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

Multimodal Neural Radiance Field

Haidong Zhu[1] Yuyin Sun[2] Chi Liu[2] Lu Xia[2] Jiajia Luo[2] Nan Qiao[2] Ram Nevatia[1] Cheng-Hao Kuo[2]

***Abstract*— This paper addresses the challenge of reconstructing a scene with a neural radiance field (NeRF) for robot vision and scene understanding using multiple modalities. Researchers have introduced the use of NeRF to represent an object for synthesizing and rendering novel views of complex scenes by optimizing a 3-D radiance field for ray casting and rendering for 2-D RGB images. However, using RGB images alone introduces additional geometry ambiguities with transparent objects or complex scenes and cannot accurately depict the 3-D shapes. We discuss and solve this problem and use multiple modalities as input for the same NeRF model to build a multimodal NeRF by incorporating point clouds and infrared image supervision to prevent such bias. In contrast to RGB images, infrared images and point clouds are typically taken by separate cameras that cannot be aligned with the RGB camera. We further introduce the alignment of different modalities based on point cloud registration to estimate the relative transformation matrices between them before training a NeRF model with multiple modalities. We evaluate our model on chosen scenes from the ScanNet and M2DGR datasets and demonstrate that it outperforms existing state-of-the-art methods.**

I. INTRODUCTION

As one of the state-of-the-art methods for novel view synthesis, Neural Radiance Field, abbreviated as NeRF [1], generates a radiance field storing the density and RGB color value for each point in the 3-D space. Compared with other volume rendering methods, NeRF can generate photographic rendering by ray casting for novel view synthesis and assist the robot to understand the scene in the 3-D space with 2-D images. However, the only input modality for most of the existing NeRF networks is the RGB images and their processed outputs, such as segmentation maps, which heavily rely on the quality and quantity of the input RGB patterns. In addition, when encountering transparent or complex geometry surfaces, inaccurate depth estimations from RGB images introduce extra ambiguities.

In this paper, we introduce the use of multiple modalities as input for the NeRF model. Different modalities can help the model find more geometrical information for reconstructing the 3-D radiance field. We show an example in Fig. 1 for two rendered results of two different point clouds which describe the same room. While the rendered images from viewpoint 1 are reasonable for both scenes, the lack of geometric accuracy makes the rendered image from viewpoint 2 suffer, as shown in Fig. 1 (b). If RGB is the only input modality and fails to provide enough variances for understanding the global geometry, models may generate

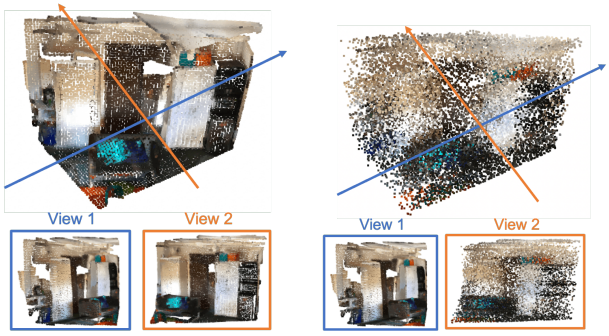


Fig. 1. Rendered results for two point clouds after ray casting. Even though these two point clouds are different in the 3-D space, the 2-D rendered results are the same for some given viewpoints (View 1), while rendered images are of different quality for novel viewpoints (View 2).

different 3-D radiance fields in the density function. Even if their 2-D rendered images are the same as the examples in the training set, the prediction of the final 3-D radiance field may be very different from the groundtruth scene, making the reconstruction of the novel view inaccurate. However, this situation will be ameliorated if we include other modalities such as point clouds or infrared images to assist the reconstruction since they contain global or local geometry information in such modality, which is not included in RGB images. With different modalities as input, the single NeRF model can construct better models by finding the strengths of each modality and combining them.

To use different modalities for the same NeRF, we need to align these modalities in the same coordinate system. Recently some research [2], [3] has focused on using modalities, such as semantic labels, in addition to the input of RGB images. Modalities directly captured from external sensors that can improve the geometry accuracy, such as point clouds from LiDAR scans or infrared images from infrared cameras, are unlikely to be perfectly aligned due to the different coordinate systems of the sensors with RGB cameras. Before fusing them with the RGB images, aligning the coordinate frames provided by different cameras and sensors is needed for the network to take all the modalities as input for generating the scene with both precise RGB patterns and geometry information.

