

We supervise each radiance field with a local subset of the video frames. This subset contains all the frames during which the radiance field was current, as well as the preceding 30 frames to provide for some overlap. The overlap is essential for achieving consistent reconstructions in the local radiance fields. We further increase consistency by blending together the rendered colors $\hat{\mathbf{C}}_j(\mathbf{r})$ of all overlapping radiance fields at any supervising frame. We use per-frame blending weights that increase/decrease linearly within the overlap region. When we create a new radiance field we stop optimizing previous ones (i.e., freeze them). At this point, we can clear any supervising frames from memory that are not needed anymore.

With this procedure, each radiance field is supervised using only a subset of the frames, granting more robustness. The high-resolution space follows the camera trajectory, allowing for more sharpness and scalability. Figure 2 shows that local radiance fields help maintain the high-quality uncontracted space around the camera trajectory, which allows for a sharper representation.

3.4. Implementation

Losses. In addition to the supervision from the input color in Equation (4), we add monocular depth and optical flow between neighbor frames as they have proven their capabilities to improve the optimization stability in challenging scenarios [8, 19]. We use RAFT [38] to estimate the frame-to-frame optical flow $\mathcal{F}_{k\to k+1}, k \in [1..P-1]$ and DPT [30] to estimate the per-frame monocular depth $\mathbf{D}$. To enable these losses, we first render the depth maps by swapping the sample's color $\mathbf{c}_i$ by the sample's distance to the ray origin d_i in Equation (2):

$$\hat{\mathbf{D}}(\mathbf{r}) = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i\delta_i))d_i. \tag{8}$$

We formulate depth supervision following the shift and scale invariant loss typically used for monocular depth training [30, 31] and radiance fields supervision [8]:

$$\mathcal{L}_d = \left|\hat{\mathbf{D}}^* - \mathbf{D}^*\right|, \tag{9}$$

where $\hat{\mathbf{D}}^*$ and $\mathbf{D}^*$ are per-frame normalized depth since monocular depth is not scale- and shift-invariant. We normalize following [31], by first estimating scale and shift:

$$t(\mathbf{D}) = \mathrm{median}(\mathbf{D}),\ s(\mathbf{D}) = \frac{1}{M}\sum_{1}^{M}|\mathbf{D} - t(\mathbf{D})|, \tag{10}$$

where we have M samples for the frame. Then, we normalize:

$$\mathbf{D}^* = \frac{\mathbf{D} - t(\mathbf{D})}{s(\mathbf{D})}. \quad (11)$$

In practice, we sample rays from 16 images in each batch and obtain scale and shift for each. To obtain the expected optical flow from our representation, we leverage the relative camera poses and the rendered depth map:

$$\hat{\mathcal{F}}_{k\to k+1} = (u, v) - \Pi\left([R|t]_{k\to k+1}\Pi^{-1}(u, v, \hat{D})\right) \quad (12)$$

Where Π projects a 3D point to image coordinates and Π^{-1} unprojects a pixel coordinate and depth into a 3D point and $[R|t]_{k\to k+1} = [R|t]_k^{-1}[R|t]_{k+1}$ is the relative camera pose between the two consecutive frames, bringing a point in the k^{th} camera's space to the $k+1^{\text{th}}$'s. We finally compare the predicted flow to the expected flow from the representation:

$$\mathcal{L}_f = \left\| \hat{\mathcal{F}}_{k\to k+1} + \mathcal{F}_{k\to k+1} \right\|_1 \quad (13)$$

We use the same process to supervise using the backward optical flow $\mathcal{F}_{k\to k-1}$. Note that optical flow computations leverage directly poses and the geometry of the scene in Equation (12), which gives a clear gradient signal for their optimization.

