2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

LATITUDE: Robotic Global Localization with Truncated Dynamic Low-pass Filter in City-scale NeRF

Zhenxin Zhu[1,2*], Yuantao Chen[1,3*], Zirui Wu[1,4], Chao Hou[1,5], Yongliang Shi[1†], Chuxuan Li[1], Pengfei Li[1], Hao Zhao[1], Guyue Zhou[1]

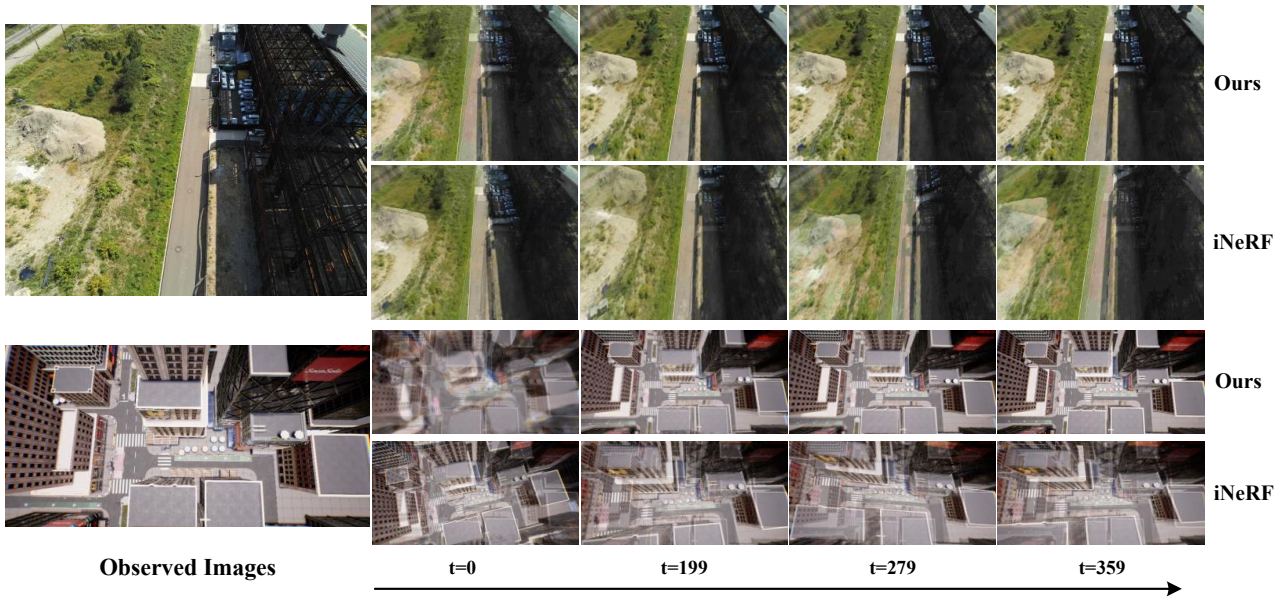

Fig. 1. When navigating in city, a new observation is rendered by NeRF for each predicted pose. During iterative optimization, the rendered observations at given positions are updated. With a coarse-to-fine optimization strategy, although a worse initial value is given, our optimization method converges to the exact position and obtains the picture that is almost identical to the actual observation in both real-world (top) and simulation (bottom) scenes.

***Abstract*— Neural Radiance Fields (NeRFs) have made great success in representing complex 3D scenes with high-resolution details and efficient memory. Nevertheless, current NeRF-based pose estimators have no initial pose prediction and are prone to local optima during optimization. In this paper, we present LATITUDE: Global Localization with Truncated Dynamic Low-pass Filter, which introduces a two-stage localization mechanism in city-scale NeRF. In place recognition stage, we train a regressor through images generated from trained NeRFs, which provides an initial value for global localization. In pose optimization stage, we minimize the residual between the observed image and rendered image by directly optimizing the pose on the tangent plane. To avoid falling into local optimum, we introduce a Truncated Dynamic Low-pass Filter (TDLF) for coarse-to-fine pose registration. We evaluate our method on both synthetic and real-world data and show its potential applications for high-precision navigation in large-scale city scenes. Codes and dataset will be publicly available at https://github.com/jike5/LATITUDE.**

*Equal contribution, †Corresponding author
[1]Institute for AI Industry Research (AIR), Tsinghua University, China, {shiyongliang, lichuxuan, lipengfei, zhaohao, zhouguyue}@air.tsinghua.edu.cn.
[2]Beihang University, China, zhuzhenxin@buaa.edu.cn.
[3]Xi'an University of architecture and technology, China, yuantao@xauat.edu.cn.
[4]Beijing Institute of Technology, China, wuzirui@bit.edu.cn.
[5]The University of Hong Kong, China, houchao@connect.hku.hk.