To align different input modalities, we introduce using the point cloud registration for the 3-D scratches extracted from different modalities for estimating the corresponding transformation matrices between them. For 2-D inputs, such as RGB images and infrared images, we extract the point cloud via depth estimation for each frame and reproject the

[1]Haidong Zhu and Ram Nevatia are with the Department of Computer Science, University of Southern California. haidongz@usc.edu
[2]Yuyin Sun, Chi Liu, Lu Xia, Jiajia Luo, Nan Qiao and Cheng-Hao Kuo are with Amazon Lab126, USA.

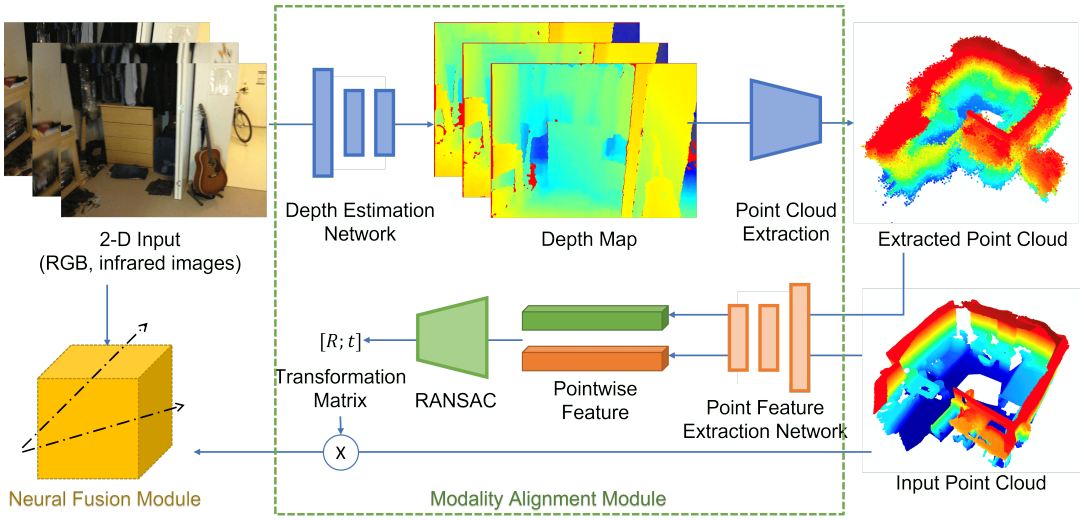


Fig. 2. Architecture of the proposed method. In addition to 2-D inputs, we introduce using point cloud as an example to supervise the generation of geometry accuracy. The concatenations of three rectangles are trainable models, while trapezoids are not trainable modules.

points back to 3-D space from 2-D images. After aligning different coordinate systems for different modalities, we utilize the geometry modalities, including point clouds from LiDar scans and infrared images from infrared cameras, to supervise the generation of density and geometry information for the radiance field in addition to the supervision of the reconstructed images. The supervision provided by the geometrical modality can ensure the correctness of the actual shape in the 3-D space along with the rendered RGB images for its appearance. We assess our results on two room-level public datasets, ScanNet [4] and M2DGR [5]. We outperform some of the state-of-the-art methods for 3-D room-level point rendering and 2-D image synthesis.

In summary, our contributions are as follows: 1) we build the first pipeline for using different modalities as the input for the same NeRF model with varying modalities of input; 2) we introduce the use of the point cloud based on the depth estimation for camera coordinate alignment between different modalities for alignment between different modalities, and 3) we introduce supervision for the generation of density fields for NeRF with geometrical information.

II. RELATED WORK

In this section, we discuss some recent progress in the Neural Radiance Field and the development of the system with multi-sensor fusion and multi-task learning.

