CN-RMA: Combined Network with Ray Marching Aggregation for 3D Indoor Object Detection from Multi-view Images

Guanlin Shen[1] Jingwei Huang[2] Zhihua Hu[3] Bin Wang[1*]
[1]School of Software, Tsinghua University, China [2]Tencent, China
[3]Nanjing University of Information Science and Technology, China

Abstract

This paper introduces CN-RMA, a novel approach for 3D indoor object detection from multi-view images. We observe the key challenge as the ambiguity of image and 3D correspondence without explicit geometry to provide occlusion information. To address this issue, CN-RMA leverages the synergy of 3D reconstruction networks and 3D object detection networks, where the reconstruction network provides a rough Truncated Signed Distance Function (TSDF) and guides image features to vote to 3D space correctly in an end-to-end manner. Specifically, we associate weights to sampled points of each ray through ray marching, representing the contribution of a pixel in an image to corresponding 3D locations. Such weights are determined by the predicted signed distances so that image features vote only to regions near the reconstructed surface. Our method achieves state-of-the-art performance in 3D object detection from multi-view images, as measured by mAP@0.25 and mAP@0.5 on the ScanNet and ARKitScenes datasets. The code and models are released at https://github.com/SerCharles/CN-RMA.

1. Introduction

3D object detection from multi-view images is a fundamental problem in various fields, including robotics, autonomous driving, and augmented reality (AR). However, since explicit scene geometry is unavailable to detect occlusion, it is an ill-posed problem to identify correspondences between image regions and 3D locations. Therefore, image features can be wrongly projected to 3D, leading to inaccurate detection. While occlusion is not a critical issue in open space and is ignored for autonomous driving scenarios [15, 32], it commonly exists among objects in complex environments like indoor scenes.

One straightforward solution for 3D object detection from multi-view images is to perform 3D scene reconstruction from multi-view images [3, 19, 22, 26, 30, 33] followed by 3D object detection from reconstructed point clouds [9, 24, 40]. However, such a solution is not ideal due to the lack of connectivity between the two stages. The first stage usually introduces noises and incompletion in the reconstructed 3D geometry with limited power of 3D reconstruction techniques, and such geometry loss is not resolved in the second stage. Moreover, aggregated color signals to inaccurate geometry cannot fully exploit rich image features and further harm the performance. An alternative method proposed by ImVoxelNet [25] involves aggregating 2D features extracted from multi-view images into 3D voxel volumes through unprojection in an end-to-end manner. However, due to the lack of scene geometry information, this exploratory aggregation approach struggles to effectively address complex occlusion issues, leading to feature voting from images to unrelated 3D locations.

In this paper, we present CN-RMA, an end-to-end novel 3D object detection method from multi-view images that seamlessly combines the reconstruction and detection networks with occlusion-aware feature aggregation. Our network mainly consists of a Multi-View Stereo (MVS) module [19] and a novel occlusion-aware aggregation module followed by a 3D detection module [24]. In the MVS module, we aim to reconstruct the rough scene geometry. We extract 2D features from the input images and feed them into the reconstruction network to generate a rough Truncated Signed Distance Function (TSDF) [7] as a 3D representation. Our key contribution is the occlusion-aware aggregation module called Ray Marching Aggregation (RMA), which leverages the reconstructed TSDF to detect occlusion based on ray marching. In comparison with conventional 3D detection methods that vote image features equally along rays to the 3D space, we associate different weights according to the signed distance values. Specifically, RMA incorporates the idea of volume density given TSDF inspired by NeuS [30] and accumulates transmittance

[1]*Corresponding author, email address: wangbins@tsinghua.edu.cn.
This work was supported by the National Natural Science Foundation of China under Grant 62072271.

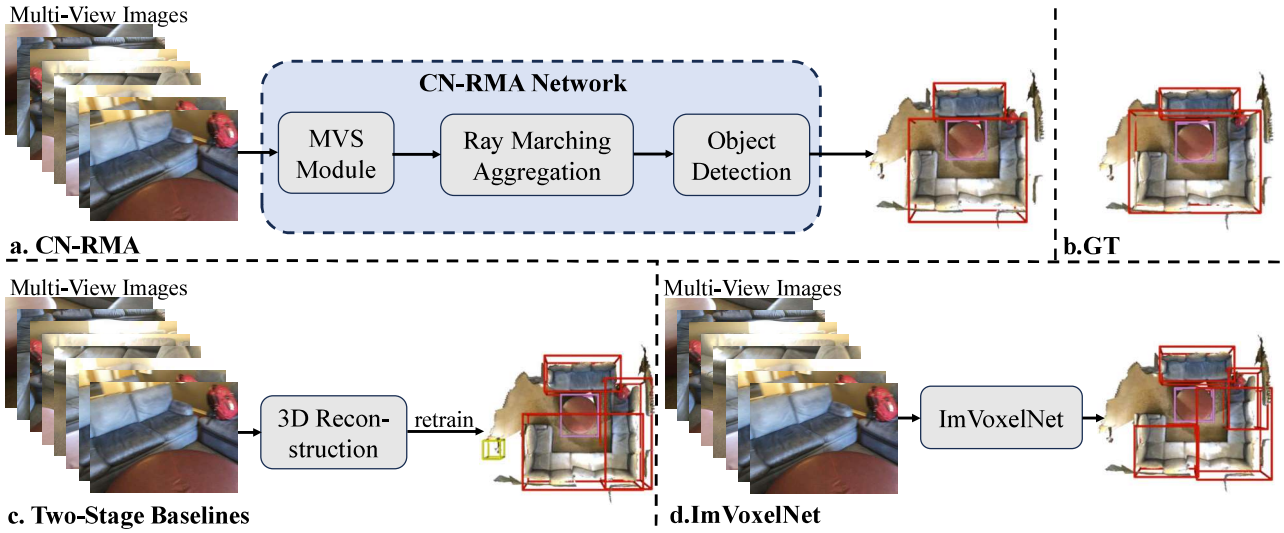

Figure 1. **The Comparison of Our CN-RMA, the two-stage method, and ImVoxelNet[25].** Our CN-RMA is an end-to-end object detection method that incorporates an occlusion-aware 2D to 3D aggregation technique. In contrast, the two-stage method lacks end-to-end trainability, while ImVoxelNet employs a heuristic aggregation method that disregards occlusion considerations.