I. Introduction

Global camera localization is an essential prerequisite for autonomous navigation tasks. Previous APR-based (Absolute Pose Regression) methods [8] [13] [5] implicitly match the scene landmarks with image features, which predict rough location from input images but are prone to poor accuracy. Map-based methods localize the global location of a given observation with explicit maps [3] [4], including point cloud, grid-based and mesh-based maps, etc. However, these methods struggle to encode detailed scene appearance using the reasonable disk and memory consumption and thus are limited in generalizing to accurate global localization in city-scale scenes.

On the other hand, the success of NeRF [12] and its follow-up works [23] represent 3D scenes with neural implicit functions and can render photo-realistic images under arbitrarily given perspectives in the scenes, which offer a possibility to further extend the accuracy of global localization methods by explicitly matching pixel-wised misplacement via stochastic gradient descent (SGD). Recently, there has been a significant advancement in the field of NeRF-based state estimation [1] [28] which iteratively optimizes the predicted camera poses and minimizes the photometric errors. Current works, however, are primarily restricted to

indoor scenes that are small in scale and in good illumination condition, which are rich in joint visibility and facilitate convergence. However, generalizing current methods to larger scenes introduces limitations in localization accuracy. This is because, in large-scale scenes, it is hard to find the co-visibility between distant viewpoints, which causes significant difficulties in optimization-based methods.

As mentioned above, optimization-based methods can localize accurate locations but have strict constraints on co-visibility between initial and actual viewpoints, while APR-based methods predict roughly located camera poses, which may potentially provide a good initialization to bootstrap the optimization. To this end, we propose the first method that combines an APR model with a city-scale NeRF for accurate global localization.

Our method have two stages: (1) in the first stage, referred to as the place recognition stage, we train an absolute pose regressor that maps the observed images to their rough global locations; (2) in the second stage, i.e., the pose optimization stage, we iteratively optimize the predicted poses based on a pre-trained large-scale NeRF [23]. During the place recognition stage, we leverage the fact that a large-scale NeRF-based map is available and thus can generate pseudo-pose-image pairs to augment original training data. In the pose optimization stage, we propose to apply a truncated dynamic low-pass filter (TDLF) on the NeRF's positional encoding, which applies a smooth mask on the encoding at different frequency bands (from non-zero to full) over the course of optimization. This coarse-to-fine optimization strategy avoids the local optimum caused by high-frequency information.

To summarize, our contributions are as follows:

- A two-stage global localization mechanism in city-scale NeRF is achieved.
  a) NeRF-assisted place recognition is achieved, which provides a reliable initial value for pose optimization.
  b) The TDLF is proposed to realize the coarse-to-fine optimization for the initial pose obtained by our NeRF-assisted place recognition.
- We release a dataset for the validation of NeRF-based localization in city-scale simulation.

II. Related Work

A. NeRF-based Representations For Large Scene

When NeRF is applied to large-scale 3D scene representation, issues arise such as changes in brightness, difficulties in external dynamic object modeling, and unbalanced rendering of the foreground and background in unconstrained scenes. Martin-Brualla et al. [11] used per-frame latent codes to eliminate the difference in brightness and dynamic object appearances. Zhang et al. [29] solved the imbalance problem of NeRF in foreground and background rendering by training two MLPs respectively. Recently, several approaches [22] [23] [27] have successfully applied NeRF to the implicit reconstruction of city-scale scenes. Among them, Mega-NeRF [23] achieves cutting-edge results in novel view synthesis by dividing the map into cells in the spatial domain and fitting a set of NeRFs in parallel to represent the whole scene. Since NeRF does not constrain volume density distribution during training, NeRF-based surface reconstruction typically encounters problems such as inaccuracy and floating objects. A feasible solution is to introduce additional supervision, e.g. depth maps [7] [14] and normal priors [24].

B. Place Recognition

Place Recongnition (PR) is crucial for the performance of a robot's navigation. A typical pipeline for the PR includes feature extraction, image retrieval and/or 2D-3D feature matching [15] [16] [17]. However, its high cost in terms of computation and memory footprint leaves room for improvement. PoseNet [8] proposes the method called absolute pose regression (APR) which first addresses this problem using a deep neural network. Still, in large-scale scenes, it is difficult to cover the entire scene from one single perspective, so optimizing poses at any position is especially important. LENS [13] has validated that introducing NeRF for data augmentation improves the localization accuracy. However, its training phase is still not memory-efficient. Direct-PoseNet [6] and DFNet [5] deliver the state-of-the-art performances.

