# Tracking Registration Method Based on Absolute Pose Regression and Novel View Synthesis

Chengzhi Wei
*Qingdao University of Science and Technology*
Qingdao, China
1551290188@qq.com

Yan Feng
*Qingdao University of Science and Technology*
Qingdao, China
fywmh@163.com

2025 5th International Conference on Computer, Control and Robotics (ICCCR) | 979-8-3315-4354-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICCCR65461.2025.11072594

***Abstract*—With the development of computer vision technology, tracking registration methods have been further studied in several fields. In this paper, we propose a tracking registration algorithm combining absolute pose regression and new view synthesis, which firstly predicts the pose of a 6-degree-of-freedom camera relative to the scene from an image and extracts equivariant features, then synthesizes a new view compensating for the exposure changes, and finally optimizes the camera pose by backpropagation and direct matching with the input image. In the absolute pose regression network, the feature extractor accomplishes equivariant feature extraction from the data to further eliminate translation and rotation errors. In order to solve the problem of exposure fluctuation destroying photometric consistency, this paper proposes an integrated position-based coding strategy and temporal photometric consistent volume rendering in new view synthesis to improve the rendering quality. On the 7-Scenes dataset, the accuracy of this paper's method is significantly improved compared with other single-image absolute pose regression methods.**

***Keywords—tracking registration,absolute pose regression, novel view synthesis, deep learning***

## I. INTRODUCTION

3D structure-based approaches [1] are common solutions to achieve high-precision tracking registration by detecting and matching visual features in an image to recognize a 3D model of the environment, using local feature descriptors to match 2D-3D relationships, and using depth information or motion structure reconstruction to obtain robust geometric correlations[2]. However, the computational and storage requirements of such methods are usually high, especially when scaling these methods to large-scale scenes, requiring the computational task to be transferred to a remote server, which not only increases the latency and error, but also may raise serious privacy concerns. Methods based on 3D structures are computationally resource-intensive, which hinders their large-scale adoption on devices, and real-time inference cannot be easily achieved within the constraints of optimal accuracy.

Absolute pose regression is a deep learning based pose regression [3] method that uses end-to-end deep learning models capable of estimating device poses using single-frame monocular images. They provide fast device-side inference with minimal storage space even in large environments and multiple scenes. However, the lack of explicit understanding of geometric relationships prevents the models from fully utilizing geometric information to improve the accuracy and robustness of pose estimation in scenes with strong geometric constraints. In addition, in complex scenes, such as lighting changes, occlusion, and similar textures, the existing photometric-based methods cannot deal with photometric distortion, and the accuracy of absolute pose regression is also affected. Therefore, the current absolute attitude regression methods cannot meet the requirements of high-precision applications. To address the above problems, this paper proposes a method based on absolute pose regression and novel view synthesis, so that the accuracy of the existing absolute pose regression method and its performance in complex scenes can be significantly improved.

## II. RELEATED WORK

### *A. Absolute pose regression*

Absolute pose regression regresses the 6-DOF camera pose of an input image by training a deep neural network. The first absolute pose regression method was PoseNet[4], which directly regresses camera position and orientation given an input image by attaching an MLP header to the GoogleNet backbone. The resulting architecture, named PoseNet, allows for pose estimation using a single forward pass, although its accuracy is much lower than 3D-based methods. Subsequent APRs have seen a number of improvements, mainly related to the backbone architecture and the loss function. Shavit[5] proposed a random discard strategy for prediction across multiple models, and Brahmbhatt[6] proposed using a long-short-term memory layer to reduce the dimensionality of the global image coding to address the overfitting problem. Other work focuses on improving the loss function for absolute pose regression to improve position and orientation related errors.

Chen[7]extended the single scene paradigm of absolute pose regression for learning multiple scenes. Park[8] first trained an absolute pose regression network on unlabeled video sequences using an external VO algorithm.The unlabeled video frames were trained by adding pairwise geometric constraints between the frames using the VO algorithm.During inference, the pose map can be optimized and then processed using the pose map to further improve performance. Mildenhall[9]fine-tuned a pre trained network to improve accuracy. However, fine-tuning the neural network during testing is very time-

consuming and in practice it is difficult to obtain unlabeled test set data in advance with limited accuracy improvement during training.