Scheduling. All parameters are optimized using Adam [13] with $\beta_1 = 0.9$ and $\beta_1 = 0.99$. We start with five poses initialized as identity and an initial TensoRF model. Then, for every 100 iterations, we add the next supervising frame in the video as described in Section 3.2 and Figure 3a and 3b. During this optimization process, we maintain all learning rates, loss weights, and TensoRF resolution at their initial states. This ensures that the radiance field does *not* overfit to the first frames. Our initial learning rates are $5{\cdot}10^{-3}$ for rotations and $5{\cdot}10^{-4}$ for translations, initial TensoRF resolution is 64^3 and initial regularizing loss weights are 1 for flow loss and 0.1 for depth. We proceed with the progressive frame registration until an estimated camera pose is beyond the uncontracted space: $\|t_p - \mathbf{t}_j\|_\infty \geq 1$, where t_p the last registered frame's translation and $\mathbf{t}_j$ the currently optimizing radiance fields center. From this point, we refine the TensoRF, and the camera poses for 600 iterations per add frame (Figure 3c). The schedulers and the regularizing losses follow an exponential decrease towards a 0.1 factor, and we upsample the TensoRF up to 640^3. After this stage, we allocate a new TensoRF following Section 3.3 and Figure 3d and disable the supervision from the first frames. We repeat this process until the entire trajectory is reconstructed. The optimization takes 30 to 40 hours for 1000 frames on a single NVIDIA TITAN RTX.

Table 1. **Novel view synthesis results.** We report the average PSNR, SSIM and LPIPS results with comparisons to existing methods on TANK AND TEMPLES dataset. The best performance is in **bold** and the second best is underscored. The COLMAP rows use COLMAP camera parameters as initialization. When fixed, we keep the poses throughout the entire optimization. When refined, we jointly optimize poses and radiance fields. The self-calibrated methods start pose optimization from scratch.

Pose estimation			PSNR ↑	SSIM ↑	LPIPS ↓
COLMAP	fixed	NeRF++ [47]	19.59	0.562	0.682
		Mega-NeRF [40]	18.09	0.548	0.622
		Nerfacto [37]	16.85	0.641	0.466
		Mip-NeRF360 [4]	21.09	**0.714**	**0.406**
		LocalRF (ours)	**22.40**	0.667	0.494
	refined	SCNeRF [11]	15.88	0.506	0.708
		BARF [17]	9.62	0.383	0.893
		LocalRF (ours)	**22.85**	**0.676**	**0.475**
Self calib.		BARF [17]	11.78	0.430	0.884
		LocalRF (ours)	**20.41**	**0.593**	**0.624**

4. Experimental Results

4.1. Datasets

Tanks and Temples. We evaluate our method on the TANKS AND TEMPLES dataset [14]. We select the sequences without dynamic elements (9 scenes out of 21), retain one in every five frames as motion is slow, downscale the video to full HD resolution (2048×1080 or 1920×1080), and keep the first 1000 images so methods with a static data loader can preload images and rays in a reasonable amount of system memory.

Static Hikes. We also collect a new dataset with hiking sequences. It contains hand-held sequences with larger camera trajectories to test scalability and pose estimation robustness. It comprises twelve 1920×1080 videos of static outdoor scenes captured with GoPro Hero10 with linear FoV, GoPro Hero9 with narrow FoV, and the wide cameras of LG V60 ThinQ and Samsung Galaxy S21.

4.2. Compared Methods

We compare our method against Mip-NeRF360 [4] and NeRF++ [47] as they are both designed to handle unbounded scenes and are suitable for outdoor scenes. For Mip-NeRF360, we set near to 0.1 as we observed clipping with the default value. Nerfacto [37] combines Instant-NGP [24]'s hash encoding and Mip-NeRF360's scene contraction to efficiently represent unbounded scenes. When preprocessed poses are required, we use MultiNeRF [23]'s script to run COLMAP. We also compare against the scalable representation Mega-NeRF [40] (using a 2×2 grid size for our experiments). Since those methods require preprocessed poses, we include SCNeRF [11] and BARF [17]