C. Pose Estimation with NeRF

Continuous scene representation makes it possible to recover the camera position of a novel view from a pre-trained NeRF. iNeRF [28] presents the first framework to directly optimize 6DoF poses with fixed network parameters by minimizing the photometric residual between rendered pixels and observed pixels. [9] [26] then success in jointly optimizing camera parameters and scene representations, enabling training NeRF under inaccurate poses. Meanwhile, BARF [9] introduces a coarse-to-fine pose registration to escape the suboptimal solutions caused by high-frequency positional encoding. Further works [21] [31] simultaneously optimize poses and implicit decoder under time-series image sequences. iMAP [21] shows the first real-time NeRF-based SLAM in the indoor environment. It has a similar pipeline to traditional visual SLAM but substitutes visual features with a single MLP in both tracking and mapping threads. Given RGB-D image sequences, iMAP tracks frames by optimizing poses like iNeRF [28] and selects keyframes for global bundle adjustment for scene mapping. However, these methods are restricted to small indoor scenes.

III. Formulation

To represent a 3D scene for novel view synthesis, Neural Radiance Fields (NeRFs) employ neural networks $\mathcal{F}(\varepsilon)$: $(\mathbf{x}, \mathbf{d}) \longmapsto (\mathbf{c}, \sigma)$ to map position $\mathbf{x}$ and ray direction $\mathbf{d}$ to an emitted color $\mathbf{c}$ and density σ. We propose a robotic global localization mechanism in a city-scale scene represented by NeRF. Given an observed image, our APR model will predict an initial pose. Then the accurate pose is optimized by the NeRF. In essence, our localization mechanism is considered as Maximum A Posterior (MAP) Estimation

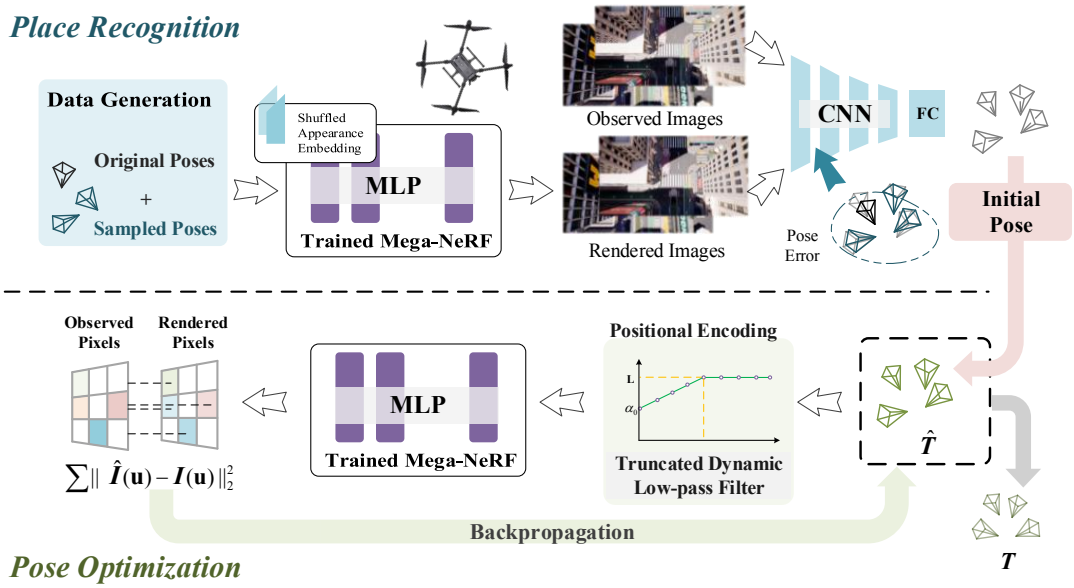

Fig. 2. The place recognition module uses camera poses to train the pose regressor. Firstly, additional camera poses are sampled around the original camera poses. Then, the pose vector is passed through Mega-NeRF with shuffled appearance embeddings. Subsequently, the initial poses of the inputted images are predicted by a pose regressor network. During optimization process, on the basis of initial pose, we render the images through Mega-NeRF and backpropagate the photometric error to get a more accurate pose with the TDLF.

problem. Obtained a prior by place recognition $p(\hat{T}_k|I_k)$, we check against the implicit map (Mega-NeRF) in the light of observations to update the prior, and derive our posterior belief about the current location, as is described in eq. (1).

$$p\left(\hat{T}_k|\mathcal{F}(\varepsilon), I_k\right) \propto p\left(\hat{I}_k|\hat{T}_k, \mathcal{F}(\varepsilon)\right) p\left(\hat{T}_k|I_k\right) \quad (1)$$

Here, the I_{k} represents the visual observation at time k, and $\hat{I}_k$ is an image rendered by our Mega-NeRF $\mathcal{F}(\varepsilon)$. To achieve an accurate state T_k, we optimize the posterior $p(\hat{T}_k|\mathcal{F}(\varepsilon), I_k)$ iteratively by minimizing the photometric loss between the observation I_k and generated image $\hat{I}_k$ by the Mega-NeRF within fixed parameters ε.