Neural Radiance Field. To render the image of a 3-D object or scene, NeRF [1] introduces building a cubic neural radiance field using 2-D RGB images from different viewpoints with RGB color values and density for each point in the cube. NeRF first determines the ray projected on each pixel when rendering the image from a specific viewpoint. After that, for each ray, it predicts the densities and corresponding RGB values for all the points on this ray and sums them up after reweighting. By constructing this 3-D radiance field, NeRF can generate the projection for any selected camera viewpoint. Based on NeRF, researchers have developed different methods for image and video-related tasks, such as video synthesis and animation [6], [7], [8], [9], model reconstruction [10], [11], etc. However, these methods only use RGB images and their processed results as input, which lack guidance from geometry modalities. Recently some researchers [12] have developed NeRFs generating multimodal output. However, these methods are still restricted to the limited camera viewpoints from the RGB images and cannot use the external captured modalities to supervise its generation.

Multi-sensor fusion and multi-task learning. Multi-sensor fusion has become increasingly popular in 3-D scene understanding and recognition tasks. There are point-level fusion methods, such as [13], [14], [15], [16], [17], which attach the predicted features to the points generated from LiDAR scans and perform the 3-D understanding task on the point-level reconstruction. In addition, there are also some proposal-level fusion methods, such as [18], [19], [20], [21], [22], [23], which focus on the object-centered detection results and are applied to each independent proposal. The model can use different sensor inputs to capture appearance and geometry information by capturing features from other input modalities. With modalities from various sensors, recently, researchers have also developed multi-task learning methods [24], [25], [26], [27] based on different modalities. By learning from different downstream tasks with varying modalities of input, the model can make the most use of the input modalities and combine their strengths with supervision for appearance and geometry.

III. METHOD

In this section, we discuss the detailed design of our model. We show our model architecture in Fig. 2, which consists of two primary submodules: a modality alignment module and a neural fusion module. The modality alignment module takes the input from different modalities and aligns different coordinate systems used by different sensors to

match them into the same coordinate system. The neural fusion module fuses the information from different modalities and generates results for each modality for training. In the remaining of this section, we first review the details of NeRF in Sec. III-A. After that, we will discuss modality alignment and neural fusion modules in Sec. III-B and III-C.

A. Neural Radiance Field

The Neural Radiance Field, abbreviated as NeRF, is a implicit volumetric space describing the continuous value for density and color for the object or scene it represents. For a pixel on the rendered image, with its viewing direction annotated as d, we find the corresponding ray r that go through the field and project this pixel. After that, we locate the points $x \in r$ where r intersects the radiance field and predict the density $\sigma(x)$ and RGB value $c(x)$ for these points by using an MLP network $\mathbf{F}$ following $\sigma(x), c(x) = \mathbf{F}(x, d)$. With the density and RGB values for all the points x on the ray r, we reweight and sum them up to predict the final RGB value of the point $\tilde{C}(r)$

$$\tilde{C}(r) = \sum_{p=1}^{m} exp(-\sum_{q=1}^{p-1} \sigma_q \delta_q)(1 - exp(-\delta_p \sigma_p))c(p) \quad (1)$$

with a random set of quadrature points $\{h_p\}_{p=1}^{m} \in [h_n, h_f]$ following [1], where h_n and h_f represent the near and far bound of the sampled points. In the equation, δ_p represents the distance between two neighbour quadrature points h_p and h_{p+1}. To train the model $\mathbf{F}$, NeRF calculates the L-2 distance between generated RGB value $\tilde{C}(r)$ with the groundtruth RGB value $C(r)$. According to [28], since the neural network will underfit the high-frequency patterns, we follow [1] to use a position encoding $\gamma(x)$ for projecting the input point x to the high dimension space following

$$\gamma(x) = (sin(2^0 \pi x), cos(2^0 \pi x), ..., sin(2^{L-1} \pi x), cos(2^{L-1} \pi x)) \quad (2)$$

where each point x is normalized to [-1, 1]. We use the three normalized coordinates of the point x with L as 10 and three of the Cartesian viewing direction unit vector for d with L as 4 following [1].

B. Modality Alignment Module

Since the input modalities are captured from different sensors with different camera coordinate systems, we need to align them into the same coordinate system to fuse the information in different modalities. We first convert input data into point cloud modalities via depth estimation and point cloud extraction, followed by the second step of point cloud registration to predict the transformation matrices. We discuss these two steps below.

