RGB-Depth Fusion GAN for Indoor Depth Completion

Haowen Wang [1], Mingyuan Wang [1], Zhengping Che [2], Zhiyuan Xu [2],
Xiuquan Qiao [1*], Mengshi Qi [3], Feifei Feng [2], Jian Tang [2*]
[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications
[2] AI Innovation Center, Midea Group
[3] School of Computer Science, Beijing University of Posts and Telecommunications
{hw.wang,wmingyuan,qiaoxq,qms}@bupt.edu.cn {chezp,xuzy70,feifei.feng,tangjian22}@midea.com

Abstract

The raw depth image captured by the indoor depth sensor usually has an extensive range of missing depth values due to inherent limitations such as the inability to perceive transparent objects and limited distance range. The incomplete depth map burdens many downstream vision tasks, and a rising number of depth completion methods have been proposed to alleviate this issue. While most existing methods can generate accurate dense depth maps from sparse and uniformly sampled depth maps, they are not suitable for complementing the large contiguous regions of missing depth values, which is common and critical. In this paper, we design a novel two-branch end-to-end fusion network, which takes a pair of RGB and incomplete depth images as input to predict a dense and completed depth map. The first branch employs an encoder-decoder structure to regress the local dense depth values from the raw depth map, with the help of local guidance information extracted from the RGB image. In the other branch, we propose an RGB-depth fusion GAN to transfer the RGB image to the fine-grained textured depth map. We adopt adaptive fusion modules named W-AdaIN to propagate the features across the two branches, and we append a confidence fusion head to fuse the two outputs of the branches for the final depth map. Extensive experiments on NYU-Depth V2 and SUN RGB-D demonstrate that our proposed method clearly improves the depth completion performance, especially in a more realistic setting of indoor environments with the help of the pseudo depth map.

1. Introduction

Nowadays, depth sensors have been widely used to provide reliable 3D spatial information in a variety of applications, such as augmented reality, indoor navigation, and 3D reconstruction tasks [8, 20, 42]. However, most existing commercial depth sensors (*e.g.,* Kinect [26], RealSense [16], and Xtion [2]) for indoor spatial perception are not powerful enough to generate a precise and lossless depth map, as shown in the top row of Fig. 1. These sensors often produce many hole regions with invalid depth pixels due to transparent, shining, and dark surfaces as well as too close or too far edges, and these holes significantly affect the performance of downstream tasks on the depth maps (a.k.a., depth images). To address the issue from imperfect depth maps, there have been a lot of approaches to reconstruct the whole depth map from the raw depth map, called *depth completion*. As RGB images provide rich color and texture information compared with depth maps, the aligned RGB image is commonly used to guide the depth completion of a depth map. To be more specific, the depth completion task is usually conducted as using a pair of raw depth and RGB images captured by one depth sensor to complete and refine the depth values.

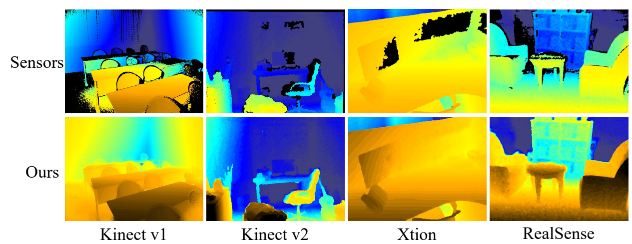

Figure 1. Showcases of the raw depth maps (top) collected by sensors from the SUN RGB-D dataset [36] and the corresponding depth completion results (bottom) of our method.

Recent studies have produced significant progress in depth completion tasks with convolutional neural networks (CNNs) [3, 12, 17, 23, 29, 32]. Ma and Karaman [23] introduced an encoder-decoder network to directly regress

*Corresponding authors.

the dense depth map from a sparse depth map and an RGB image. The method has shown great progress compared to conventional algorithms [21, 34, 39], but its predicted dense depth maps are often too blurry. To further generate a more refined completed depth map, lots of works have recently arisen, which can be divided into two groups with different optimization methods. The first group of works [3, 22, 29] learn affinities for relative pixels and iteratively refine depth predictions. These methods highly rely on the accuracy of the raw global depth map and suffer the inference inefficiency. Other works [12, 17, 18, 32] analyze the geometric characteristic and adjust the feature network structure accordingly, for instance, by estimating the surface normal or projecting depth into discrete planes. These methods require depth completeness without missing regions, and the model parameters may not be efficiently generalized to different scenes. In any case, the RGB image is merely used as superficial guidance or auxiliary information, and few methods deeply consider the textural and contextual information. At this point, the depth completion task is more or less degraded to a monocular depth estimation task that is conceptually simple but practically difficult.

More remarkably, most of the above methods [3, 18, 23] uniformly randomly sample a certain number of valid pixels from the dense depth image $\mathbf{d}_{raw}$ and $\mathbf{d}_{gt}$ to mimic the sparse depth map $\mathbf{d}^*$ for training and evaluation, respectively. Such sampling strategy is credible in some scenes, such as the outdoor range-view depth map generated by LiDAR. However, the sampled patterns are quite different from the real missing patterns, such as the large missing regions and semantic missing patterns shown in Fig. 1, in indoor depth maps. Therefore, though existing methods are shown to be effective for completing uniformly sparse depth maps, it remains unverified whether they perform well enough for indoor depth completion.[†]

To solve these problems, we propose a novel two-branch end-to-end network to generate a completed dense depth map for indoor environments. Inspired by generative adversarial networks (GANs) [14, 15, 24, 27], we introduce the *RGB-depth Fusion GAN* (RDF-GAN) for fusing an RGB image and a depth map. RDF-GAN maps a conditional RGB image from the RGB domain to a dense depth map from the depth domain through the latent spatial vector generated by the incomplete depth map. We further design a *constraint network* to restrict the depth values of the fused map, with the help of *weighted-adaptive instance normalization* (W-AdaIN) modules and a *local guidance module*. Afterwards, a *confidence fusion head* concludes the final depth map completion.