IV. Method

A. System Overview

As is shown in Fig.2, the system includes two modules: place recognition and pose optimization. We first train a NeRF to implicitly model a city-scale scene. The next, combined with actual observed images, a set of images are generated through the Mega-NeRF to train a pose regressor for place recognition. After an initial pose is obtained by the trained regressor, the Mega-NeRF with fixed parameters can optimize it through backpropagation according to loss between rendered and observed images. During optimization, a TDLF is introduced to achieve coarse-to-fine registration.

B. Place Recognition

Place recognition [8] is a prevailing method to address the problem of global localization. In our system, an Absolute Pose Regression method (APR) is adopted to provide reliable initial values at any location in the scene. Our work is divided into two parts: data augmentation and network architecture.

Data Augmentation: Data augmentation is useful to improve performance and outcomes of learning models [19]. In this work, a set of poses sampled near original poses are passed through to the trained Mega-NeRF, and the generated images are considered as data enhancement. Due to the factor that altitude is usually constant in a conventional flight path. To begin, we uniformly sample several positions in a horizontal $H \times W$ rectangle area around each position in training set where H and W is a given parameter. Then, we need to define the camera orientation attached to these positions for each virtual camera pose. To avoid degenerate views, we copy the pose orientations of the training set, add random perturbations on each axis drawn evenly in $[-\theta, \theta]$, where θ is the maximum amplitude of the perturbation. Besides, to avoid memory explosion, we generate the poses using the method above and use Mega-NeRF to render images during specific epochs of pose regression training. Additionally, Mega-NeRF's appearance embeddings are selected by randomly interpolating those of the training set, which can be considered as a data augmentation technique to improve the robustness of the APR model under different lighting conditions.

Network Architecture: Built on top of VGG16's [30] light network structure, we use 4 full connection layers to learn pose information from image sequences. Given an input image I_{real} and its corresponding ground truth pose $T_{real}(\mathbf{x}_{real}, \mathbf{q}_{real})$, we first use the generated virtual poses $T_{syn}(\mathbf{x}_{syn}, \mathbf{q}_{syn})$ and render a set of virtual images I_{syn}. Then the two types of images are inputted into the regressor to get estimated poses $T_{syn}(\hat{\mathbf{x}}_{syn}, \hat{\mathbf{q}}_{syn})$ and $T_{real}(\hat{\mathbf{x}}_{real}, \hat{\mathbf{q}}_{real})$. Finally, our loss functions shown as in eq. (2), and to balance them we introduce two scale factors, where γ keeping the expected value of position

and orientation error equally, and β ensuring the quality of rendered images. In general, the model should trust more on real-world data and learn more from it.

$$\mathcal{L}_{real} = \|\hat{\mathbf{x}}_{real} - \mathbf{x}_{real}\|_2 + \gamma \left\| \hat{\mathbf{q}}_{real} - \frac{\mathbf{q}_{real}}{\|\mathbf{q}_{real}\|} \right\|_2 \tag{2}$$

$$\mathcal{L}_{syn} = \|\hat{\mathbf{x}}_{syn} - \mathbf{x}_{syn}\|_2 + \gamma \left\| \hat{\mathbf{q}}_{syn} - \frac{\mathbf{q}_{syn}}{\|\mathbf{q}_{syn}\|} \right\|_2$$

$$\mathcal{L} = \mathcal{L}_{real} + \beta \mathcal{L}_{syn} \tag{3}$$

C. Pose Optimization

The initial pose obtained from the place recognition is limited in terms of precision. In order to further improve the accuracy, we optimize pose on tangent plane to ensure a smoother convergence on one hand, and on the other hand by implementing the TDLF can avoid from falling into the local optimum during optimization course.