### B. Novel view synthesis

Novel view synthesis is to generate new camera views based on scene image samples [10] and synthesize more data to extend the training space.Moreau [11] samples the scene boundaries and synthesizes virtual views using uniformly generated virtual camera poses, with limitations such as its expensive offline computational efficiency and lack of compensation for domain gaps between the synthesized and real images. Kellnhofer [12] proposed to utilize feature correspondence to obtain realistic pose regression training data, but relies on pre-computed 3D reconstructed maps.

Yenamandra[13]learns hybrid weights from retained real image data, which learns depth features present on top of reconstructed grids or dense depth maps, but is unable to simulate dynamic scenes over time because static scenes cannot be updated on the screen.Han[14] represents hierarchical scenes through predictive depth maps, but scaling up to higher-resolution images is subject to temporal and spatial complexity limitations.

## III. METHODOLOGY

In order to improve the feature representation capability of the tracking registration method, this paper proposes an equivariant feature-based pose regression network, which can explicitly encode the camera motion information in the feature space and is more adaptive to camera pose changes. In addition, in order to further adapt to complex scenes and deal with lighting changes, occlusion, similar textures, etc., this paper also proposes a novel view synthesis based on integrated position coding and temporal photometric consistency, which optimizes the camera pose by backpropagation and direct matching with the input image.

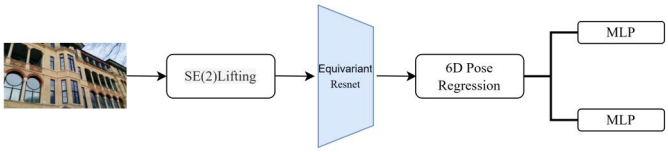


Fig. 1. Absolute pose regression network structure diagram

The algorithmic framework proposed in this paper is shown in Fig. 1, which contains three working modules: absolute pose regression network, new view synthesis and direct matching. The absolute pose regression network contains a rotationally translational isovariant ResNet backbone and two fully connected multilayer perceptrons to obtain the camera's pose with respect to the scene, and the isovariant features are utilized to improve the performance and generalization ability of the pose regression model in complex scenes. New view synthesis is then implemented using the improved NeRF to further process the output from the pose regression network, using an integrated position-based high frequency function to encode the relative relationship between coordinates and the position information, which helps the network to better understand the spatial structure of the input data and perform 3D reconstruction. Then equally or adaptively spaced samples are taken along the light and volumetric data is acquired at each sampling point, based on which volumetric rendering is performed and opacity and color synthesized images are computed. Finally the synthetic image is directly matched to the input image to optimize the camera pose by minimizing the photometric loss.

### A. Equivariant Features.

Equivariant features are special features that are related to the properties of data under specific transformations, exhibit unique properties when the data undergoes specific transformations, and are important in many tasks. Traditional APR methods are deficient in accuracy due to the lack of geometric information in classical convolutional neural network features. An equivariant feature is a feature whose feature extraction function satisfies a specific condition. A function f is said to be equivariant if it satisfies the condition that after the input data has been transformed g, its output features have also been transformed $g'$ accordingly.

When some kind of transformation is applied to the input data, the features obtained by the equivariant feature extraction function will also be transformed according to specific rules. Invariant features are a special case of equivariant features, i.e., when the transformation is a constant mapping, then the features are invariant features. For example, in image recognition, the invariant features do not change with the image rotation, scaling and other transformations, while the equivariant features will change with these transformations according to certain rules.

The absolute pose regression network in this paper is shown in Fig. 1, where the SE(2) Lifting operation is first performed to map the image from planar coordinates to a special Euclidean group, and the rotationally translationally isovariant ResNet is used for feature extraction, and the feature maps are subsequently fed into the 6D pose regression module, where the two fully-connected multilayer perceptrons lift the features to a higher dimensional space to obtain the rotational information and translational information, respectively. The equivariant ResNet is constructed by using equivariant convolution, pooling, normalization, and nonlinear operations, which allows the extracted features to have equivariant properties of rotation and translation, thus enabling better encoding of feature information. The absolute pose regression network function in this paper is shown in Eq.1:

$$L(x) = b + P \cdot E\big(F(x)\big)\#(1)$$

where x denotes the input image, F denotes the feature extractor, E denotes the nonlinear embedding, P denotes the linear projection, and b denotes the bias term.