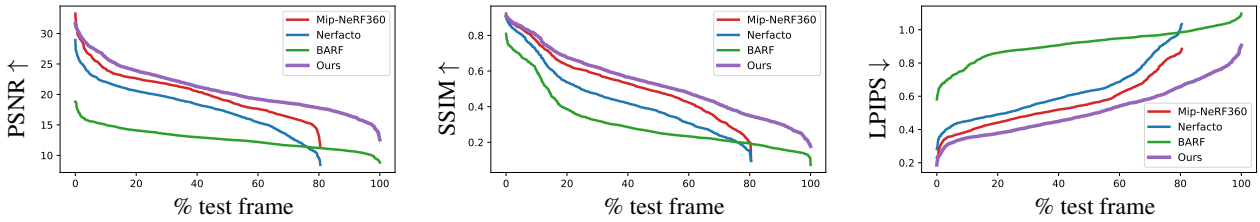


Figure 6. **Novel view synthesis evaluation on the STATIC HIKES dataset.** We plot the sorted frame-wise metrics to assess the quality distribution across the dataset. Since COLMAP is not able to register poses for all frames, Mip-NeRF360's and Nerfacto's curves terminate early. Only BARF and our method estimate pose for each frame, but BARF cannot maintain high accuracy when optimizing a large number of camera poses at once.

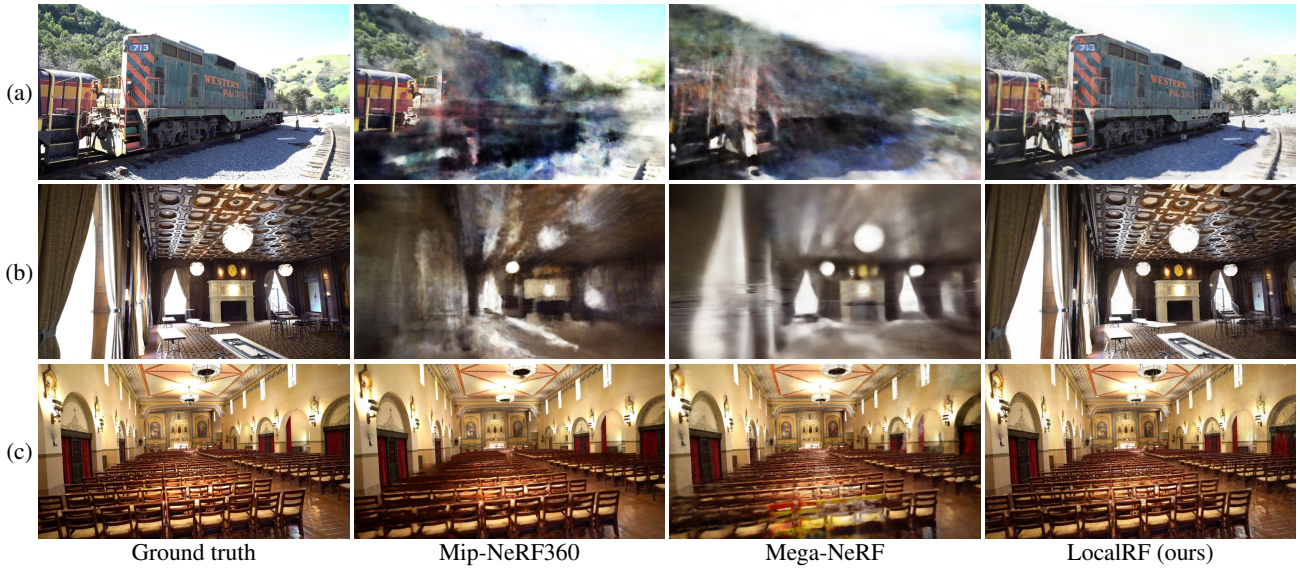


Figure 7. **Novel view synthesis results on TANKS AND TEMPLES dataset.** (a) and (b) Locality allows for more robustness to illumination changes and pose estimation failures. (c) We can obtain sharper results since the less contracted space follows the trajectory.

for self-calibrated experiments. For SCNeRF, we use the NeRF++ codebase to better represent unbounded scenes. Note that SCNeRF requires COLMAP as initialization: it fails in our experiments when optimizing poses from scratch with both the NeRF and NeRF++ bases (we get NaN renders). We, therefore, had to exclude SCNeRF from the fully self-calibrated evaluations.