Optimization on Tangent Plane: Generally, gradient-based optimization on SE(3) is utilized to solve for the pose estimation [28]. However, as mentioned in [1], optimization on the tangent plane can performs smoother and quicker convergence. We use the same approach as in [1] and formulate the problem as eq. (4), where I is an observed image and $\hat{T}_0$ is the associated initial pose.

$$\hat{\xi} = \underset{\xi \in \mathfrak{se}(3)}{\arg\min}\, \mathcal{L}(\xi | \hat{T}_0, I, \theta) \tag{4}$$

We use eq. (5) to obtain the optimal pose estimation, where $\hat{\xi}$ is the optimization variables.

$$\hat{T} = \exp(\hat{\xi})\hat{T}_0 \tag{5}$$

To further demonstrate our loss function $\mathcal{L}$, we introduce the 3D world coordinates, where T denotes the transformation from camera coordinates to the world coordinates. Then given some sampled set ($\mathcal{R}$) of pixel coordinates: $\mathbf{u} \in \mathcal{R}$, the loss function can be expanded as eq. (6), where K is camera intrinsic parameters and $\mathcal{I} : \mathbb{R}^{4N} \rightarrow \mathbb{R}^3$ is compositing function. Our goal is to optimize the pose by back propagation of the loss in eq. (6).

$$\mathcal{L} = \sum_{\mathbf{u} \in \mathcal{R}} \|\mathcal{I}\left(\mathcal{F}(TK\mathbf{u}; \theta)\right) - I(\mathbf{u})\|_2^2 \tag{6}$$

Truncated Dynamic Low-pass Filter: Typically, images are complex signals, which means it's easy to fall into local minima when using gradient descent with photometric loss. In the field of 2D image alignment, a very simple but effective operation is blurring the image first to make the signal smoother and ensure that local minimum can be stepped out. Similarly, [9] proposed the dynamic low-pass filtering as as shown in Fig. 3(a), where the frequency of positional encoding increases from zero to full linearly during iterations. It demonstrated that dynamic low-pass filter can suppress high-frequency signals to achieve the same effect as blurring images.

However, when a trained network is given, it's unnecessary to set the frequency threshold α start from zero, for it will lead to the network infer an invalid or even wrong RGB

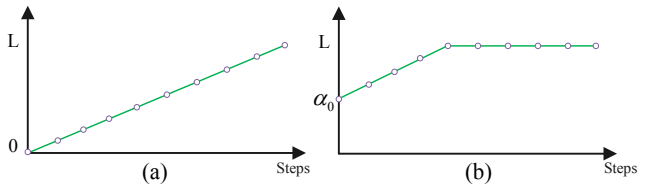

Fig. 3. (a) Dynamic Low-pass Filter. (b) Truncated Dynamic Low-pass Filter. Method (a) starts from zero which will lead to unreasonable inference results, on the contrary, method (b) ensures the validity of the inference information in the initial stage.

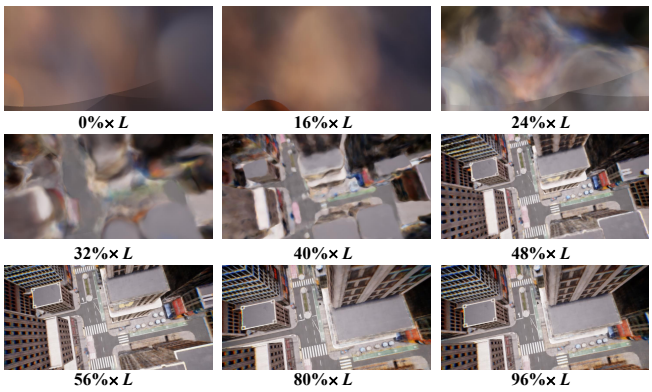

Fig. 4. Given a trained NeRF network, the above results are obtained by changing α from 0 to L dynamically during the inference. It shows that the images rendered before $\alpha = 24\% \times L$ contains invalid or even wrong color information, which will lead to pose optimization easily scattered.

results at the first place, as shown by the first line in Fig.4. Through experiments we found that the pose estimation diverge easily, due to the inference results contain a lot of invalid appearance. To solve the problem, our TDLF set α starts from α_0 to L as shown in Fig. 3(b), where α_0 is a non-zero number, and L is the full positional encoding frequency. By using TDLF, we can obtain more accurate color information at the early stage of optimization to ensure the proper convergence of the pose, while suppressing high-frequency signals as much as possible to avoid falling into local optima. Formally, the k-th positional encoding function we use is eq.(7), where the weight ω_k is calculated as in eq. (8) and α is the ratio of the current iteration to the total number of iterations.

$$\gamma_k(\mathbf{x}, \alpha) = \omega_k(\alpha) \cdot [\cos(2^k \pi \mathbf{x}), \sin(2^k \pi \mathbf{x})] \tag{7}$$

$$\omega_k(\alpha) = \begin{cases} 1 & k \leq (\alpha + \frac{\alpha_0}{L})L \\ 0 & k > (\alpha + \frac{\alpha_0}{L})L \end{cases} \tag{8}$$

V. Experiments and Results

A. Datasets

We evaluate our method against both our own Urban Minimum Altitude Dataset (UMAD) and Mill 19 [23] dataset. The UMAD is a virtual-scene dataset made by the simulator AirSim [18], which is built on top of the Unreal Engine. In order to ensure the simulation data is as close to real

as possible, on the one hand, we use a realistic city scene model which comes from Kirill Sibiriakov [20], on the other hand, we collect the drone trajectory data and image data separately to ensure the frequency and quality derived from methods in the paper [25]. In comparison to the other urban datasets [2] [10] which are created using the oblique aerial photography technique, our dataset has higher fidelity when it comes to the texture details.