### B. Integrated Position Encoding

This study utilizes an integrated location coding mechanism based on Gaussian kernel functions to achieve feature aggregation of local neighborhoods in 3D space through probability density weighting. This coding strategy breaks through the limitations of traditional deterministic position coding, and uniformly maps spatial coordinates and their uncertainty distributions to the frequency domain feature space. The specific implementation is divided into three key

stages: firstly, a Gaussian distribution model is established to describe the spatial probability density of the neighborhood of the target point, then the mathematical expectation of this probability distribution on the Fourier basis function is calculated, and finally, the feature level fusion is used to construct the frequency domain representation with noise-resistant characteristics. In order to realize the integrated position coding firstly, Eq. 2 is converted into Fourier features as shown in Eq. 3.

$$\gamma(x) = [sin(x), cos(x), ..., sin(2^{L-1}x), cos(2^{L-1}x)]^T \quad (2)$$

$$P = \begin{bmatrix} 1 & 0 & 0 & 2 & 0 & 0 & \cdots & 2^{L-1} & 0 & 0 \\ 0 & 1 & 0 & 0 & 2 & 0 & & 0 & 2^{L-1} & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & & 0 & 0 & 2^{L-1} \end{bmatrix}^T \#(3)$$

### C. *Volume rendering based on temporal photometric consistency*

To deal with occlusion and complex material cases, a common approach is to directly render the depth map and pass the image features to the surface specified by the depth map; however, the above method is more sensitive to the quality of the reconstructed resulting depth map. To improve the robustness, this paper proposes to sample the points on each pixel ray and calculate the transparency of each point based on the cumulative transmittance, as shown in Eq.4:

$$\widehat{\mathbf{C}}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma\big(\mathbf{r}(t)\big)\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt$$
$$\text{where} T(t) = \exp\left(-\int_{t_n}^{t} \sigma\big(\mathbf{r}(s)\big)ds\right). \#(4)$$

where $t_n$ and $t_f$ denote the lower and upper bounds of the integration of light rays, the positions where the light rays start and end their integration in the scene and T(t) denotes the transparency transfer function of the light rays from the point of view to the position, which describes the extent to which the light rays are absorbed and scattered in the propagation process.

In order to capture dynamic scenes, this paper extends the previously described static scenes by including time in the domain and modeling the 3D action as a dense flow field scene. For a given 3D point x and time i, the model predicts not only reflectivity and transparency, but also forward and backward 3D scene flow, which represent 3D offset vectors pointing to the x position at time i+1 and i-1, respectively, and the motion occurring between the observed time sequences is linear. In order to handle motion compensation in 3D space, compensation weights are also predicted. Thus, the dynamic model is defined as:

$$(\mathbf{c}_i, \sigma_i, \mathcal{F}_i, \mathcal{W}_i) = F_\Theta^{\text{dy}}(\text{x}, \text{d}, i) \#(5)$$

where c denotes the color phase value, σ denotes the transparency, F denotes the scene flow, W denotes the weight value, x denotes the point in 3D space, d denotes the view direction vector, and i denotes the time frame.

In the state of 3D scene streaming, to ensure that the scene at time i is consistent with the scene at the neighboring time j, this is done by volumetrically rendering the scene at time i, firstly from the camera's point of view at time i, and secondly, by transforming the scene from time j to time i in order to remove any motion that occurs from i to j. The scene at time j is then transformed from the camera's viewpoint at time i to time i in order to remove any motion that occurs from time i to j. During volumetric rendering, the predicted scene flow field is transformed along the light by transforming the position of each 3D sampling point , thus finding the RGB color and transparency at adjacent times. This results in a rendered image that transforms both camera and scene motion to time i.

$$\widehat{\mathbf{C}}_{j\to i}(\mathbf{r}_i) = \int_{t_n}^{t_f} T_j(t)\sigma_j\big(\mathbf{r}_{i\to j}(t)\big)\mathbf{c}_j\big(\mathbf{r}_{i\to j}(t), \mathbf{d}_i\big)dt \#(6)$$

$$\mathbf{r}_{i\to j}(t) = \mathbf{r}_i(t) + \mathbf{f}_{i\to j}\big(\mathbf{r}_i(t)\big) \#(7)$$

where $t_n$,$t_f$ denote the start and end times, T denotes the transparency transfer function, r denotes the light position, c denotes the color value, and d denotes the light direction.