1) Point Cloud Extraction: For our system, there are two types of input signals from different sensors: 3-D signals, such as point clouds P_L from LiDAR scans, and 2-D images, such as infrared and RGB images, $\{I_{ir}\}$ and $\{I_{rgb}\}$. Compared with the projected images in the 2-D space, 3-D modalities preserve the original geometry information that can help us reconstruct the relative relationship between different modalities. Since we are matching room-level reconstruction results, the point cloud can be subsampled for lighter calculations while preserving its geometry features, making it the best representation for us to estimate the transformation matrix.

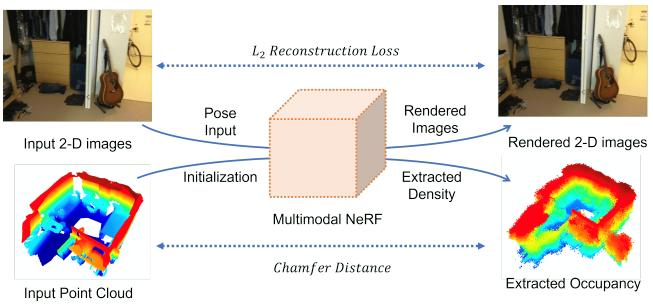


Fig. 3. The neural fusion module. We use a multimodal NeRF for encoding different 2-D input modalities and simultaneously supervise the generation of the density, which is represented as point clouds.

For the two sets of 2-D images, $\{I_{ir}\}$ and $\{I_{rgb}\}$, we first estimate the relative transformation matrix using COLMAP [29], [30] for the images in the same modality. In this way, we can compute the corresponding position for each image in the given modality to build their coordinate system. With known camera poses for each image in their modality, we can convert the points on each image back to 3-D space and combine them into the complete point cloud following

$$\begin{aligned} P_{i,n} &= \sum_{x \in X_{i,n}, y \in Y_{i,n}} Cam2World(x, y, D(x, y, I_{i,n})) \\ P_n &= \sum_i P_{i,n}, i \in \{1, 2, ..\}, n \in \{rgb, ir\} \end{aligned} \quad (3)$$

where i is the frame number for the modality n. X and Y are the positions for the pixels in the input image $I_{i,n}$ $D(x, y, I)$ is the depth estimation network for estimating the depth the corresponding point (x, y) in image I. By combing all the points from each frame together, we generate the point clouds P_{ir} and P_{rgb} for the corresponding modalities I_{ir} and I_{rgb}.

2) Point Cloud Registration: With generated point clouds P_{ir} and P_{rgb} from 2-D images along with the input point cloud P_L, we compute the transformation matrix following

$$T_{m,n} = RANSAC(F(P_m), F(P_n)), m \neq n \quad (4)$$

where m and n are two input modalities and $F(\cdot)$ is the feature extraction network. After using RANSAC to align different modalities, we can estimate the relative transformation matrix $T_{m,n}$ between P_m and P_n. In this way, we can convert P_n to $T_{m,n}P_n$, which makes it in the same coordinate system as P_m after alignment.

C. Neural Fusion Module

With the modalities for different inputs aligned into the same camera coordinate system, our next step is to fuse these different modalities into the same field to combine them. We show the neural fusion module in Fig. 3. To fuse

these different modalities, we introduce two branches: the reconstruction branch for 2-D inputs, which are RGB and infrared images, and the regression for the object density in the 3-D space for the supervision of density.

For the reconstruction of the RGB and infrared images, we follow Eq. 1 to predict the rendered images for the two 2-D modalities with corresponding camera viewpoints. For decoding two different branches, we use the shared-MLP for the first few layers before using different layers for the last few layers to decode their corresponding features. During training, we introduce two different reconstruction losses, L_{rgb} and L_{ir} for the reconstruction of RGB images and IR images, respectively, which are defined as

$$\begin{aligned} L_{rgb} &= \sum_{r \in R} ||\tilde{C}_r(r) - C_r(r)||_2 \\ L_{ir} &= \sum_{r \in R} ||\tilde{C}_i(r) - C_i(r)||_2 \end{aligned} \tag{5}$$

where $\tilde{C}_i(\cdot)$ and $\tilde{C}_r(\cdot)$ are the predicted infrared and RGB value for the corresponding ray. R is the collection for all the rays go through the pixel and $C_i(\cdot)$ and $C_r(\cdot)$ are the corresponding groundtruth values. In this way, we can use RGB and infrared images to supervise the same NeRF model.