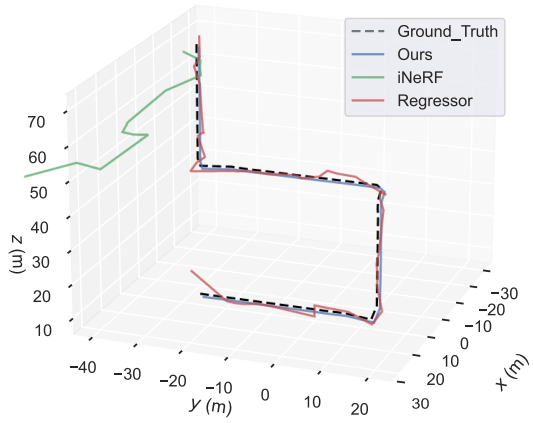

Fig. 5. Trajectory comparison on partial sequence of UMAD.

B. Implementation Details

We follow the architectural settings from Mega-NeRF [23], still, with several modifications for scene representation. The synthetic scene is divided into four parts. We train each part of the scene for 300,000 iterations. The submodule MLP consists of 8 layers of 256 hidden units and a final fully connected ReLU layer of 128 channels. We resize the images to 960 × 480 pixels and randomly sample 1024 pixel rays during the training steps. We use an Adam optimizer with an initial learning rate of 5×10^{-4} decaying exponentially to 5×10^{-5}. We discard the practice of adding Gaussian noise to the sigma output proposed by the original NeRF [12] because it harms the final results.

To train the place recognition model, we resize the rendered images to 480 × 240 pixels, then we use the CNN model to extract the features. An Adam optimizer is used, whose initial learning rate is 1×10^{-4} decaying exponentially to 2×10^{-5}. We set $\gamma = 250$ and $\beta = 0.8$ to balance the learning of location and orientation, real images and synthetic images.

In order to implement TDLF in the positional encoding layer, we add the parameter ω in eq. (8) to the network module of the layer. For less frequent changes of this parameter, we set its value every 50 steps during the pose optimization stage. The initial threshold α_0 is set to $40\% \times L$ in the experiments. Experiments are conducted on an NVIDIA RTX3090 GPU with 24GB of memory.

C. Global Localization

In place recognition stage, the image sequence from the UMAD is given into the trained regressor, outputting the initial pose values. In optimization stage, the initial poses are optimized using our method and iNeRF (in fairness, we replaced its MLP with mega-nerf, and the same was done in ablation study) respectively. As can be seen in Fig. 5, the prediction of regressor is within error tolerance of our pose optimization method, and the results are almost consistent with the ground truth, while iNeRF fails to converge. According to the quantitative evaluation results of UMAD scene in Table I, the pose error obtained from the regressor is within the range of $0 \sim 8\ m$, and our optimization method can further converges to the exact position. The average error is only less than 0.05 m, and the minimum error is even less than 0.01 m. The average error of iNeRF is up to 12.36 m, and the minimum error is also more than 5 m. Our method, compared to iNeRF, also performs better in the Mill 19.

TABLE I
QUANTITATIVE RESULTS ON GLOBAL TRANSLATION ERROR (M).

Scenes	Methods	Max	Mean	Min	Rmse	Std
UMAD	Regressor	7.03	1.69	0.34	2.07	1.20
	iNeRF	56.18	12.36	4.70	18.28	13.48
	Ours	**0.25**	**0.05**	**0.01**	**0.06**	**0.04**
Mill 19	Regressor	8.77	2.22	0.39	3.13	2.20
	iNeRF	33.81	17.52	19.42	20.43	10.51
	Ours	**0.15**	**0.11**	**0.06**	**0.11**	**0.02**

D. Ablation Study Of Optimization

Our state estimation has improved on the basis of iNeRF. On the one hand, we perform optimization on the tangent plane which attributes to the noisy photometric loss landscape over the SE(3) manifold; on the other hand, we introduce a TDLF to apply a smooth mask on the encoding at different bands (from low to high) over the course of optimization. Therefore, our ablation study will conduct a quantitative analysis of the error anti-interference ability and convergence accuracy of pose estimation according to the improvements. In the absence of TDLF and manifold-based optimization, our method is equivalent to iNeRF, just replacing the original NeRF with Mega-NeRF, which is more suitable for large scenes.