through ray marching to calculate the weight of each point along a ray, effectively addressing the occlusion issues encountered in complex environments. Then, we can aggregate image features by weights in the 3D space aware of occlusion. Finally, we extract points with aggregated features near the reconstructed surface and pass the point cloud to the 3D detection module for object detection. Given the challenging task, we propose a pre-training and fine-tuning method to train the entire network, making the components cooperate to achieve the best performance. Figure 1 illustrates the comparison between our proposed CN-RMA, ImVoxelNet [25] and the two-stage method.

We evaluate our method on the ScanNet [8] and ARKitScenes [1] datasets to assess its performance and compare it with existing methods. Our approach outperforms other methods, achieving significant improvements in mAP@0.25 and mAP@0.5, including 3.2 and 3.0 in ScanNet, and 7.4 and 13.1 in ARKitScenes, respectively.

In summary, our contributions are three-fold:

- We establish a seamless connection between the multi-view 3D reconstruction network and 3D object detection network, enabling better exploitation of image features in 3D space for improved performance.
- We propose an innovative occlusion-aware aggregation method, RMA, which leverages the reconstructed scene TSDF to address the complex occlusion issues.
- We adopt a pretraining and finetuning scheme, and achieve the state-of-the-art performance for indoor 3D object detection from multi-view images.

2. Related Work

2.1. 3D Object Detection from Multi-View Images

3D object detection from multi-view images has been a hot topic in the vision community for many years. It aims to estimate the classes, poses, and sizes of objects from images. For outdoor scenes, a lot of methods project the features to Bird's Eye View (BEV) for the sake of memory and better performance [10, 12, 13, 29, 34]. However, the BEV representation is not suitable for 3D object detection in indoor scenes due to object stacking and occlusion. In recent years, several methods have tried to aggregate 2D features in 3D space [15, 25, 32]. For instance, ImVoxelNet [25] projects the 2D features from images into 3D space and aggregates the features with 3D CNN in the voxel form. DETR3D [32] detects objects by generating random 3D object queries and linking 3D positions to images with camera transformation. PETR [15] produces the 3D position-aware features by encoding the position information of 3D coordinates into image features. However, these exploratory aggregation methods have not taken full advantage of the scene geometry. ImGeoNet [28] introduces geometry implicit, while NeRF-Det [36] incorporates NeRF [18]. However, these methods have not considered the occlusion during the feature aggregation process, leading to inaccurate detection results.

2.2. Neural Implicit Reconstruction

To recover the 3D geometry from multi-view images, neural implicit representations are often adopted, such as the Signed Distance Function (SDF) [3, 7, 11, 19, 26]. The scene mesh can be obtained from the SDF using techniques

like Marching Cubes [17]. For instance, Atlas [19] predicts the Truncated Signed Distance Function (TSDF) of the scene with 3D CNN. And many subsequent methods have made improvements based on the Atlas network. NeuralRecon [26] splits one complete scene into fragments and uses a Gated Recurrent Unit (GRU) [5] network to fuse the 3D features of the fragments to save time and memory. VolumeFusion [3] employs deep MVS [37] techniques to predict the TSDF. However, these improvements, while enhancing the effectiveness of 3D reconstruction, have made the network more complex and challenging to combine with other networks.

In recent years, NeRF [18] based methods have utilized neural implicit fields in novel view synthesis and 3D reconstruction [2, 16, 31, 39].

For example, NeuS [30] and VolSDF [38] incorporate the SDF into neural radiance fields by integrating it into the density function, bridging the gap between the SDF and the volume density of points along each ray. It shows the possibility of sampling and weighting points in 3D space with the Truncated Signed Distance Function (TSDF), which can be used to address the occlusion issues.

2.3. 3D Object Detection from Point Clouds

3D object detection from point clouds is much more straightforward. According to the representation of point clouds, it can be divided into point cloud-based and voxel-based methods. For point cloud-based methods, the voting scheme introduced by VoteNet [21] is broadly adopted [31, 35], while PointNet++ [20] is often used to extract the features of point clouds. However, voting-based object detection methods require additional data annotation on the point clouds, making them difficult to combine with reconstruction networks. Voxel-based methods usually convert point clouds into 3D voxels and utilize 3D CNN for voxel processing [12, 40]. However, dense volumetric representation and 3D CNN are memory-consuming. Thus, sparse convolution based on sparse voxels has been introduced to improve the performance of 3D object detection [9, 24].

Our approach focuses on enhancing the performance of 3D indoor object detection from multi-view images by integrating 3D scene reconstruction methods and 3D object detection methods from point clouds. To take full advantage of the scene geometry and handle occlusions, we designed an occlusion-aware aggregation method based on neural implicit representations.

3. Method

3.1. Problem Formulation

We aim to achieve precise 3D object detection in a cluttered scene with complex occlusions using multi-view images and their corresponding camera parameters. To accomplish this, we propose a pipeline that combines a MVS reconstruction module and a 3D detection network through our occlusion-aware aggregation approach, as shown in Figure 2.