In addition to supervising the generation of the reconstructed infrared and RGB images, we also supervise the generation of density in the radiance field. This can ensure the points are of good quality, reflect the real case of the scene rendered in the field, and restrict non-reasonable results for the occupancies. We follow PointNeRF [31] for generating the point field as the real occupancy of the scene the radiance field is rendering and use the chamfer distance to supervise the generation of the point cloud along with the original points following

$$\begin{aligned} L_{CD}(S_1, S_2) = &\sum_{x \in S_1} \min_{y \in S_2} ||x - y||^2 \\ &+ \sum_{x \in S_2} \min_{y \in S_1} ||x - y||^2 \end{aligned} \tag{6}$$

where S_1 and S_2 are two sets of points, which are the groundtruth points and generated prediction in our experiment. Our final loss is used as

$$L = \alpha L_{rgb} + \beta L_{ir} + L_{CD} \tag{7}$$

where α and β are two activation functions. Since the infrared RGB images are from two different cameras, it is likely that for some camera viewpoints, only one modality is available while the other is not. These two activation values are 1 when such modality is available at the camera viewpoint while keeping to 0 when the corresponding camera viewpoint for this angle does not exist.

IV. Experiments

In this section, we discuss the details of our experiment. We first discuss the datasets we use in our experiment in Sec. IV-A, followed by the implementation details of our experiment in Sec. IV-B, and finally, we discuss the baseline methods for comparison as well as the metrics we use in our experiment in Sec. IV-C.

A. Dataset

In our experiment, we assess our model with two different datasets, ScanNet and M2DGR. Both datasets provide at least two different input modalities: a sequence of RGB images for the scanned room and the processed point cloud.

ScanNet [4] is a room-level dataset with sequences of RGB images captured with hand-holds cameras. In addition to the RGB images, ScanNet also provides the room-level mesh result as the second modality, which we use to supervise the generation of the density map and occupancy. In our experiment, we choose three different rooms to reconstruct with scene ID 0000, 0072, and 0101. For the scenes of each ScanNet room, we select at most 1,000 frames as the image set. We sample one from every five frames for training and use the remaining frames for evaluation.

M2DGR [5] is a dataset provided with several scanned sequences for the various scenarios in real-world environments with a rich pool of sensory information, including vision, LiDAR, IMU, etc. In addition to the RGB images, M2DGR also provides thermal-infrared images, which capture the 2-D geometry information in greyscale images with infrared sensors. In our experiment, we use the scans for the hall as our experiment setting. For the selected M2DGR scene, we select 165 infrared images along with 280 RGB images as 2-D input for our network. We randomly select 4/5 of the images for training and the remaining 1/5 as the test set for both modalities in the experiment.

B. Implementation Details

To extract depth from 2-D images for building the point cloud, we use MVSNet [32] as our depth estimation network $D(\cdot)$ for I_{rgb}. Since a more accurate transformation matrix relies on the accuracy of the depth prediction, we pretrain the model in on DTU dataset [33] to gain more knowledge for depth estimation. For I_{ir}, due to the lack of groundtruth and pretrained models, we follow [12] to use sparse annotated depth estimation for training $D(\cdot)$ for dense depth estimation.

For point cloud registration, we use the GeDi descriptor [34] pretrained on the 3DMatch dataset [35] dataset for feature extraction F in Eq. 4. GeDi [34] is a latest state-of-the-art method on several point cloud feature extraction datasets, including 3DMatch [35], ETH [36] and KITTI [37], and it has the best performance among all of these descriptors for point cloud registration in the comparison with other point cloud registration methods such as DIP [38] and SpinNet [39]. To achieve this, we subsample 10,000 points from each extracted point cloud and use GeDi to compute their local and global features. After that, we apply the GeDi descriptor [34] for the feature extraction for the subsampled points on each point cloud and estimate their relative transformation matrices with RANSAC for point-to-point distance with scaling estimation enabled.