We carried out ablation experiments on the UMAD dataset, taking four different positions in the same configuration and recording their average results in Table II and Table III. The tables show that, our method has the largest tolerance for both translation and rotation perturbations, and the highest convergence accuracy. Meanwhile, it also proves the effectiveness of the TDLF in the NeRF-based state estimation method. Results in Fig.6 show that, in the initial phase of pose estimation, low-frequency contour information is used for registration, and high-frequency information is gradually used, which is similar to the idea of branch-and-bound. It guarantees a good way to escape from local minima generated by the high-frequency non-convex position encoding, while without TDLF, it is easy to fall into the local optimum. Besides, compared with the direct optimization on SE(3) in iNeRF, the optimization on the tangent plane

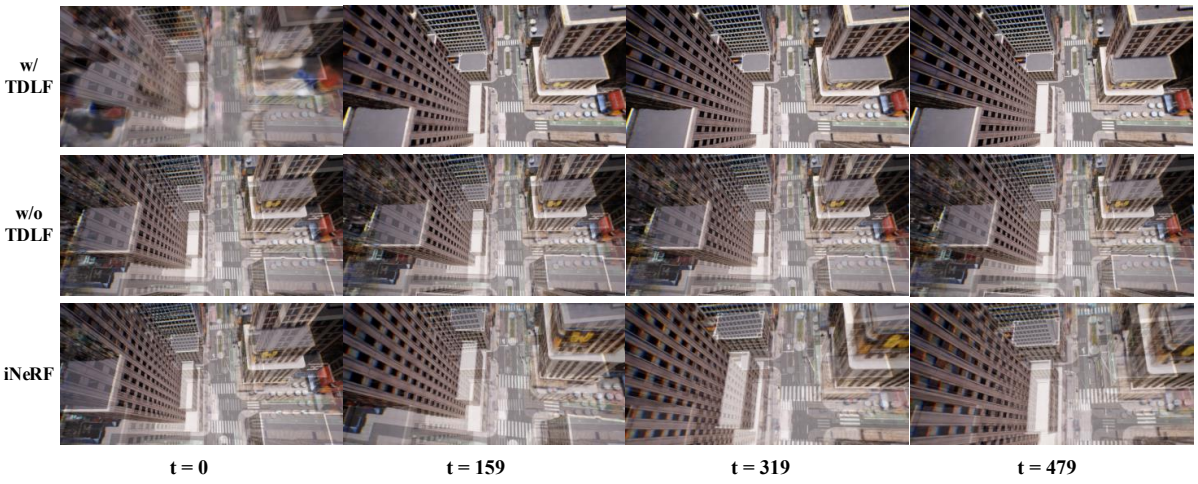

Fig. 6. We visualize the average results of rendered images in repeated experiments based on the estimated pose at iteration t and test image to compare our model with iNeRF. The utilization of the dynamic low-pass filter enables escaping from local minima, which is shown by the first line. The comparison of the third line and the second line implies that recursively optimizing on SE(3) instead of tangent plane leads to incorrect estimation results.

TABLE II

ABLATION STUDY WITH DIFFERENT INITIAL TRANSLATION ERROR.

Initial Error(m)	Manifold Optimization	TDLF	Translation Error(m)	Rotation Error(°)
4	×	×	0.19	0.69
	✓	×	0.83	0.21
	✓	✓	**0.02**	**0.10**
8	×	×	22.78	8.81
	✓	×	1.97	0.17
	✓	✓	**0.02**	**0.09**
12	×	×	7.14	4.30
	✓	×	2.10	4.47
	✓	✓	**0.03**	**0.10**
16	×	×	7.41	5.86
	✓	×	4.19	2.10
	✓	✓	**3.84**	**0.97**

TABLE III

ABLATION STUDY WITH DIFFERENT INITIAL ROTATION ERROR.

Initial Error(°)	Manifold Optimization	TDLF	Translation Error(m)	Rotation Error(°)
4	×	×	8.72	13.5
	✓	×	0.86	1.03
	✓	✓	**0.02**	**0.09**
8	×	×	20.95	17.00
	✓	×	3.18	5.45
	✓	✓	**0.02**	**0.10**
12	×	×	7.81	28.92
	✓	×	6.12	11.03
	✓	✓	**0.70**	**3.39**
16	×	×	9.99	19.74
	✓	×	6.12	17.67
	✓	✓	**3.24**	**4.80**

also shows its superiority. Additionally, when the translation error reaches 16m and the rotation error reaches 16°, the localization performance of our method also degrades due to the small overlap between the observed image and the rendered image at larger offsets leading to a local optimum.