Given the input images $\{\mathbf{I}_i \in R^{h_I \times w_I \times 3}\}$, along with corresponding camera intrinsics $\{\mathbf{K}_i \in R^{3\times3}\}$ and extrinsics $\{\mathbf{R}_i \in R^{4\times4}\}$, our goal is to predict the 3D bounding boxes $\{\mathbf{b}_j\}$ and label scores $\{\mathbf{s}_j\}$ of objects in the scene. Our pipeline borrows components from multi-view stereo (MVS) [19] and point cloud-based 3D detection methods [24]. We first extract the C-channel 2D features $\mathbf{F}_i \in R^{h\times w\times C}$ for each image with the 2D backbone $\mathcal{F}$. Then we aggregate the image features $\{\mathbf{F}_i\}$ as volume feature $\mathbf{G} \in R^{W\times H\times D\times C}$ with the assistance of associated camera parameters $\{\mathbf{K}_i\}$ and $\{\mathbf{R}_i\}$ using unprojection and average pooling [19, 25, 26], which is denoted as $\mathcal{A}$.

Then we predict the rough scene TSDF $\mathbf{S} \in R^{W\times H\times D}$ from the 3D volumes $\mathbf{V}$ using the 3D reconstruction network $\mathcal{R}_i$ (Section 3.2). Section 3.3 introduces a novel occlusion-aware aggregation module $\mathcal{A}^*$ to extract the 3D geometry as a point cloud with features $\mathbf{P} \in R^{N_{pt}\times(3+C)}$, based on the rough scene TSDF $\mathbf{S}$ and a ray-marching-based voting scheme. Finally, the point cloud with features $\mathbf{P}$ are passed through the detection network $\mathcal{D}$ to obtain the 3D bounding boxes $\{\mathbf{b}_j\}$ and their corresponding label scores $\{\mathbf{s}_j\}$ (Section 3.4).

3.2. Multi-View Stereo Module

Complete reconstruction is important to avoid missing detection. While NeRF-based methods [18, 30, 38] deliver complete results, they require fitting model parameters for each specific scene. Since our task requires obtaining model parameters that can be universally applicable to all validation scenes, selecting them as our MVS module may lead to lower generalization ability or increased complexity in network training. Among end-to-end 3D reconstruction methods, we find Atlas [19] a proper choice as our MVS module since it can be trained and used to predict the reconstruction in an end-to-end manner, including a 2D backbone and a 3D reconstruction network.

Specifically, for an input image I_i, we first extract the 2D features $\mathbf{F}_i$ with C channels using the ResNet50-FPN [14] backbone. We then lift per-view 2D features into 3D via back projection given camera parameters $\{\mathbf{K}_i\}$ and $\{\mathbf{R}_i\}$, and aggregate them to generate 3D volume features $\mathbf{V}$ with the voxel size of $4cm^3$ by average-pooling [19, 25, 26].

Next, we feed the 3D volume features into the 3D CNN reconstruction network in Atlas [19], which features an encoder-decoder structure with skip connections [23] with a $1 \times 1 \times 1$ convolutional head, to obtain the rough scene TSDF $\mathbf{S}$. As suggested by [19], we employ the L1 loss at

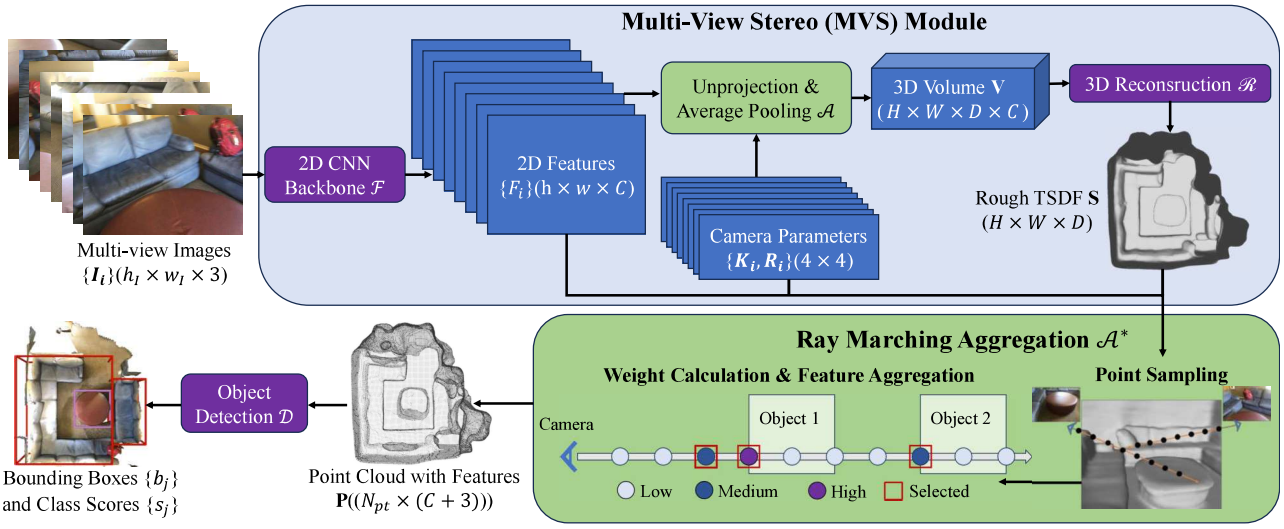

Figure 2. **The overall architecture of our CN-RMA method.** The purple blocks represent neural networks, while the green blocks represent modules without trainable neurons. Following Atlas [19], the 2D CNN backbone $\mathcal{F}$ is a ResNet50-FPN network [14], and the 3D reconstruction network $\mathcal{R}$ is a 3D CNN network that features an encoder-decoder structure with skip connections with a $1 \times 1 \times 1$ convolutional head. Following FCAF3D [24], the object detection network $\mathcal{D}$ is a sparse 3D convolutional network comprising a ResNet34 backbone [4] and a 4-layer decoder network.