With the pretrained models for registration between different modalities, we follow PointNeRF [31] to build the

Methods	SSIM (↑)	PSNR (↑)	RMSE (↓)
NeRF [1]	0.879	28.02	0.041
PointNeRF [31]	0.902	29.10	0.035
Ours	**0.911**	**29.97**	**0.033**

TABLE I

NUMERICAL RESULTS ON SELECTED SCANNET SCENES FOR USING RGB IMAGES WITH POINT CLOUDS COMPARED WITH NERF [1] AND POINTNERF [31]. (↑) INDICATES HIGHER VALUES ARE BETTER, WHILE (↓) SHOWS LOWER VALUES ARE BETTER.

Methods	SSIM (↑)	PSNR (↑)	RMSE (↓)
NeRF [1]	0.795	24.52	0.059
PointNeRF [31]	0.817	25.80	0.056
Ours (RGB + IR)	0.820	25.92	0.052
Ours (RGB + PC)	0.828	26.05	0.049
Ours (All)	**0.831**	**26.10**	**0.047**

TABLE II

NUMERICAL RESULTS ON M2DGR HALL 01 FOR RECONSTRUCTED RGB IMAGES COMPARED WITH NERF [1] AND POINTNERF [31]. MODALITIES IN THE PARENTHESES FOR OUR METHODS ARE USED FOR TRAINING, WHERE 'IR' IS THE INFRARED IMAGES AND 'PC' REPRESENTS POINT CLOUD, WHILE 'ALL' IS TO USE ALL THREE MODALITIES, RGB, IR AND POINT CLOUD.

NeRF model for radiance field construction. In addition to the decoder of the RGB images, we also introduce an independent branch of the same architecture as the RGB branch to decode the point-level feature of the greyscale infrared images. We have a four-layer MLP with 256 as dimensionality for the hidden feature for each branch. During training, if the modality is available for some image for the branch, we backpropagate the loss to the feature for the PointNeRF and optimize the pointwise feature embeddings.

During training, we tune our multimodal NeRF network for 200,000 iterations for each scene in our experiment. We set the learning rate as $5e^{-4}$ with an Adam optimizer [40] and normalize the prediction of RGB values to [0,1] for every channel by clamping the predicted output from the network.

C. Metrics and Baselines

We use three metrics during inference: RMSE, PSNR, and SSIM. For comparison, we choose PointNeRF [31] in addition to the original implemented NeRF [1] as our baseline methods as we build our model based on these two methods. Since we are proving that multimodal input helps generate a more precise NeRF and the lack of similar methods using the same modalities, we do not include other NeRF-related methods in our comparison.

V. RESULTS

In this section, we present our results on the two datasets in our experiment with both quantitative and qualitative comparisons. In addition, we present some further ablations about the different modalities used in the experiment to assess the performance of different modalities used as input in our experiment.

Methods	SSIM (↑)	PSNR (↑)	RMSE (↓)
NeRF [1]	0.881	28.03	0.031
PointNeRF [31]	0.907	28.27	0.026
Ours (IR + RGB)	0.906	28.28	0.026
Ours (IR + PC)	0.908	28.30	0.025
Ours (All)	0.911	28.33	0.024

TABLE III

NUMERICAL RESULTS ON M2DGR HALL 01 FOR RECONSTRUCTED INFRARED IMAGES COMPARED WITH NERF [1] AND POINTNERF [31]. MODALITIES IN THE PARENTHESES FOR OUR METHODS ARE USED FOR TRAINING.

A. Numerical Results

For the numerical results, we first present the quantitative performance on two datasets and then include some ablation studies for the effect of different modalities.

1) Results on ScanNet Scenes: We first show our average performance on the selected ScanNet scenes in Table I. We compare our method with two baseline methods, basic NeRF implementation[1] and PointNeRF[2]. Our proposed method outperforms the other two methods with large margins for all three criteria we use. This shows that, with the assistance of external input from other modalities for the supervision of the geometric generation, the model can capture a more accurate scene for the final generation of the rendered images.

2) Results on M2DGR Scenes: In addition to the results for ScanNet, we show the numbers for the M2DGR in Table II and III. We compare our methods with NeRF and PointNeRF for the room we render for both infrared and RGB image synthesis. We note that the multimodal NeRF still outperforms the other two methods. For the RGB image synthesis, the model cannot extract perfect geometry information for the transparent surface and fails to give precise geometry information for rendering from novel camera viewpoints with only RGB images. By introducing either infrared images or point clouds, we have external geometry supervision that helps build a more accurate density prediction network and shows better-rendered images. In addition, global geometry supervision with point clouds can also help the local geometry modalities (infrared images) construct a more accurate local shape pattern with global awareness.