E. Effect of Frequency Threshold

With different frequency threshold α_0, we have evaluated our method on both UMAD and Mill 19 dataset with a translation error of 8m and a rotation error of 8°, and Table IV shows that our method performs best around a threshold α_0 of $40\% \times L$. Here, the TDLF acts as if it were a BARF when α_0 is $0\% \times L$. The results indicate that TDLF optimizes for more accurate alignment compared to the baselines. This highlights the effectiveness of TDLF utilizing a coarse-to-fine strategy for localization.

TABLE IV

QUANTITATIVE RESULTS IN DIFFERENT DEGREES OF TRUNCATION ON THE WHOLE FREQUENCY.

Scenes	Error	0%	10%	30%	40%	50%	70%
UMAD	Trans(m)	200.07	81.18	0.04	**0.02**	0.03	0.06
	Rot(°)	34.10	50.24	28.49	**0.10**	0.18	0.20
Mill 19 Building	Trans(m)	0.69	0.18	0.09	**0.06**	0.07	0.10
	Rot(°)	7.92	4.83	1.97	0.20	**0.14**	3.24

VI. CONCLUSION

In this work, we propose a two-stage global localization mechanism under city-scale NeRF. A pose regressor is trained to provide an initial pose for a robot at arbitrary position. Besides, we introduce a TDLF for optimization to achieve coarse-to-fine pose registration. By conducting extensive experiments on both simulation and real-world dataset, our method, compared to recent works, is superior in both accuracy and tolerance of errors.

ACKNOWLEDGEMENTS

This work was sponsored by Tsinghua-Toyota Joint Research Fund (20223930097).

REFERENCES

[1] Michal Adamkiewicz, Timothy Chen, Adam Caccavale, Rachel Gardner, Preston Culbertson, Jeannette Bohg, and Mac Schwager. Vision-only robot navigation in a neural radiance world. *IEEE Robotics and Automation Letters*, 7(2):4606–4613, 2022.

[2] Anthony Brunel, Amine Bourki, Olivier Strauss, and Cédric Demonceaux. FLYBO: A Unified Benchmark Environment for Autonomous Flying Robots. In *3DV 2021 - 9th International Conference on 3D Vision*, pages 1420–1431, Virtual, United Kingdom, December 2021.

[3] Fernando Caballero and Luis Merino. Dll: Direct lidar localization. a map-based localization approach for aerial robots. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5491–5498. IEEE, 2021.

[4] Athanasios Chalvatzaras, Ioannis Pratikakis, and Angelos A Amanatiadis. A survey on map-based localization techniques for autonomous vehicles. *IEEE Transactions on Intelligent Vehicles*, 2022.

[5] Shuai Chen, Xinghui Li, Zirui Wang, and Victor Adrian Prisacariu. Dfnet: Enhance absolute pose regression with direct feature matching, 2022.

[6] Shuai Chen, Zirui Wang, and Victor Prisacariu. Direct-posenet: Absolute pose regression with photometric consistency. *CoRR*, abs/2104.04073, 2021.

[7] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[8] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[9] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[10] Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: the urbanscene3d dataset. In *ECCV*, 2022.

[11] 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.

[12] 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. *ECCV*, pages 405–421, 2020.

[13] Arthur Moreau, Nathan Piasco, Dzmitry Tsishkou, Bogdan Stanciulescu, and Arnaud de La Fortelle. LENS: localization enhanced by nerf synthesis. *CoRR*, abs/2110.06558, 2021.

[14] Barbara Roessle, Jonathan T. Barron, Ben Mildenhall, Pratul P. Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[15] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. *CoRR*, abs/1911.11763, 2019.

[16] Paul-Edouard Sarlin, Ajaykumar Unagar, Mans Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, et al. Back to the feature: Learning robust camera localization from pixels to pose. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3247–3257, 2021.

[17] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1744–1756, 2017.

[18] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics*, pages 621–635. Springer, 2018.

[19] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[20] Kirill Sibiriakov. Artstation page https://www.artstation.com/vegaart, 2022.

[21] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew Davison. iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[22] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P. Srinivasan, Jonathan T. Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. *ArXiv*, abs/2202.05263, 2022.

[23] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12922–12931, June 2022.

[24] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. *arXiv preprint*, 2022.

[25] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[26] 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.

[27] 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 *The European Conference on Computer Vision (ECCV)*, 2022.

[28] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021.

[29] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *CoRR*, abs/2010.07492, 2020.

[30] Xiangyu Zhang, Jianhua Zou, Kaiming He, and Jian Sun. Accelerating very deep convolutional networks for classification and detection, 2015.

[31] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.