## IV. EXPERIMENTS

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

### A. *7-Scenes Dataset*

The 7-Scenes dataset is a classic RGB-D dataset introduced by Microsoft Research Cambridge, which is mainly used to evaluate the performance of algorithms such as camera relocation, visual SLAM (Simultaneous Localization and Map Architecture), and 3D reconstruction in indoor scenes. The dataset contains seven different indoor scenes, each with unique structure and challenges. Each scene is provided with RGB images, depth maps, camera poses (6-degree-of-freedom poses), and in-camera references. RGB and depth images are provided at a resolution of 640×480 with a frame rate of 30 Hz using a Microsoft Kinect sensor (v1). The total data size is about tens of gigabytes and each scene contains multiple training and testing sequences.

### B. *Evaluation Metrics*

In this study, Median Translation Error (m) and Rotation Error (°) were used to assess the absolute attitude regression network performance.The translation error measures the deviation of the predicted position from the true position in a straight line distance in space. Median Translation Error, on the other hand, is the value in the middle of a set of samples after sorting the values of the translation error (if the number of samples is even, the average of the two middle values is taken). The median translation error can be used to determine how far the vehicle or robot position estimated by the system deviates from the actual position. The smaller the value, the higher the accuracy of localization. It avoids the excessive influence of a few extreme error values on the overall evaluation and more robustly reflects the general performance of the algorithm on position estimation .

In three-dimensional space, the true position coordinates are ($x_{gt}$, $y_{gt}$, $z_{gt}$) and the predicted position coordinates are ($x_{pred}$, $y_{pred}$, $z_{pred}$), the translation error d for this sample is expressed as shown in Eq.8:

$$d = \sqrt{(x_{pred} - x_{gt})^2 + (y_{pred} - y_{gt})^2 + (z_{pred} - z_{gt})^2} \#(8)$$

The rotational error is used to measure the difference between the predicted rotational attitude and the true rotational attitude and is usually expressed in units of angle (°). It describes how accurately the direction of rotation of an object in space is estimated . A smaller rotational error means that the system can more accurately control or estimate the direction of rotation of an object. Common calculations are based on rotational matrices, quaternions, and other representations. For example, in the calculation based on quaternions, the real rotation quaternion qgt and the predicted rotation quaternion qpred are first normalized, and then the angle between them is calculated as the rotation error. The specific formula is shown in Eq.9:

$$\theta = 2 \arccos\left(\left|q_{gt} \cdot q_{pred}\right|\right) \#(9)$$

## C. Analysis of Experimental Results

*a) Ablation Experiments:* This method performs ablation experiments on the 7-Scenes dataset and reduces the average median error metric by 8.3% and the average rotational error metric by 3.25% after utilizing an isovariant convolutional layer in place of the classical convolutional layer. Further analysis shows that the method achieves 11.5% and 8.63% accuracy improvement in structured scenes (Chess) and complex texture environments (Fire), respectively. With the novel view synthesis, it takes 300k iterations to converge (training time 28 hours), while the L2 loss of this method combined with the adaptive weighting strategy (translation term α=0.7, rotation term β=0.3) is optimal in 150k iterations (training time 14 hours), which is a two-fold improvement in the convergence speed. The ablation experiment confirms that the imbalance coefficient of the translation-rotation term exceeds 1:2 will lead to an increase in parameter sensitivity (condition number $\kappa > e^4$), whereas the balancing factor designed in this paper stabilizes the condition number of the Hessian matrix at $\kappa < e^3$, which effectively improves the optimization stability.