3) Ablation results: For the ablation studies, we investigate the impact of different modalities and their effects on the final rendered images. We show the results on the M2DGR dataset in Table II for RGB image reconstruction and III for infrared image reconstruction since M2DGR provides both infrared images in addition to the point clouds from LiDAR scans for geometry supervision.

[1]https://github.com/yenchenlin/nerf-pytorch
[2]https://github.com/Xharlie/pointnerf

NeRF [1] Ours Groundtruth

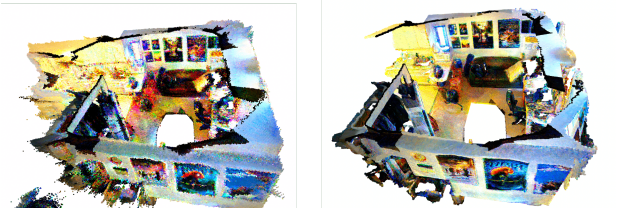
Fig. 4. Visualization of rendered scenes in selected ScanNet rooms. Images in the first, second, and third columns are generated or selected from NeRF, our method and groundtruth, respectively. For patterns like bike and clothes, original NeRF fails to construct a clear rendered results from novel viewpoint, while ours can give a clear shape for each pattern.

From the Tables, we note that if we use one of the geometry information inputs along with the RGB images, the network can already show better results than RGB images as the only input modality. Using point clouds performs better than infrared images, while the model is the best with both of the modalities. Point clouds have more global and room-level geometry information than infrared images. In contrast, infrared images focus more on frame-wise results and details geometry information for the final rendered room reconstruction. We can achieve the best performance by providing both local and global geometry information in addition to the RGB frames. We also note that the introduction of geometry input has greater help on RGB reconstruction than infrared image reconstruction. Since infrared images only include local geometry and shape information, RGB images do not contribute much to the rendering results for infrared images due to the lack of shape and geometry information not affected by RGB patterns. Point clouds, however, slightly improve the quality of the rendered infrared images with more global geometry information.

B. Visualization Results

For visualization results, we show 2-D rendered RGB images along with the projected 3-D rendered points with the given point cloud input in Fig. 4, 5 and 6 respectively. The visualization of the 2-D images can help us verify the final render quality, while the visualization of the projected 3-D point clouds from the two different levels of scales can help us assess both density estimation and RGB prediction for each point in the 3-D space. For both settings, we show the results on the ScanNet dataset since the rooms described in ScanNet have more indoor objects for rendering.

For the 2-D rendered images in Fig. 4, we compare our method with the NeRF with RGB images as input. We note that for the patches where NeRF shows blurred rendered patterns, our method can preserve a clear boundary and show more details. For example, in the first example of the first column, NeRF cannot reconstruct a precise shape for the bike in the scene. However, with more geometrical information, the rendered results with multimodal NeRF are much clearer, with shapes that are easily recognizable.



(a) PointNeRF [31] (b) Ours

Fig. 5. Comparison for the room-level rendered results between PointNeRF [31] and our method.

(a) PointNeRF [31] (b) Ours

Fig. 6. Results for point rendering of the partial room scene compared with PointNeRF [31].

In addition to the 2-D rendered results, we compare our 3-D point cloud rendering for the whole room in Fig. 5. We choose ScanNet room 0101 for the final comparison with PointNeRF, which also outputs the point-wise room reconstruction. We note that the point cloud generated from PointNeRF includes many artifacts and error predictions without the assistance of geometrical supervision. For the patterns that cannot be visually distinguished with PointNeRF [31], for example, the images on the wall, our method with multimodal NeRF can render clear shapes and patterns for all these local patterns in the point cloud of the space.

Moreover, we zoom into a small corner of the room as Fig. 6 to visualize the local patterns for the rendered point cloud. From the two rendered corners of the room, we cannot distinguish the detailed patterns for PointNeRF. For example, for books on the shelf, we cannot easily separate them from each other with the rendered point cloud from PointNeRF, while our multimodal NeRF shows clear boundaries between different books with the rendered 3-D points. With better and more precise geometry supervision during the generation of the radiance field in the training of the NeRF, we can generate more accurate rendered results in the 3-D space and preserve more fine-grained details.