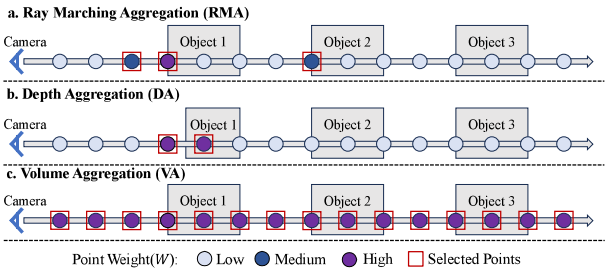

Figure 3. The 1D illustration comparing our Ray Marching Aggregation (RMA) method, with the Depth Aggregation method (DA) based on depth prediction, and the Volume Aggregation method(VA) based on unprojection [19, 25, 26]. The points depicted in the illustration represent sample points along a ray, with their colors indicating their respective weights. The points enclosed within one red square represent selected points.

three different scales to boost the training:

$$L_{recon} = \sum_{i=1}^{3} \|\text{sgn}(\mathbf{S}_i)\ln(1+|\mathbf{S}_i|) - \text{sgn}(\hat{\mathbf{S}}_i)\ln(1+|\hat{\mathbf{S}}_i|)\|_1 \quad (1)$$

where $\mathbf{S}_i$ and $\hat{\mathbf{S}}_i$ denotes the predicted and ground truth TSDF values from coarse to fine, and we denote the predicted TSDF with the finest scale $\mathbf{S}_3$ as the rough scene TSDF $\mathbf{S}$ utilized in the following sections.

3.3. Ray Marching Aggregation

Although we predict a 3D feature volume by directly averaging the lifted image features similar to [19, 25, 26] in the reconstruction stage, volume features are polluted from certain views since image features can vote to unobserved space due to the lack of consideration for occlusion. An illustration of such aggregation from view to volume is illustrated in Figure 3(c).

A straightforward solution to handle occlusion would be to directly render a depth map and vote image features only to the surface specified by the depth map (Figure 3(b)). However, such a voting scheme is sensitive to the quality of the depth map produced from the reconstruction. In order to improve robustness, we introduce a soft occlusion-aware aggregation scheme called Ray Marching Aggregation (RMA), inspired by NeRF [18] and NeuS [30]. Specifically, we compute the volume density given TSDF according to NeuS [30]. We sample points on the ray of each pixel by ray marching and compute the opacity of each point by accumulating transmittance according to NeRF [18]. As a result, we can compute 3D features by averaging image features weighted by the transmittance from different views. As illustrated in Figure 3(a), RMA can vote image features into the scene by softly considering occlusions. Finally, we extract points and aggregated features near the reconstructed surface and pass the point cloud to the 3D detection module for object detection.

In detail, given a 2D feature map $\mathbf{F}_i$ and corresponding camera parameters $\mathbf{K}_i$ and $\mathbf{R}_i$, we assume that a ray

$\mathbf{p}_{i,u,v}(t) = \mathbf{o}_i + t \cdot \mathbf{d}_{i,u,v}$ is emitted from the camera towards the object represented by the pixel $\mathbf{F}_i(u, v, :)$. For convenience, we denote the ray as $\mathbf{p}(t) = \mathbf{o} + t \cdot \mathbf{d}$ in the following sections, where $\mathbf{o}$ and $\mathbf{d}$ can be determined as the origin and direction of the ray given the camera parameter and pixel location. We use ray marching to sample a set of points $\{\mathbf{p}(t_i)\}$ along the ray with t_i in increasing order. Inspired by NeuS [30], the opacity value at the interval $[t_i, t_{i+1}]$ can be modeled as

$$\alpha(\mathbf{p}(t_i)) = \max\left(\frac{\Phi(\mathbf{S}(\mathbf{p}(t_i))) - \Phi(\mathbf{S}(\mathbf{p}(t_{i+1})))}{\Phi(\mathbf{S}(\mathbf{p}(t_i)))}, 0\right), \quad (2)$$

with $\Phi(x)$ as the sigmoid function, and $\mathbf{S}(\mathbf{p}(t_i))$, which denotes the TSDF value of sample point $\mathbf{p}(t_i)$, is obtained by querying the voxel closest to $\mathbf{p}(t_i)$. Then, the opacity value $W(\mathbf{p}(t_i))$ of each sampled point $\mathbf{p}(t_i)$ can be computed according to NeRF [18] as

$$W(\mathbf{p}(t_i)) = T(\mathbf{p}(t_i)) \cdot \alpha(\mathbf{p}(t_i)), \quad (3)$$

where, $T(\mathbf{p}(t_i))$ is the accumulated transmittance at the interval $[0, t_i]$ as

$$T(\mathbf{p}(t_i)) = \prod_{j=0}^{i-1} (1 - \alpha(\mathbf{p}(t_j))) \quad (4)$$

We retain only those points with an opacity greater than the threshold θ_{rma}.

To compute the 3D feature of each retained point, we average image features $\mathbf{F}_i(u, v, :)$ weighted by the opacity of the point for i-th image, where (u, v) is the projected location of the point to the image. As a result, we obtain the point cloud with features $\mathbf{P}$ by concatenating 3D coordinates and 3D features.