4.3. Quantitative Evaluation

To quantitatively evaluate the synthesized novel views, we select every ten frames as a *test* image. We show the PSNR, SSIM, and LPIPS [48] between the synthesized views and the corresponding ground truth views in Table 1. Averages are computed in the square error domain for PSNR and $\sqrt{1-\text{SSIM}}$ domain for SSIM, following [5]. For the self-calibrated experiments, we estimate the test poses by adding iterations that only optimize the poses, *without* updating the intrinsic or the radiance fields' parameters.

Table 1 shows that, although the videos are not what we target, with a smaller camera path and several inward-looking 360 scenes, our method provides competitive results. With COLMAP poses, we obtain similar quality as Mip-NeRF360 [4]. Mega-NeRF [40], being designed for a different type of input data, shows lower quality despite using several radiance fields. Compared to other self-calibrating radiance fields methods [11,17], we obtain much better results when optimizing poses from scratch thanks to our progressive optimization that allows for fewer parameters to be estimated from scratch at once and adds a flexible locality prior to the camera pose.

Figure 6 shows that, on the STATIC HIKES dataset featuring longer trajectories and more challenging scenes, we consistently obtain better results than the self-calibrated method BARF [17]. Mip-NeRF360 [4], relying on COLMAP, cannot produce results for 19.5% of the test frames. In addition, we show higher quality on the rendered frames.

4.4. Qualitative evaluation

Figure 7 compares the results on the TANKS AND TEMPLES dataset, and Figures 8 and 9 show results on the STATIC HIKES dataset we acquired. Our approach can provide re-

Figure 8. **Novel view synthesis results on the STATIC HIKES dataset.** Local radiance fields allow us to maintain sharpness throughout the trajectory. Some Mip-NeRF360 results, relying on preprocessed poses, are missing. Our method can optimize poses robustly, which allows good results even in scenes where other methods are less reliable.

Figure 9. **Input path deviation.** We can render novel views that deviate from the input path.

Table 2. **Ablation study.** We report PSNR, SSIM and LPIPS on five scenes of the STATIC HIKES dataset.

	PSNR ↑	SSIM ↑	LPIPS ↓
Ours w/o progressive optimization	14.55	0.275	0.918
Ours w/o local RF	16.00	0.306	0.838
LocalRF (ours)	**18.83**	**0.507**	**0.564**

sults for all frames and maintain better sharpness throughout the entire trajectory.

4.5. Ablation Study

Table 2 shows that our progressive optimization and local radiance fields are both necessary to obtain our results. Furthermore, Figure 4 highlights that progressive optimization is crucial when estimating poses for long sequences, and Figure 5 shows that local radiance fields grant more robustness and allow for scalability.

4.6. Limitations

We show that our method can estimate long camera trajectories robustly while maintaining a high-resolution representation. However, our pose estimation and progressive scheme assume that we are working with a continuous video *without shot changes*. This means that our method is not fit to reconstruct a scene from a collection of unstructured frames without coherence. We also do not tackle dynamic elements. Dynamic elements in the last row of Figure 8 lead to blurry regions. Another limitation that we observed is that sudden rotations can break pose estimation, leading to poorly rendered images.

5. Conclusions

We have presented a new method for reconstructing radiance fields of a large scene from a casually captured video. The core ideas of our work are 1) a *progressive* optimization scheme for jointly estimating camera poses and radiance fields and 2) dynamically instantiating *local* radiance fields. From extensive evaluation on two datasets, we show that the proposed method substantially improves the robustness and fidelity of radiance field reconstruction in this challenging scenario.

References

[1] Benjamin Attal, Jia-Bin Huang, Christian Richardt, Michael Zollhoefer, Johannes Kopf, Matthew O'Toole, and Changil Kim. HyperReel: High-fidelity 6-DoF video with ray-conditioned sampling. In *CVPR*, 2023. 3

[2] Benjamin Attal, Jia-Bin Huang, Michael Zollhöfer, Johannes Kopf, and Changil Kim. Learning neural light fields with ray-space embedding networks. In *CVPR*, 2022. 3

[3] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. *ICCV*, 2021. 3

[4] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022. 1, 2, 3, 4, 6, 7

[5] Dominique Brunet, Edward R. Vrscay, and Zhou Wang. On the mathematical properties of the structural similarity index. *TIP*, 2012. 7