In addition, we propose an exploitation technique, which samples raw depth images to produce pseudo depth maps for training. According to the characteristic of the indoor depth missing, we utilize the RGB images and semantic labels to produce masking regions for raw depth maps, which is more realistic than the simple uniform sampling. Experiments show that the model learning from pseudo depth maps can more effectively fill in large missing regions for raw depth images captured indoors.

[†]Please refer to Section 1 of the supplementary for more discussions.

Our main contributions are summarized as the following:

- We propose a novel end-to-end GAN-based network, which effectively fuses a raw depth map and an RGB image to reproduce a reasonable dense depth map.
- We design and utilize the pseudo depth maps, which are in line with the raw depth missing distribution in indoor scenarios. Training with pseudo depth maps significantly improves the model's depth completion performance, especially in more realistic settings of indoor environments.
- Our proposed method achieves the state-of-the-art performance on NYU-Depth V2 and SUN RGB-D for depth completion and proves its effectiveness in improving downstream task performance such as object detection.

2. Related Work

Depth Completion. Recent works have extensively applied deep neural networks for depth estimation and completion tasks with remarkable improvements. Ma and Karaman [23] used an encoder-decoder structure with CNNs to predict the full-resolution depth image directly from a set of depth samples and RGB images. On this basis, some methods incorporating additional output branches to assist in the generation of depth maps have been proposed. Qiu *et al*. [32] produced dense depth using the surface normal as the intermediate representation. Huang *et al*. [12] applied the boundary consistency to solve the issue of vague structures. Lee *et al*. [18] introduced the Plane-Residual representation to interpret depth information and factorized the depth regression problem into a combination of discrete depth plane classification and plane-by-plane residual regression. Zhang *et al*. [40] uses GANs to solve both semantic segmentation and depth completion tasks in outdoor scenarios. Cheng *et al*. [3] proposed the convolutional spatial propagation network (CSPN) and generated the long-range context through a recurrent operation to lessen the burden of directly regressing the absolute depth information. Park *et al*. [29] improved CSPN by non-local spatial and global propagations. These methods prove that the encoder-decoder network can effectively perform depth completion and obtain a more refined depth map through additional optimization. In this work, we extend the encoder-decoder structure to build our depth completion model.

RGB-D Fusion. The fusion of both RGB and depth data (a.k.a., the RGB-D fusion) is essential in many tasks such as semantic segmentation and depth completion. While most existing methods [23, 25] only concatenate aligned pixels

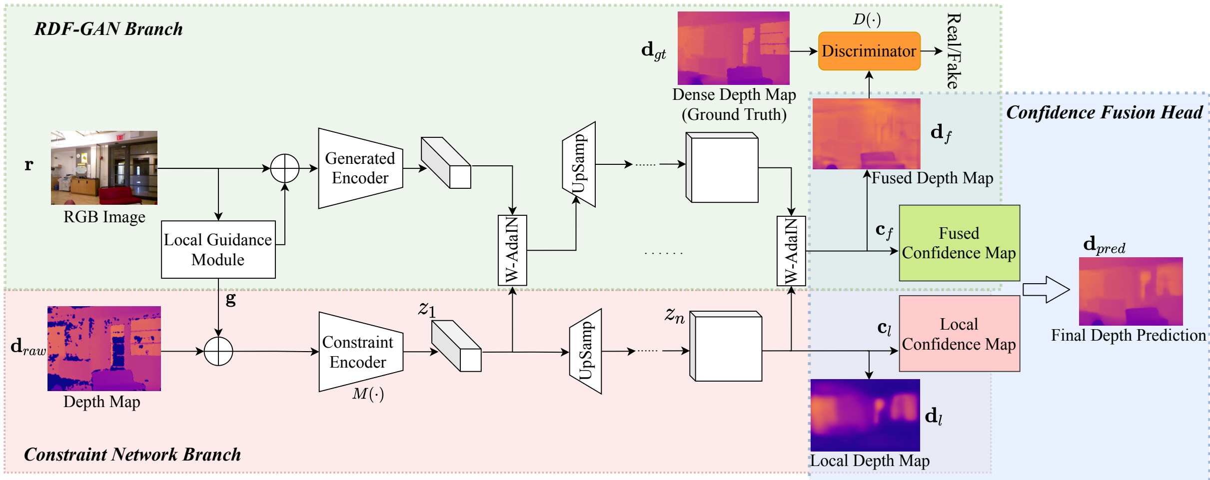

Figure 2. The overview of the proposed end-to-end depth completion method. The architecture consists of two main components: the RDF-GAN branch and the constraint network branch. The generator of RDF-GAN generates a fused depth map, which is distinguished as real/fake by the discriminator. The local guidance module and W-AdaIN share the features across stages. The confidence fusion head merges the fused depth map and local depth map generated by the two branches to produce the final prediction.

from RGB and depth features, more effective and advanced RGB-D fusions have been proposed recently. Cheng *et al*. [4] designed a gated fusion layer to learn the different weights of each modality in different scenes. Park *et al*. [30] fused multi-level RGB-D features in a very deep network through residual learning. Du *et al*. [6] proposed a novel cross-modal translate network to represent the complementary information and enhance the discrimination of extracted features. In this work, we design the two-branch structure and the W-AdaIN modules to better capture and fuse RGB and depth features.