TABLE I. RESULTS OF ABLATION EXPERIMENTS

| | PoseNet | PoseNet+EF | PoseNet+NVS |
|---|---|---|---|
| Chess | 0.20m,13.11° | 0.18m,13.30° | 0.16m,13.18° |
| Fire | 0.31m,12.84° | 0.29m,12.63° | 0.27m,12.74° |
| Heads | 0.26m,8.59° | 0.21m,8.27° | 0.24m,8.36° |
| Office | 0.18m,5.74° | 0.16m,5.68° | 0.15m,5.53° |
| Pumpkin | 0.35m,13.27° | 0.33m,13.16° | 0.32,12.96° |
| Kitchen | 0.23m,9.76° | 0.21m,9.53° | 0.21m,9.31° |
| Stairs | 0.14m,12.15° | 0.12m,12.06° | 0.13m,12,09° |
| Average | 0.24m,10.78° | 0.22m,10.43° | 0.21m, 10.37° |

*b) Comparison Experiments:* In response to the network structure based on equivariant features and new view synthesis in this paper, the quantitative comparison results of multiple classes of pose estimation methods on 7-Scenes dataset covering PoseNet, LSTM PN, DSO, and MapNet are presented.Table Ⅱ statistically summarizes the key performance indexes of the different methods in each scene. The present method achieves a localization accuracy of 0.21m/5.3° in the Kitchen scene with a 13.6% improvement compared to the PoseNet geometrically constrained baseline, and maintains a localization accuracy of 0.27m/6.8° with a 7.3% reduction in the standard deviation in the Office scene with dynamic illumination interference. The experimental data show that the median translation error mean of this paper's algorithm is reduced by 4.36% and the rotation error mean is reduced by 2.51% compared to the best-performing MapNet, while the sequence-level error fluctuates in the range of 1.89 cm to 2.31 cm under dynamic illumination environments.

TABLE II. RESULTS OF COMPARISON EXPERIMENTS

| Scence | PoseNet | LSTM PN | DSO | MapNet | Ours |
|---|---|---|---|---|---|
| Chess | 0.35m,7.95° | 0.17m,5.49° | 0.23m,7.67° | 0.13m,2.53° | 0.11m,4.19° |
| Fire | 0.51m,14.5° | 0.26m,10.1° | 0.17m,25.4° | 0.22m,15.11° | 0.25m,9.5° |
| Heads | 0.31m,11.7° | 0.31m,12.45° | 0.42m,31.64° | 0.42m,11.6° | 0.31m,10.47° |
| Office | 0.49m,7.57° | 0.29m,6.4° | 0.24m,21.82° | 0.31m,8.47° | 0.22m,7.36° |
| Pumpkin | 0.46m,8.65° | 0.36m,8.36° | 0.56m,13.74° | 0.14m,6.52° | 0.16m,9.41° |
| Kitchen | 0.63m,8.42° | 0.47m,10.28° | 0.52m,11.58° | 0.39m,6.3° | 0.29m,7.53° |
| Stairs | 0.51m,15.1° | 0.25m,9.63° | 0.22m,15.62° | 0.29m,9.46° | 0.17m,5.64° |
| Average | 0.47m,10.56° | 0.3m,8.96° | 0.34m,18.21° | 0.24m,8.57° | 0.22m,7.73° |

*c) Visualization of comparison results*

The visualization results of the quantitative evaluation of camera positioning for the 7-Scenes dataset are shown in Fig. 2, with each subfigure containing the dual analysis dimensions of 3D spatial trajectory visualization and rotational error heat map. In the upper 3D coordinate system, the green trajectory characterizes the true camera position and the red trajectory shows the predicted path, and the spatial agreement between the two is quantified by the Hausdorff distance. The lower chromaticity bar uses a continuous color spectrum to map the rotational error (in degrees), where the warm regions indicate the time periods when the keyframe pose estimation deviates significantly.

In order to visually assess the visual performance of this paper's method, Figure 3 demonstrates the results of multiple comparison experiments. The visual comparison results show

that the proposed method maintains a high degree of consistency with the NeRF benchmark method in key visual dimensions such as colour fidelity, edge sharpness and texture details. In the complex reflection region and high-frequency texture region, the method maintains the high fidelity characteristics of the synthesized image without visible colour shifts or loss of details. Quantitative analysis shows that the difference between the two methods in local structural similarity (SSIM local window metric) is less than 0.015, which verifies the effectiveness of the present method in maintaining the view synthesis quality.