VI. CONCLUSION

In this paper, we propose using point cloud registration to align the different input modalities after lifting all of them into 3-D space to train a multimodal NeRF. In addition to the RGB images, which are widely used in NeRF, we supervise the generation of the density stored in NeRF with point clouds and infrared images to ensure the precision of the volume density. We assess our method on three modalities, RGB images and point cloud and infrared images, with two public datasets, ScanNet and M2DGR, where our method shows state-of-the-art performance.

REFERENCES

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European conference on computer vision*. Springer, 2020, pp. 405–421.

[2] H. Guo, S. Peng, H. Lin, Q. Wang, G. Zhang, H. Bao, and X. Zhou, "Neural 3d scene reconstruction with the manhattan-world assumption," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5511–5520.

[3] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 838–15 847.

[4] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5828–5839.

[5] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2266–2273, 2021.

[6] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *CVPR*, 2021, pp. 10 318–10 327.

[7] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *ICCV*, 2021, pp. 5865–5874.

[8] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *ICCV*, 2021.

[9] S. Peng, Y. Zhang, Y. Xu, Q. Wang, Q. Shuai, H. Bao, and X. Zhou, "Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9054–9063.

[10] M. Oechsle, S. Peng, and A. Geiger, "Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5589–5599.

[11] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *arXiv preprint arXiv:2106.10689*, 2021.

[12] Y. Wei, S. Liu, Y. Rao, W. Zhao, J. Lu, and J. Zhou, "Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5610–5619.

[13] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 1907–1915.

[14] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgb-d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 918–927.

[15] R. Nabati and H. Qi, "Centerfusion: Center-based radar and camera fusion for 3d object detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1527–1536.

[16] X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "Futr3d: A unified sensor fusion framework for 3d detection," *arXiv preprint arXiv:2203.10642*, 2022.

[17] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1090–1099.

[18] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4604–4612.

[19] C. Wang, C. Ma, M. Zhu, and X. Yang, "Pointaugmenting: Cross-modal augmentation for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 794–11 803.

[20] T. Yin, X. Zhou, and P. Krähenbühl, "Multimodal virtual point 3d detection," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 494–16 507, 2021.

[21] S. Xu, D. Zhou, J. Fang, J. Yin, Z. Bin, and L. Zhang, "Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 3047–3054.

[22] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 182–17 191.

[23] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3d object detection," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 641–656.

[24] E. Xie, Z. Yu, D. Zhou, J. Philion, A. Anandkumar, S. Fidler, P. Luo, and J. M. Alvarez, "Mˆ 2bev: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation," *arXiv preprint arXiv:2204.05088*, 2022.

[25] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," *arXiv preprint arXiv:2203.17270*, 2022.

[26] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7345–7353.

[27] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," *arXiv preprint arXiv:2205.13542*, 2022.

[28] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5301–5310.

[29] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[30] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] Q. Xu, Z. Xu, J. Philip, S. Bi, Z. Shu, K. Sunkavalli, and U. Neumann, "Point-nerf: Point-based neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5438–5448.

[32] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "Mvsnet: Depth inference for unstructured multi-view stereo," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 767–783.

[33] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs, "Large scale multi-view stereopsis evaluation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 406–413.

[34] F. Poiesi and D. Boscaini, "Learning general and distinctive 3d local deep descriptors for point cloud registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[35] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: Learning local geometric descriptors from rgb-d reconstructions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1802–1811.

[36] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, "Challenging data sets for point cloud registration algorithms," *The International Journal of Robotics Research*, vol. 31, no. 14, pp. 1705–1711, 2012.

[37] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[38] F. Poiesi and D. Boscaini, "Distinctive 3d local deep descriptors," in *2020 25th International conference on pattern recognition (ICPR)*. IEEE, 2021, pp. 5720–5727.

[39] S. Ao, Q. Hu, B. Yang, A. Markham, and Y. Guo, "Spinnet: Learning a general surface descriptor for 3d point cloud registration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 753–11 762.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.