Generative Adversarial Networks. Generative adversarial networks (GANs) have achieved great success in a variety of image generation tasks such as image-style transfer, realistic image generation, and image synthesis. Mirza *et al*. [27] proposed the conditional GAN to direct the data generation process by combining the additional information as a condition. Karras *et al*. [15] introduced a style-based GAN to embed the latent code into a latent space to affect the variations of generated images. Ma *et al*. [24] proposed a GAN for infrared and visible images. In this work, we use a GAN-based structure fusing RGB images and depth maps to generate dense depth maps with fine-grained textures.

3. Method

In this section, we describe our end-to-end depth completion method, as shown in Fig. 2. The proposed model takes a raw (noisy and possibly incomplete) depth map and its corresponding RGB image as the input, and outputs the completed and refined dense depth map estimation. The model mainly consists of two branches: a constraint network branch (Section 3.1) and an RGB-depth Fusion GAN (RDF-GAN) branch (Section 3.2). The constraint network and RDF-GAN take a depth map and an RGB image as the input, respectively, and produce their depth completion results. To fuse the representations between the two branches, a local guidance module and a series of intermediate fusion modules called W-AdaIN (Section 3.3) are deployed at different stages of the model. Finally, a confidence fusion head (Section 3.4) combines the outputs of the two channels and provides more reliable and robust depth completion results. Moreover, we introduce the training strategy with pseudo depth maps (Section 3.5) and describe the overall loss function for training (Section 3.6).

3.1. Constraint Network Branch

The first branch is composed of a constraint network, which reproduces a local full-resolution depth map and a confidence map through a convolutional encoder-decoder structure. The encoder-decoder structure is based on ResNet-18 [10] and pre-trained on the ImageNet dataset [5]. As illustrated in Fig. 3 and the bottom-left part of Fig. 2, given the raw depth image $\mathbf{d}_{raw} \in \mathbb{R}^{H\times W\times 1}$ and the RGB image $\mathbf{r}$, the network outputs a dense local depth map $\mathbf{d}_l \in \mathbb{R}^{H\times W\times 1}$ and a local confidence map $\mathbf{c}_l \in \mathbb{R}^{H\times W\times 1}$.

The input of this branch is a concatenation of the one-channel raw depth image $\mathbf{d}_{raw}$ and the two-channel local guidance map $\mathbf{g}$ from the RGB image. Given this input, the encoder downsamples the feature size to $\frac{H}{32} \times \frac{W}{32}$ and expands the feature dimension to 512. The encoder $M(\cdot)$ learns the mapping from the depth map to the depth latent space z as the fused depth feature information for RDF-GAN. The decoding stage applies a set of upsampling

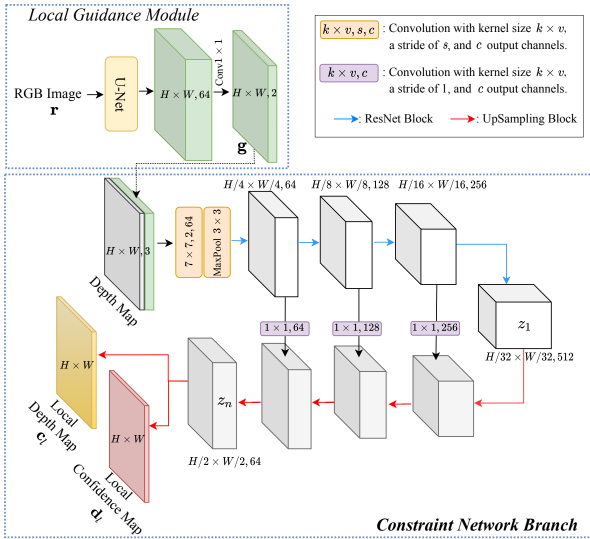

Figure 3. An Illustration of the constraint network.

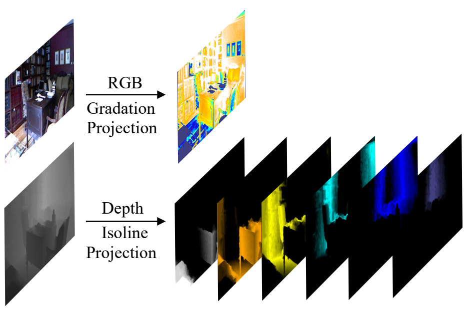

Figure 4. An example of projecting an RGB image to color gradations (top) and projecting a depth map to different depth planes.

blocks to increase the feature resolution with skip connection from the encoder. The output of the decoder is a local depth map and its corresponding local confidence map.

3.2. RDF-GAN Branch

To generate the fine-grained textured and dense depth map, we propose the second branch in our model, which is a GAN-based structure for RGB and depth image fusion. Different from most existing fusion methods that directly concatenate inputs from different domains, our fusion model, named as RDF-GAN, is inspired by the conditional and style GANs [15, 27]. As illustrated in the top-left part of Fig. 2, we use the depth latent vector mapping from incomplete depth image as the input and the RGB image as the condition to generate a dense fused depth prediction and a fused confidence map, and use a discriminator to distinguish the real (ground truth) depth images from generated ones. The generator $G(\cdot)$ has a similar structure to the constraint network. Given the crossponding RGB image $\mathbf{r}$ as the condition, the generator $G(\cdot)$ with the depth latent vector z generates a fused dense depth map $\mathbf{d}_f$ and a fused confidence map $\mathbf{c}_f \in \mathbb{R}^{H\times W\times 1}$ for the scene. The latent vector z propagates the depth information to the RGB image using the proposed W-AdaIN described in Section 3.3. We distinguish the fused depth map $\mathbf{d}_f$ and the real depth image $\mathbf{d}_{gt}$ by the discriminator $D(\cdot)$, whose structure is based on PatchGAN [14]. We adopt the objective function of WGAN [9] for training RDF-GAN. To be more specific, the RDF-GAN loss includes the discriminator loss L_D and the generator loss L_G:

$$L_D = \mathop{\mathbb{E}}_{\mathbf{d}_{\text{raw}}\sim\mathcal{D}_{\text{raw}}}\left[D(G(M(\mathbf{d}_{\text{raw}}))\,|\mathbf{r}]\right] - \mathop{\mathbb{E}}_{\mathbf{d}_{\text{gt}}\sim\mathcal{D}_{\text{gt}}}\left[D(\mathbf{d}_{\text{gt}}|\mathbf{r})\right], \tag{1}$$

$$L_G = \lambda_g L_1(G(M(\mathbf{d}_{\text{raw}})) - \mathop{\mathbb{E}}_{\mathbf{d}_{\text{raw}}\sim\mathcal{D}_{\text{raw}}}\left[D(G(M(\mathbf{d}_{\text{raw}}))\,|\mathbf{r}\right], \tag{2}$$

where d_{raw} and d_{gt} are the raw and ground-truth depth images drawn from the domains $\mathcal{D}_{\text{raw}}$ and $\mathcal{D}_{\text{gt}}$, respectively.

3.3. Feature Fusion Modules

To allow the feature information to be shared across all stages of the two branches, we design the local guidance module and W-AdaIN and apply them in the network.

Local Guidance Module. We adopt U-Net [33] as a feature extractor to produce a local guidance map $\mathbf{g} \in \mathbb{R}^{H\times W\times 2}$ from an RGB image $\mathbf{r} \in \mathbb{R}^{H\times W\times 3}$. The first and the second channels of the local guidance map represent the foreground probability and semantic features, respectively. Therefore, the local guidance module can guide the constraint network to focus on local depth correlations.

W-AdaIN. As shown in Fig. 4, we project depth pixels of the depth map into multiple discretized depth planes, according to the distance between the depth pixels and a pre-defined set of discrete depth values. Local regions are easier to be classified into the same depth plane because they have similar depth values. We also find that similar color gradations in a local region usually have similar depth values. Hence, we propose a W-AdaIN module for fusing the features of RGB and depth images. It is extended from AdaIN [15] and is defined as:

$$\text{W-AdaIN}(z, f_r) = A \cdot y_s \cdot \left(\frac{f_r - \mu(f_r)}{\sigma(f_r)}\right) + B \cdot y_b, \tag{3}$$

where f_r is the feature map of RGB image; $A = \text{Attention}(z)$ and $B = \text{Attention}(f_r)$ are the weight matrices that are generated by the self-attention mechanism [38] on z and f_r, respectively; y_s and y_b are the spatial scaling and bias factors obtained by affine transformations [15] with the latent matrix z; $\mu(\cdot)$ and $\sigma(\cdot)$ are the mean and variance, respectively. By its design, A assigns similar weight values to the regions with similar depth values. Similarly, B smoothes the depth blocks by assigning similar weight values of the local similar color gradations.

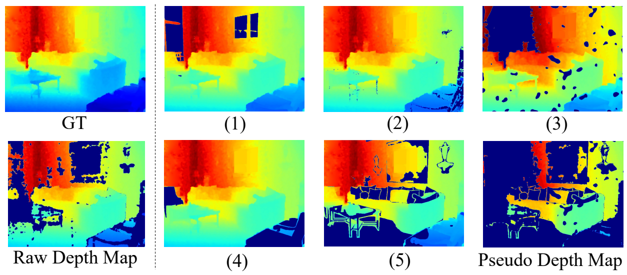

Figure 5. Visualizations of the proposed pseudo depth map and five sampling methods. GT represents the reconstructed (ground-truth) depth map. The shown pseudo depth map is generated from the raw depth image by applying all sampling methods together.

3.4. Confidence Fusion Head

In our framework, either branch has its role. The RDF-GAN branch estimates the missing depth based on the textural features of the RGB image but may produce obvious outliers, i.e., estimation deviations from the raw depth values. The constraint network branch, with an encoder-decoder structure, generates a locally accurate depth map by relying more on valid raw depth information. Hence, we introduce the confidence maps [37] to integrate the depth maps from two branches by a confidence fusion head, which is shown in the right of Fig. 2. We introduce the confidence maps [37] of both branches to assign more attention to reliable depth prediction regions through the learned confidence. In general, the local depth map obtains higher confidences in regions whose raw depth values are more accurate, while the fused depth map has higher confidences in large missing and noisy regions. The sum of the two depth maps weighted by the corresponding confidence maps is the final depth prediction, which is formulated as:

$$d_{pred}(i,j) = \frac{e^{c_l(i,j)} \cdot d_l(i,j) + e^{c_f(i,j)} \cdot d_f(i,j)}{e^{c_l(i,j)} + e^{c_f(i,j)}}. \quad (4)$$

3.5. Pseudo Depth Map for Training

Most existing depth completion methods are trained and evaluated with the random sparse sampling method [18, 23, 29]. The sampled depth map mimics outdoor depth well, but its depth distribution and missing patterns are quite different from the real indoor depth completion scene. The randomly downsampled depth pixels cover almost all areas of the scene, while the missing depth pixels in indoor environments usually form continuous areas. Hence, we propose a set of synthetic methods to produce depth maps for model training, which rely on RGB images and semantic masks to map the raw depth image to reasonable incomplete (pseudo) depth maps. Pseudo depth map mimics the depth missing patterns and is more like real raw depth data than the randomly sampled depth maps.

We design five methods to obtain the pseudo depth map:

(1) *Highlight masking*. We segment the regions of probably specular highlights [1] in RGB images and mask them in raw depth maps.

(2) *Black masking*. We randomly mask the depth pixels whose RGB values are all in [0, 5] (i.e., dark pixels).

(3) *Graph-based segmentation masking*. We mask the probably noisy pixels of depth maps obtained by graph-based segmentations [7] on RGB images.

(4) *Semantic masking*. As depth values for objects with some particular materials are usually missing, we mask one or two objects randomly by their semantic labels and only keep depth pixels on their edges.