[6] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. *ECCV*, 2022. 3, 4

[7] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM TOG*, 1988. 4

[8] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *ICCV*, 2021. 3, 5

[9] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019. 3

[10] Hyeonjoong Jang, Andréas Meuleman, Dahyun Kang, Donggun Kim, Christian Richardt, and Min H. Kim. Egocentric scene reconstruction from an omnidirectional video. *ACM TOG (Proc. SIGGRAPH 2022)*, 2022. 3

[11] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Animashree Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *ICCV*, 2021. 2, 3, 4, 6, 7

[12] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM TOG*, 1984. 4

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[14] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *TOG*, 2017. 6

[15] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *CVPR*, 2021. 3

[16] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 3

[17] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *ICCV*, 2021. 1, 2, 3, 4, 6, 7

[18] Yu-Lun Liu, Chen Gao, Andreas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust dynamic radiance fields. In *CVPR*, 2023. 3

[19] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *TOG*, 2020. 5

[20] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*, 2021. 3

[21] Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul P. Srinivasan, and Jonathan T. Barron. NeRF in the dark: High dynamic range view synthesis from noisy raw images. *CVPR*, 2022. 3

[22] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2, 3

[23] Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, Peter Hedman, Ricardo Martin-Brualla, and Jonathan T. Barron. MultiNeRF: A Code Release for Mip-NeRF 360, Ref-NeRF, and RawNeRF, 2022. 6

[24] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *TOG*, 2022. 3, 6

[25] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 2015. 3

[26] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 2017. 3

[27] Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *CVPR*, 2022. 3

[28] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *ICCV*, 2021. 3

[29] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *TOG*, 2021. 3

[30] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *ArXiv preprint*, 2021. 5

[31] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *TPAMI*, 2020. 5

[32] Konstantinos Rematas, Andrew Liu, Pratul P. Srinivasan, Jonathan T. Barron, Andrea Tagliasacchi, Tom Funkhouser, and Vittorio Ferrari. Urban radiance fields. *CVPR*, 2022. 3

[33] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, 2022. 3

[34] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 3

[35] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *CVPR*, 2022. 3

[36] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul Srinivasan, Jonathan T. Barron, and Henrik Kretzschmar. Block-NeRF: Scalable large scene neural view synthesis. *arXiv*, 2022. 3

[37] Matthew Tancik*, Ethan Weber*, Evonne Ng*, Ruilong Li, Terrance Wang Brent Yi, Alexander Kristoffersen, Jake Austin, Abhik Ahuja Kamyar Salahi, David McAllister, and Angjoo Kanazawa. Nerfstudio: A framework for neural radiance field development, 2022. 6

[38] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 5

[39] Zachary Teed and Jia Deng. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. *Advances in neural information processing systems*, 2021. 3

[40] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *CVPR*, 2022. 3, 5, 6, 7

[41] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. NeRF−−: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 2, 3, 4

[42] Xiuchao Wu, Jiamin Xu, Zihan Zhu, Hujun Bao, Qixing Huang, James Tompkin, and Weiwei Xu. Scalable neural indoor scene rendering. *TOG*, 2022. 3

[43] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *CVPR*, 2021. 3

[44] Yuanbo Xiangli, Linning Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In *ECCV*, 2022. 3

[45] Xingrui Yang, Hai Li, Hongjia Zhai, Yuhang Ming, Yuqian Liu, and Guofeng Zhang. Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation. *arXiv:2210.15858*, 2022. 3

[46] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018. 3

[47] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv:2010.07492*, 2020. 3, 6

[48] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7

[49] Xiaoshuai Zhang, Sai Bi, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Nerfusion: Fusing radiance fields for large-scale scene reconstruction. *CVPR*, 2022. 3

[50] Wang Zhao, Shaohui Liu, Hengkai Guo, Wenping Wang, and Yong-Jin Liu. Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In *ECCV*, 2022. 3

[51] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 3

[52] Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. On the continuity of rotation representations in neural networks. In *CVPR*, 2019. 3

[53] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *CVPR*, 2022. 3