3.4. 3D Object Detection Network

We feed the reconstructed point cloud with aggregated feature $\mathbf{P}$ into the 3D detection network $\mathcal{D}$ for the final detection results. We apply FCAF3D [24] as our detection network considering efficiency, memory consumption, and performance.

Firstly, we transform $\mathbf{P}$ into sparse voxels [4] with a voxel size of $1cm^3$. Then we pass these sparse voxels into FCAF3D [24] to predict the classification scores $\mathbf{s}$, bounding box regression parameters $\mathbf{b}$, and 3D centerness c [24, 25, 27] of each voxel. The detection loss $\mathcal{L}_D$ is the same as proposed in FCAF3D:

$$L_{det} = \frac{1}{N_{pos}} \sum_{\hat{x},\hat{y},\hat{z}} (L_{cls}(\hat{\mathbf{s}}, \mathbf{s}) + m \cdot L_{reg}(\hat{\mathbf{b}}, \mathbf{b}) + m \cdot L_{cntr}(\hat{c}, c)) \quad (5)$$

Where $\{(\hat{x}, \hat{y}, \hat{z})\}$ represents the sparse voxel coordinates, $m = \mathbb{1}_{\{\hat{\mathbf{s}}_{\hat{x},\hat{y},\hat{z}} \neq 0\}}$ indicates whether a sparse voxel matches an object, $N_{pos} = \sum_{\hat{x},\hat{y},\hat{z}} m$ denotes the number of voxels matching an object. L_{cls} is the focal loss to supervise $\mathbf{s}$, L_{reg} is the IOU loss for $\mathbf{b}$, and L_{cntr} is the binary cross-entropy loss for c.

3.5. Training Procedure

Due to the complexity of our architecture, which combines a MVS module and a detection network, training the modules from scratch may lead to overfitting. For example, to effectively train the detection network, it is essential to feed the network with high-quality point clouds with features, which makes the initialization of the reconstruction network important. Therefore, we employ a pre-training and joint fine-tuning scheme in our training procedure to strike a balance between the 3D reconstruction network and the 3D detection network.

Firstly, we pre-train the 2D backbone and 3D reconstruction network using only the reconstruction loss L_{recon}, to fully leverage the 3D geometry. Subsequently, we freeze the aforementioned networks and proceed to pre-train the 3D detection network by utilizing the ray marching aggregation module and solely considering the detection loss L_{det}. Finally, to obtain the ultimate 3D detection results, we jointly fine-tune the entire network with the total loss denoted as $L_t = \lambda \cdot L_{recon} + L_{det}$. Where λ is a constant to balance the reconstruction loss L_{recon} and detection loss L_{det}.

4. Experiments

Section 4.1 introduces the datasets, metrics, and baselines in detail. The implementation details are introduced in Section 4.2. We compare our method with the state-of-the-art methods in Section 4.3, where we show a clear advantage in terms of mAP@0.25 and mAP@0.5. Section 4.4 presents the ablation studies, and the improvements in the detection results demonstrate the effectiveness of our proposed ideas.

4.1. Datasets, Metrics, and Baselines

We evaluate our method, CN-RMA, using two indoor object detection datasets: ScanNet [8] and ARKitScenes [1]. Following the settings of prior methods [25, 28, 36], we detect Axis-Aligned Bounding Boxes (AABB) for objects across 18 categories in ScanNet. The dataset is divided into 1201 training scans and 312 testing scans. For ARKitScenes, which comprises 4498 training scans and 549 testing scans, we detect Oriented Bounding Boxes (OBB) for objects across 17 categories. It is worth noting that the 3D geometric annotations in the ARKitScenes dataset are relatively rough. The depth map resolution of ARKitScenes is 192×256, which is much lower compared to the 480×640 resolution of ScanNet. For the evaluation metrics, we choose the normally used mean average precision (mAP) with thresholds of 0.25 and 0.5 in both