(5) *Semantic XOR masking*. We train U-Net [33] on 20% of the training set of RGB images and use the trained model to segment the other RGB images. We mask the depth pixels where the segmentation result and ground-truth are different, i.e., conducting the XOR operation on the segmentation results and the ground-truth to obtain the masking.

Finally, we randomly pick and combine the mask from the above five methods to generate the pseudo depth map, mimicking a more plausible missing depth distribution. The pseudo depth maps are used to train a more robust depth completion model for indoor scenarios. More details can be found in Section 2 of the supplementary.

3.6. Loss Function

We use the L_1 loss on the local depth map and final prediction. The overall loss function is defined as:

$$L_{overall} = L_D + L_G + \lambda_l L_1(\mathbf{d}_l) + \lambda_{pred} L_1(\mathbf{d}_{pred}), \quad (5)$$

where λ_g in Eq. 2, λ_l, and λ_{pred} are weight hyperparameters for different terms in the loss function, which are set to be 0.5, 1, and 10, respectively.

4. Experiments

4.1. Datasets and Metrics

We conducted experiments on two widely-used benchmarks: NYU-Depth V2 [28] and SUN RGB-D [36].

NYU-Depth V2. The NYU-Depth V2 dataset [28] contains pairs of RGB and depth images collected from Microsoft Kinect in 464 indoor scenes. Densely labeled image pairs are split into the training set with 795 images and the test set with 654 images, and each set includes RGB images, raw depth images from sensors, labeled (reconstructed) depth maps, and segmentation masks. Following existing methods, we utilized the unlabeled ~50K images for training and the labeled 654 images in the test set for evaluation. The input images were resized to 320×240 and center-cropped cropped to 304×228.

Setting	Method	RMSE ↓	Rel ↓	$\delta_{1.25}$ ↑	$\delta_{1.25^2}$ ↑	$\delta_{1.25^3}$ ↑
$\mathcal{R} \Rightarrow \mathcal{T}$	DC-BCS [12]	0.271	0.016	98.1	99.1	99.4
	RGB-GU [37]	0.260	0.017	97.9	99.3	99.7
	MS-CHN [19]	0.190	0.018	98.8	99.7	99.9
	DM-LRN [35]	0.205	0.014	98.8	99.6	99.9
	NLSPN [29]	0.153	0.015	98.6	99.6	99.9
	Ours	**0.139**	**0.013**	98.7	99.6	99.9
$\mathcal{R}^* \Rightarrow \mathcal{T}$	Sparse2Dense [23]	0.335	0.060	94.2	97.1	98.8
	CSPN [3]	0.500	0.139	85.7	92.9	96.3
	NLSPN [29]	0.348	**0.043**	93.0	96.7	98.5
	Ours	**0.309**	0.053	93.6	97.6	99.0
$\mathcal{T}^* \Rightarrow \mathcal{T}$	Sparse2Dense [23]	0.230	0.044	97.1	99.4	99.8
	CSPN [3]	0.117	0.016	99.2	99.9	100.0
	3coeff [13]	0.131	0.013	97.9	99.3	99.8
	DGCG [17]	0.225	0.046	97.2	–	–
	DeepLidar [32]	0.115	0.022	99.3	99.9	100.0
	NLSPN [29]	**0.092**	**0.012**	99.6	99.9	100.0
	PRR [18]	0.104	0.014	99.4	99.9	100.0
	Ours	0.103	0.016	99.4	99.9	100.0

Table 1. Quantitative results on the NYU-Depth V2 dataset. $\mathcal{R}$ and $\mathcal{T}$ represent the raw and reconstructed depth map, respectively. $\cdot^*$ represents the random sparse sampling, where Spares2Dense and DGCG in $\mathcal{T}^* \Rightarrow \mathcal{T}$ use 200 pixels and others use 500 pixels.

$\mathcal{R} \Rightarrow \mathcal{T}$	RMSE ↓	Rel ↓	$\delta_{1.25}$ ↑	$\delta_{1.25^2}$ ↑	$\delta_{1.25^3}$ ↑
Sparse2Dense [23]	0.329	0.074	93.9	97.0	98.1
CSPN [3]	0.295	0.137	95.6	97.5	98.4
DeepLidar [32]	0.279	0.061	96.9	98.0	98.4
NLSPN [29]	0.267	0.063	97.3	98.1	98.5
Ours	**0.255**	**0.059**	96.9	98.4	99.0

Table 2. Quantitative results on the SUN RGB-D dataset.

SUN RGB-D. The SUN RGB-D dataset [36] contains 10,335 RGB-D images captured by four different sensors. This dataset, with different scenes and sensors, is diverse and helpful to effectively evaluate model generalization. Besides, its dense semantic annotations and 3D bounding boxes enable the evaluations of more training strategies and downstream tasks. Following the official split, we used 4,845 images for training and 4,659 for testing in 19 major scene categories. We used the refined depth map based on multiple frames [36] as the ground truths for evaluation. The input images were resized to 320×240 and randomly cropped to 304×228.

Evaluation Metrics. We adopted three metrics for the dense depth prediction evaluation: root mean squared error (RMSE), absolute relative error (Rel), and δ_i, which is the percentage of predicted pixels whose relative error is within a relative threshold [23].