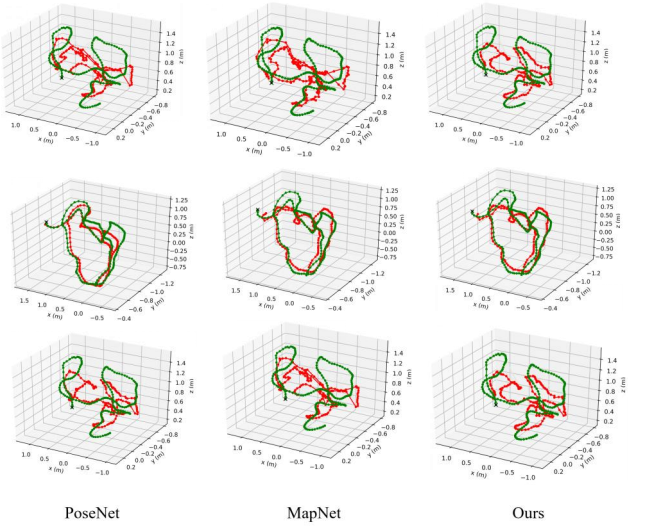


Fig. 2. Visualization results

Fig. 3. Comparison of real image(U) and rendered image(D)

## V. CONCLUSION

In this paper, we propose algorithms that combine absolute pose regression networks and new view synthesis to provide better performance while maintaining a low inference cost. Isotropic features are introduced into the absolute pose regression model, and the equivariant features can provide richer geometric representations, which helps to improve the performance and generalization ability of the pose regression model in complex scenarios. The use of integrated position coding in NVS provides more information for the network to capture high-frequency variations, and the introduced loss of temporal luminosity consistency reduces the rendering error of static regions in dynamic scenes and improves the rendering quality of the new view synthesis system. Finally, the photometric difference between the real and synthesized images is obtained by direct matching and used to constrain the parameters of the absolute pose regression network.

## REFERENCES


[1] Liangchen Song, Gang Yu, Junsong Yuan, and Zicheng Liu.Human pose estimation and its application to action recognition: A survey. Journal ofVisual Communication and Image Representation, 76:103055, 2021.

[2] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Gdrnet: Geometry-guided direct regression network for monocular 6d object pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 16611–16621, June2021.

[3] Chao Zhang, Ignas Budvytis, Stephan Liwicki, and Roberto Cipolla.Rotation equivariant orientation estimation for omnidirectional localization. In Proceedings ofthe Asian Conference on Computer Vision (ACCV), November 2020.

[4] Osman Semih Kayhan and Jan C. van Gemert. On translationinvariance in cnns: Convolutional layers can exploit absolute spatial location. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

[5] Shavit Y, Ferens R, Keller Y. Learning multi-scene absolute pose regression with transformers[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 2733-2742.

[6] Brahmbhatt S, Gu J, Kim K, et al. Geometry-aware learning of maps for camera localization[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 2616-2625.

[7] Chen S, Li X, Wang Z, et al. Dfnet: Enhance absolute pose regression with direct feature matching[C]//European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2022: 1-17.

[8] Park K, Sinha U, Barron J T, et al. Nerfies: Deformable neural radiance fields[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5865-5874.

[9] Mildenhall B, Srinivasan P P, Tancik M, et al. Nerf: Representing scenes as neural radiance fields for view synthesis[J]. Communications of the ACM, 2021, 65(1): 99-106.

[10] Kim S, Kim I, Vecchietti L F, et al. Pose estimation utilizing a gated recurrent unit network for visual localization[J]. Applied Sciences, 2020, 10(24): 8876.

[11] Moreau A, Piasco N, Tsishkou D, et al. Lens: Localization enhanced by nerf synthesis[C]//Conference on Robot Learning. PMLR, 2022: 1347-1356.

[12] Kellnhofer P, Jebe L C, Jones A, et al. Neural lumigraph rendering[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 4287-4297.

[13] Yenamandra T, Tewari A, Bernard F, et al. i3dmm: Deep implicit 3d morphable model of human heads[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 12803-12813.

[14] D. Han, J. Ryu, S. Kim, S. Kim, J. Park, and H.-J. Yoo, “Metavrain:A mobile neural 3-d rendering processor with bundle-frame-familiarity-based nerf acceleration and hybrid dnn computing,” IEEE Journal of Solid-State Circuits, 2023.