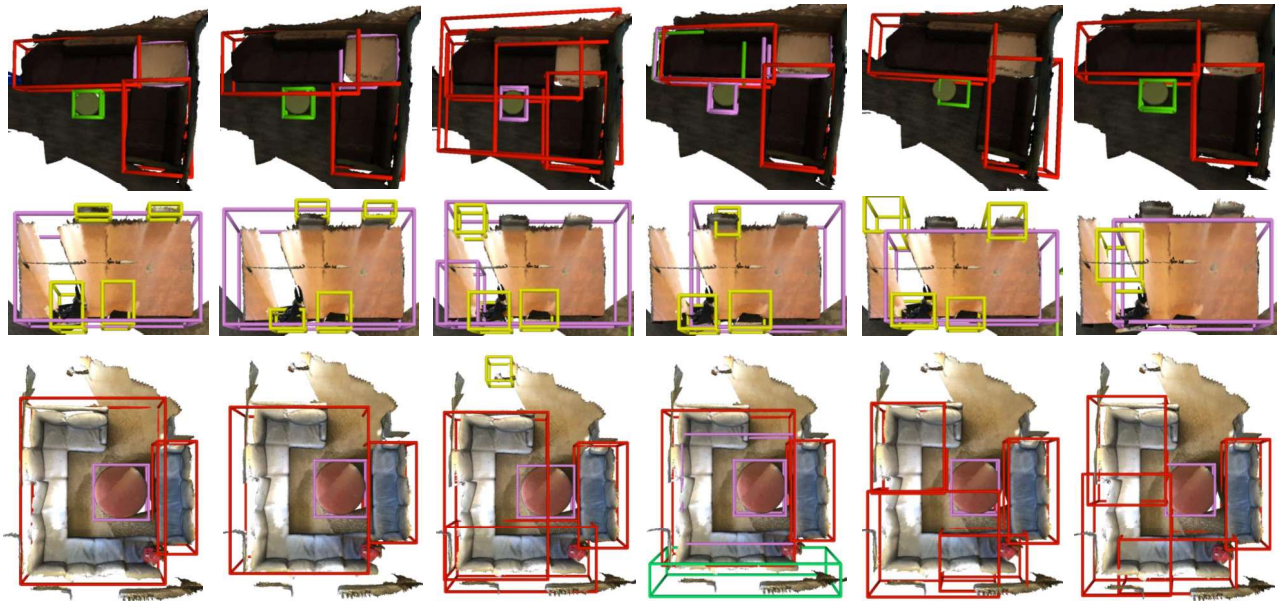

Figure 4. **Visualization of 3D object detection results from ScanNet [8].** From above to below are scene0559_01, scene0598_00, and scene0701_00 from ScanNet. Atlas+FCAF denotes the two-stage baseline combining Atlas [19] and FCAF3D [24], and Neucon+FCAF denotes the two-stage baseline combining NeuralRecon [26] and FCAF3D.

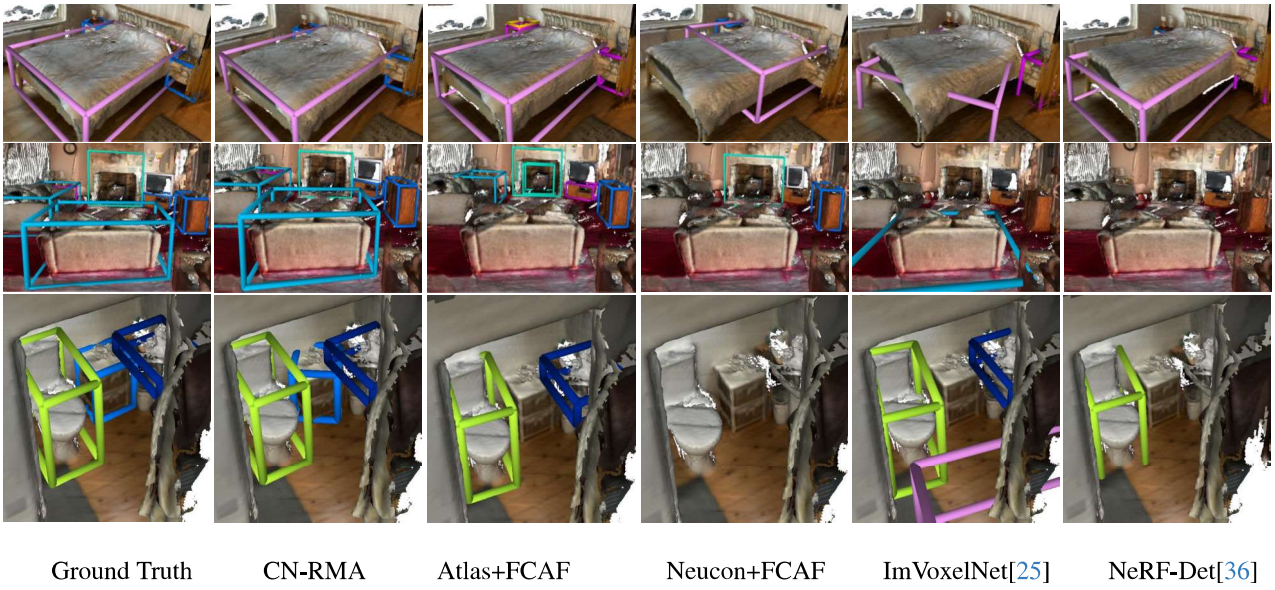

Figure 5. **Visualization of 3D object detection results from ARKitScenes [1].** From above to below are scenes 44358583, 45663154, and 45261181 from ARKitScenes. Atlas+FCAF denotes the two-stage baseline combining Atlas [19] and FCAF3D [24], and Neucon+FCAF denotes the two-stage baseline combining NeuralRecon [26] and FCAF3D.

datasets. For a fair comparison, we choose the previous state-of-the-art methods for indoor 3D object detection from multi-view images, namely ImVoxelNet [25], NeRF-Det [36], and ImGeoNet [28], as well as the two-stage baselines. As mentioned before, the two-stage baseline is a straightforward combination of 3D reconstruction and 3D

Method	mAP@0.25↑	mAP@0.5↑
ImVoxelNet [25]	46.7	23.4
NeRF-Det [36]	53.5	27.4
ImGeoNet [28]	54.8	28.4
Atlas [19]+FCAF3D [24]	55.4	33.8
NeuralRecon [26]+FCAF3D	51.5	31.6
Ours (CN-RMA)	**58.6**	**36.8**

Table 1. mAP@0.25 and mAP@0.5 results of the ScanNet [8] dataset. We directly cite the experimental results from the ImGeoNet [28] paper.

Method	mAP@0.25↑	mAP@0.5↑
ImVoxelNet [25]	27.3	4.3
NeRF-Det [36]	39.5	21.9
ImGeoNet [28]	60.2	43.4
Atlas [19]+FCAF3D [24]	51.3	40.6
NeuralRecon [26]+FCAF3D	36.3	24.9
Ours (CN-RMA)	**67.6**	**56.5**

Table 2. mAP@0.25 and mAP@0.5 results of the ARKitScenes [1] dataset. We directly cite the experimental results from the ImGeoNet [28] paper.

detection methods. Specifically, Atlas [19] and NeuralRecon [26] are utilized to reconstruct the 3D point clouds, while FCAF3D [24] is applied for the 3D detection.