4.2. Comparisons with State-of-the-Art Methods

NYU-Depth V2. To draw a comprehensive performance analysis, we set up three different training and evaluation schemes. In the test, we use three different inputs to predict and reconstruct depth maps $\mathcal{T}$ respectively, which are raw depth maps $\mathcal{R}$, sparse depth maps with randomly sampled 500 valid depth pixels in raw depth map $\mathcal{R}^*$, and sparse depth maps with randomly sampled 500 valid depth pixels in reconstructed depth map $\mathcal{T}^*$. For more descriptions of the schemes, please refer to Section 3 in the supplementary. The performance comparison of our method and the other state-of-the-art methods on NYU-Depth V2 are shown in Tab. 1. Given the results, we concluded the following:

- $\mathcal{R} \Rightarrow \mathcal{T}$: We used the pseudo depth maps generated in Section 3.5 as the input to train the proposed model and NLSPN [29]. Meanwhile, we compared with several baselines [12, 19, 35, 37] that are trained in the synthetic semi-dense sensor data [35]. Compared to all the baselines, our proposed method improves significant performance, especially on RMSE and Rel. We selected two representative scenes and visualized our prediction results in the last column of Fig. 6. The model trained by pseudo depth maps produced more accurate and textured depth predictions in the missing depth regions.
- $\mathcal{R}^* \Rightarrow \mathcal{T}$: Following the previous works [3, 18, 23, 29], we used the RGB image and the sparse depth map with randomly sampled depth pixels of raw depth image as the input for training. In the test stage, the input was the same as that for training, and the reconstructed depth map was used as the ground truth. We observed that our model outperformed the baseline with big margins on RMSE. The qualitative results were shown in the second and fourth rows of Fig. 6. Our method accurately predicts the contour of the sofa and smooth windows in red boxes compared to other methods. This proves that our dense depth predictions are well integrated with the textural information of RGB images by the RDF-GAN branch.
- $\mathcal{T}^* \Rightarrow \mathcal{T}$: The setting is consistent with most existing works of depth completion [3, 18, 23, 29]. Our model without any iteration processing is only lower than the NLSPN [32] (but ours is 1.5× faster in inference time than NLSPN). The visualizations shown in the first and third rows of Fig. 6 as well as Fig. 7 further indicate the superiority of our method.
- As shown in green boxes of Fig. 6, the downsampled input from the reconstructed depth map ($\mathcal{T}^* \Rightarrow \mathcal{T}$) reveals ground truth depth values, which is unavailable in practice, to the models. This supported the claim that the raw input setting ($\mathcal{R} \Rightarrow \mathcal{T}$) is more practicable for realistic indoor depth completion.

SUN RGB-D. On SUN RGB-D, we adopted the pseudo depth maps as the input and the raw depth data as the ground truth for training. In the test set, the raw depth image and the depth map synthesized by multiple frames were used as the input and the ground truth, respectively. In Tab. 2, our proposed method achieves the best performance in most metrics. From the visualization results in Fig. 1, our model complements the missing depth regions as much as possible

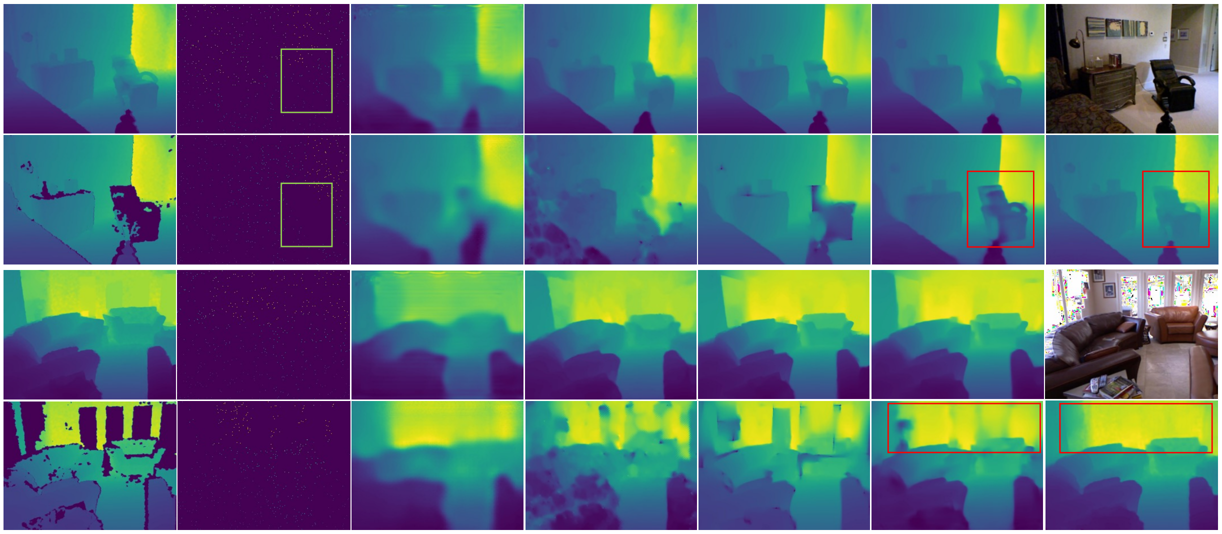

Figure 6. Depth completion comparisons of different methods with different training strategies and inputs on NYU-Depth V2. The first and third rows take sparse samples on reconstructed depth maps as the input ($\mathcal{T}^* \Rightarrow \mathcal{T}$). The second and fourth rows take sparse samples on raw depth maps as the inputs ($\mathcal{R}^* \Rightarrow \mathcal{T}$). The last column shows the result of our model trained with pseudo maps ($\mathcal{R} \Rightarrow \mathcal{T}$).

Figure 7. Depth completion comparisons on NYU-Depth V2 with $\mathcal{T}^* \Rightarrow \mathcal{T}$. Our model recovers more textural details in the red boxes.

Setting	λ_g	λ_l	λ_{pred}	RMSE ↓	Rel ↓	$\delta_{1.25}$ ↑
A	-	-	✓	0.207	0.032	97.8
B	✓	-	✓	0.212	0.038	97.8
C	-	✓	✓	0.174	0.025	98.3
D	0.5	1	10	0.103	0.016	99.4

Table 3. Quantitative comparisons of different L_1 loss settings.