4.2. Implementation Details

CN-RMA is implemented using the MMDetection3D [6] framework. We set the feature channels C to 32. The weight threshold of our aggregation approach, θ_{rma}, is set to 0.05. The loss weight λ is set to 0.5. We sample 300 points for each pixel in ray marching, and the maximum t is set as the diagonal length of the volume $\mathbf{V}$. All experiments are conducted on 4 NVIDIA A6000 GPUs with a batch size of 1. More details are included in the supplementary material.

4.3. Comparison

We compare our method CN-RMA with the previous state-of-the-art method and two-stage baselines, as shown in Tables 1 and 2. Our method achieves superior performance on both the ScanNet [8] and ARKitScenes [1] datasets, outperforming other approaches in terms of mAP@0.25 and mAP@0.5. Our method surpasses the previous state-of-the-art method ImGeoNet [28] by 3.8 for mAP@0.25 and 8.4 for mAP@0.5 in ScanNet, and 7.4 for mAP@0.25 and 13.1 for mAP@0.5 in ARKitScenes. When compared to the two-stage baseline combining Atlas [19] and FCAF3D [24], our method outperforms it by 3.2 for mAP@0.25 and 3.0 for mAP@0.5 in ScanNet, and 16.3 for mAP@0.25 and 15.9 for mAP@0.5 in ARKitScenes. As shown in Figure 4, ImVoxelNet and NeRF-Det often predict inaccurate bounding boxes due to insufficient utilization of geometric information. As for two-stage baselines, it is easy to predict inaccurate bounding boxes and miss some objects, due to noises and incomplete scene geometry reconstructed with MVS methods. Figure 5 shows the detection results of the ARKitScenes dataset. It is shown that the proposed method can also predict good results even with relatively low-quality reconstructed geometry, demonstrating the robustness of our method.

Method	parameter	mAP@0.25↑	mAP@0.5↑
VA	–	31.1	11.3
RMA	$\theta_{rma} = 0.02$	**58.6**	**37.0**
RMA	$\theta_{rma} = 0.05$	**58.6**	36.8
RMA	$\theta_{rma} = 0.10$	57.3	35.4
DA	$k = 1$	57.1	33.8
DA	$k = 2$	56.9	34.1
DA	$k = 3$	56.3	34.2
DA	$k = 4$	57.9	34.7

Table 3. **Ablation study results of different aggregation schemes and parameters.** VA refers to the Volume Aggregation method based on unprojection [19, 25, 26]. DA denotes the Depth Aggregation method relying on depth prediction. θ_{rma} represents the weight threshold for selecting sample points in our RMA method, while k denotes the number of point pairs selected in the DA method. All experiments are conducted in ScanNet [8] with our proposed parameters following our standard training steps.

Training scheme	mAP@0.25↑	mAP@0.5↑
Joint Train From Scratch	48.2	28.8
P-MVS + JFT	50.3	30.9
P-MVS + P-Det	55.8	34.7
P-MVS + P-Det + JFT	**58.6**	**36.8**

Table 4. **Ablation study results of different training schemes.** P-MVS denotes pre-training the MVS module, P-Det denotes pre-training the detection network, and JFT denotes jointly fine-tuning the entire network. All experiments are conducted in ScanNet [8] with our proposed parameters.

4.4. Ablation Study

In this section, we present the experimental results conducted on the ScanNet dataset to compare various aggregation schemes with different hyper-parameters and training schemes of our method.

4.4.1 Aggregation Schemes

We begin by comparing our occlusion-aware aggregation approach RMA, with the other two schemes shown in Figure 3: Volume Aggregation (VA, Figure 3(c)) and Depth Aggregation (DA, Figure 3(b)). The VA method lifts per-view 2D features into 3D via back projection [19, 25, 26] directly. Specifically, we directly convert the global feature volume $\mathbf{V}$ used in the 3D reconstruction network into a point cloud with features for detection. The DA method directly lifts 2D features to point clouds through depth maps of each view obtained from the reconstruction results. In detail, for points along a ray, we define $\mathbf{p}(t_i)$ as the First Hitting Point (FHP) if

$$i = \arg\min_j \{j \mid \mathbf{S}(\mathbf{p}(t_j)) \cdot \mathbf{S}(\mathbf{p}(t_{j+1})) \leq 0\} \tag{6}$$

which represents the first intersecting point. Considering the possible errors in 3D reconstruction, we select $2 \cdot k$ points from $\mathbf{p}(t_{i-k+1})$ to $\mathbf{p}(t_{i+k})$. The weight of a selected point decreases linearly as it moves farther away from the FHP.

To study the sensitivity of the hyper-parameters, we explore different values of θ_{rma} in our RMA module with 0.02, 0.05, and 0.10. We refrain from testing smaller θ_{rma} values due to excessive GPU memory usage. For DA, we experiment with different values of k, including 1, 2, 3, and 4.

Table 3 presents the results of the different aggregation schemes and hyper-parameters. Our RMA method achieves the best performance in both mAP@0.25 and mAP@0.5, surpassing the VA method significantly by 27.5 in mAP@0.25 and 25.7 in mAP@0.5. The comparison reveals that integrating the rough scene TSDF obtained from 3D reconstruction into the aggregation process effectively enhances detection performance by providing valuable 3D geometry information and considering occlusion. Additionally, our RMA method outperforms the best results of DA by 0.7 in mAP@0.25 and 2.3 in mAP@0.5. It indicates that with the possible errors in the reconstructed scene TSDF, the DA method that directly chooses the first intersecting point along a ray may not always yield optimal results. In contrast, our RMA method offers more flexibility by combining local geometry information conveyed by α and accumulated ray information conveyed by T. Moreover, the results do not change much with different θ_{rma} indicating the robustness of our RMA method.