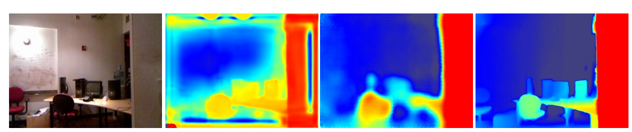

Figure 8. Qualitative comparisons of different L_1 loss settings.

with more detailed texture information for different sensors.

4.3. Ablation Studies

We conducted ablation studies with the setting of $\mathcal{T}^* \Rightarrow \mathcal{T}$ on the NYU-Depth V2 dataset.

Settings of λs. We investigated the effects on model performance in different settings of λs in the loss function, and the results are shown in Tab. 3. We compared the following four settings and found that including all L_1 loss terms leads to the best model. In Setting A, we only calculated the L_1 loss for the final depth prediction, and in Setting B, both L_1 losses for the final depth prediction and fused depth map were calculated. In these two settings, the model overly focused on textural information resulting in generating many local outliers, as shown in Fig. 8(a), and the predicted depth values in many regions had a large deviation from the ground-truth values. In Setting C, we took the L_1 losses for the local depth map and the final depth

Module	Method	RMSE ↓	REL ↓	$\delta_{1.25}$ ↑
Fusion Head	Conv.	0.118	0.022	99.0
	Confidence Fusion	0.117	0.019	99.1
Local Guidance	Concat.	0.113	0.017	99.2
	U-Net	0.107	0.016	99.4
	U-Net (I)	0.106	0.016	99.4
	U-Net (N)	0.101	0.015	99.5
Stage Fusion	IN	0.106	0.016	99.4
	AdaIN	0.110	0.017	99.3
	W-AdaIN	0.103	0.016	99.4

Table 4. Ablation study results for different modules. 'Conv.' means the convolution operation for the concatenation of the outputs from the two branches. 'U-Net (I)' and 'U-Net (N)' represent pre-training with ImageNet and NYU-Depth V2, respectively.

Method	mAP@25	mAP@50
VoteNet [31]	59.07	35.77
Ours+VoteNet [31]	**60.64**	**37.28**
H3DNet [41]	60.11	39.04
Ours+H3DNet [41]	**61.03**	**39.71**

Table 5. Performance comparisons of 3D object detection results with the raw and completed depth maps on SUN RGB-D.

prediction. Although its performance is slightly better, as the model degenerated to the encoder-decoder structure, the depth completion result was shaped towards a blurry depth image, as shown in Fig. 8(b). Calculating L_1 for both branches (Setting D) is the final setting we adopted, which obtained significant improvement in all metrics and generated the reasonable depth prediction, as shown in Fig. 8(c).

Modules. On the basis of the two-branch structure, we evaluated the impact of different modules by comparing them with alternative components. Based on the results shown in Tab. 4, we observe the following:

- For the fusion head, the confidence fusion performs better than the convolution operation (Conv.). In addition, Fig. 9 shows the fused confidence map of an RGB image. The confidence values are high for the foreground objects with richer textural information. It indicates that RDF-GAN makes better use of rich textural information to improve the depth completion.
- Using the local guidance module clearly improves the performance. The modules using U-Net [33] are better than the method of the direct concatenation (Concat.) of RGB and depth images, and pre-training on ImageNet [5] further boosts the performance. By utilizing additional semantic information of the test scenes, i.e., pre-training U-Net with semantic segmentation on NYU-Depth V2, our method can achieve even better performance.
- For the stage fusion modules, W-AdaIN outperforms the others (IN [11] and AdaIN [15]) by a clear margin.

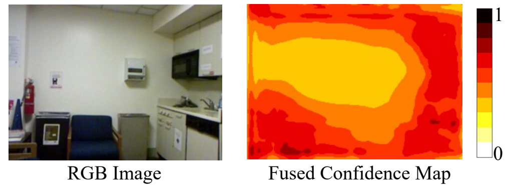

Figure 9. Visualization of the confidence map from RDF-GAN.

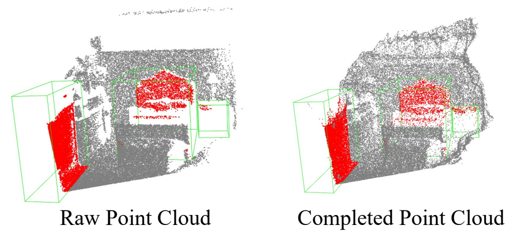

Figure 10. Visualizations of point clouds converted by the depth map. The green boxes are the predicted bounding boxes of the detected objects. The red points are the points inside the boxes.

4.4. Object Detection on the Completed Depth Map

We used the completed depth map as the input of the 3D object detection task on the SUN RGB-D dataset [36] to evaluate the quality of our depth completions. Two SOTA models, VoteNet [31] and H3DNet [41], were used as the detectors. Tab. 5 shows that the two models both obtain a significant improvement with our completed depth map. As shown in Fig. 10, the point cloud converted from the completed depth map contains more points and better covers the shape of the object than the raw depth map. More discussions can be found in Section 4 of the supplementary.

5. Conclusion

In this work, we propose a novel two-branch end-to-end network for indoor depth completion. We design the RDF-GAN model to produce the fine-grained textural depth map and restraint it by a constraint network. In addition, we propose a novel and effective sampling method to produce pseudo depth maps for training indoor depth completion models. Extensive experiments have demonstrated that our proposed solution achieves state-of-the-art on the NYU-Depth V2 and SUN RGB-D datasets.

6. Acknowledgements

This work was done when Haowen Wang took an internship at Midea Group. This work was supported in part by the Funds for International Cooperation and Exchange of NSFC under Grant 61720106007, the Fundamental Research Funds for the Central Universities (2019XD-A03-3), the 111 Project under Grant B18008, and Shanghai Pujiang Program (21PJ1420300).

References

[1] Mirko Arnold, Anarta Ghosh, Stefan Ameling, and Gerard Lacey. Automatic segmentation and inpainting of specular highlights for endoscopic imaging. *EURASIP Journal on Image and Video Processing*, 2010:1–12, 2010. 5

[2] ASUS. Asus xtion. www.asus.com/Multimedia/Xtion_PRO/. 1

[3] Xinjing Cheng, Peng Wang, and Ruigang Yang. Depth estimation via affinity learned with convolutional spatial propagation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 103–119, 2018. 1, 2, 6

[4] Yanhua Cheng, Rui Cai, Zhiwei Li, Xin Zhao, and Kaiqi Huang. Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3029–3037, 2017. 3

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. Ieee, 2009. 3, 8

[6] Dapeng Du, Limin Wang, Huiling Wang, Kai Zhao, and Gangshan Wu. Translate-to-recognize networks for rgb-d scene recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[7] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision (IJCV)*, 59(2):167–181, 2004. 5

[8] Yanping Fu, Qingan Yan, Long Yang, Jie Liao, and Chunxia Xiao. Texture mapping for 3d reconstruction with rgb-d sensor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1

[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 4

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[11] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 8

[12] Yu-Kai Huang, Tsung-Han Wu, Yueh-Cheng Liu, and Winston H. Hsu. Indoor depth completion with boundary consistency and self-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct 2019. 1, 2, 6

[13] Saif Imran, Yunfei Long, Xiaoming Liu, and Daniel Morris. Depth coefficients for depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 6

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 4

[15] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 3, 4, 8

[16] Leonid Keselman, John Iselin Woodfill, Anders Grunnet-Jepsen, and Achintya Bhowmik. Intel realsense stereoscopic depth cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 1

[17] Byeong-Uk Lee, Hae-Gon Jeon, Sunghoon Im, and In So Kweon. Depth completion with deep geometry and context guidance. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3281–3287. IEEE, 2019. 1, 2, 6

[18] Byeong-Uk Lee, Kyunghyun Lee, and In So Kweon. Depth completion using plane-residual representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13916–13925, June 2021. 2, 5, 6

[19] Ang Li, Zejian Yuan, Yonggen Ling, Wanchao Chi, shenghao zhang, and Chong Zhang. A multi-scale guided cascade hourglass network for depth completion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020. 6

[20] Bing Li, Juan Pablo Munoz, Xuejian Rong, Qingtian Chen, Jizhong Xiao, Yingli Tian, Aries Arditi, and Mohammed Yousuf. Vision-based mobile indoor assistive navigation aid for blind people. *IEEE Transactions on Mobile Computing (TMC)*, 18(3):702–714, 2019. 1

[21] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 716–723, 2014. 2

[22] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. In *NIPS*, 2017. 2

[23] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 4796–4803. IEEE, 2018. 1, 2, 5, 6

[24] Jiayi Ma, Wei Yu, Pengwei Liang, Chang Li, and Junjun Jiang. Fusiongan: A generative adversarial network for infrared and visible image fusion. *Information Fusion*, 48:11–26, 2019. 2, 3

[25] Michael Maire, Takuya Narihira, and Stella X Yu. Affinity cnn: Learning pixel-centric pairwise relations for figure/ground embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 174–182, 2016. 2

[26] Microsoft. Kinect for windows. https://developer.microsoft.com/en-us/windows/kinect/. 1

[27] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2, 3, 4

[28] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*, 2012. 5

[29] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *European Conference on Computer Vision (ECCV)*, pages 120–136. Springer, 2020. 1, 2, 5, 6

[30] Seong-Jin Park, Ki-Sang Hong, and Seungyong Lee. Rdfnet: Rgb-d multi-level residual feature fusion for indoor semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4980–4989, 2017. 3

[31] 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 International Conference on Computer Vision (ICCV)*, pages 9277–9286, 2019. 8

[32] Jiaxiong Qiu, Zhaopeng Cui, Yinda Zhang, Xingdi Zhang, Shuaicheng Liu, Bing Zeng, and Marc Pollefeys. Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3313–3322, 2019. 1, 2, 6

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4, 5, 8

[34] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng. Learning depth from single monocular images. In *NIPS*, 2005. 2

[35] Dmitry Senushkin, Mikhail Romanov, Ilia Belikov, Nikolay Patakin, and Anton Konushin. Decoder modulation for indoor depth completion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2021, Prague, Czech Republic, September 27 - Oct. 1, 2021*, pages 2181–2188. IEEE, 2021. 6

[36] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 567–576, 2015. 1, 5, 6, 8

[37] Wouter Van Gansbeke, Davy Neven, Bert De Brabandere, and Luc Van Gool. Sparse and noisy lidar completion with rgb guidance and uncertainty. In *2019 16th international conference on machine vision applications (MVA)*, pages 1–6. IEEE, 2019. 5, 6

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 4

[39] Qingxiong Yang. Stereo matching using tree filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 37(4):834–846, 2014. 2

[40] Chongzhen Zhang, Yang Tang, Chaoqiang Zhao, Qiyu Sun, Zhencheng Ye, and Jürgen Kurths. Multitask gans for semantic segmentation and depth completion with cycle consistency. *IEEE Transactions on Neural Networks and Learning Systems*, 32(12):5404–5415, 2021. 2

[41] Zaiwei Zhang, Bo Sun, Haitao Yang, and Qixing Huang. H3dnet: 3d object detection using hybrid geometric primitives. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 311–329. Springer, 2020. 8

[42] Yiqin Zhao and Tian Guo. Pointar: Efficient lighting estimation for mobile augmented reality. In *European Conference on Computer Vision (ECCV)*, pages 678–693. Springer, 2020. 1