4.4.2 Training Schemes

We compare our training scheme, which involves subsequent pre-training of the MVS module and the detection network followed by joint fine-tuning of the entire network, with three other schemes to demonstrate the effectiveness of our proposed approach. The straightforward training scheme is to train the entire network from scratch without any pre-training. There are also several possible schemes considering the pre-training and fine-tuning. Specifically, the second scheme that we compare focuses on pre-training only the MVS module and then jointly training the entire network. The last scheme involves pre-training the MVS module and then freezing it to pre-train the detection module, without joint fine-tuning of the entire network.

The comparison results presented in Table 4 demonstrate that our three-step training scheme with pre-training and fine-tuning achieves the best performance in both mAP@0.25 and mAP@0.5. Our training scheme outperforms the scheme without any pre-training by 10.4 in mAP@0.25 and 8.0 in mAP@0.5. Additionally, it outperforms the scheme without pre-training of the detection network by 8.3 in mAP@0.25 and 5.9 in mAP@0.5. These comparisons highlight the importance of both pre-training the MVS module and pre-training the reconstruction network for optimal performance. This is necessary to avoid potential overfitting caused by the complexity of our architecture and to provide a solid geometry foundation for our RMA aggregation scheme, which heavily relies on reliable geometry information from the reconstructed scene TSDF. Furthermore, our training scheme outperforms the scheme without fine-tuning by 2.8 in mAP@0.25 and 2.1 in mAP@0.5, demonstrating the effectiveness of fine-tuning. Fine-tuning facilitates knowledge transfer and synergistic interaction between the MVS module and the detection network, contributing to improved performance.

Overall, our experimental results validate the efficacy of our training scheme, emphasizing the significance of pre-training, fine-tuning, and the interplay between the MVS module and the detection network in achieving superior performance for our method.

5. Conclusion

In this paper, we introduced CN-RMA, a novel 3D indoor object detection method from multi-view images. Our proposed approach surpasses previous state-of-the-art methods and outperforms two-stage baselines. We also present an effective occlusion-aware technique for aggregating 2D features into 3D point clouds using rough scene TSDF, which holds potential for integration into other 3D scene understanding tasks from multi-view images.

Future work should focus on exploring techniques for further improving the performance of CN-RMA, such as investigating alternative aggregation schemes or incorporating additional contextual information, which could be beneficial. We anticipate continued advancements in 3D indoor object detection and related research areas by addressing these limitations and building upon our findings.

References

[1] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 2, 5, 6, 7, 1, 3, 4

[2] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021. 3

[3] Jaesung Choe, Sunghoon Im, Francois Rameau, Minjun Kang, and In So Kweon. Volumefusion: Deep depth fusion for 3d scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16086–16095, 2021. 1, 2, 3

[4] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019. 4, 5

[5] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 3

[6] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020. 7

[7] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996. 1, 2

[8] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 2, 5, 6, 7, 1, 3

[9] JunYoung Gwak, Christopher Choy, and Silvio Savarese. Generative sparse detection networks for 3d single-shot object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 297–313. Springer, 2020. 1, 3

[10] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 2

[11] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568, 2011. 2

[12] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019. 2, 3

[13] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 2

[14] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3, 4

[15] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022. 1, 2

[16] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227. Springer, 2022. 3

[17] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 3

[18] 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. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3, 4, 5

[19] Zak Murez, Tarrence Van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 414–431. Springer, 2020. 1, 2, 3, 4, 6, 7, 8

[20] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 3

[21] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019. 3

[22] Yufan Ren, Tong Zhang, Marc Pollefeys, Sabine Süsstrunk, and Fangjinhua Wang. Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16685–16695, 2023. 1

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference,*

Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, pages 234–241. Springer, 2015. 3

[24] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Fcaf3d: Fully convolutional anchor-free 3d object detection. In *European Conference on Computer Vision*, pages 477–493. Springer, 2022. 1, 3, 4, 5, 6, 7, 2

[25] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2397–2406, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[26] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. 1, 2, 3, 4, 6, 7, 8

[27] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 5

[28] Tao Tu, Shun-Po Chuang, Yu-Lun Liu, Cheng Sun, Ke Zhang, Donna Roy, Cheng-Hao Kuo, and Min Sun. Imgeonet: Image-induced geometry-aware voxel representation for multi-view 3d object detection. In *Proceedings of the IEEE international conference on computer vision*, 2023. 2, 5, 6, 7, 1, 3

[29] Haiyang Wang, Chen Shi, Shaoshuai Shi, Meng Lei, Sen Wang, Di He, Bernt Schiele, and Liwei Wang. Dsvt: Dynamic sparse voxel transformer with rotated sets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13520–13529, 2023. 2

[30] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 1, 3, 4, 5

[31] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 3

[32] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022. 1, 2

[33] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Hf-neus: Improved surface reconstruction using high-frequency details. *Advances in Neural Information Processing Systems*, 35:1966–1978, 2022. 1

[34] Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. Frustumformer: Adaptive instance-aware resampling for multi-view 3d detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5096–5105, 2023. 2

[35] Qian Xie, Yu-Kun Lai, Jing Wu, Zhoutao Wang, Dening Lu, Mingqiang Wei, and Jun Wang. Venet: Voting enhancement network for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3712–3721, 2021. 3

[36] Chenfeng Xu, Bichen Wu, Ji Hou, Sam Tsai, Ruilong Li, Jialiang Wang, Wei Zhan, Zijian He, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Nerf-det: Learning geometry-aware volumetric representation for multi-view 3d object detection. In *ICCV*, 2023. 2, 5, 6, 7, 1, 3, 4

[37] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 3

[38] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 3

[39] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 3

[40] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 1, 3