Learning to Reconstruct and Understand Indoor Scenes From Sparse Views

Jingyu Yang, Senior Member, IEEE, Ji Xu, Kun Li, Member, IEEE, Yu-Kun Lai, Member, IEEE, Huanjing Yue, Member, IEEE, Jianzhi Lu, Hao Wu, and Yebin Liu, Member, IEEE

Abstract—This paper proposes a new method for simultaneous 3D reconstruction and semantic segmentation for indoor scenes. Unlike existing methods that require recording a video using a color camera and/or a depth camera, our method only needs a small number of (e.g., 3–5) color images from uncalibrated sparse views, which significantly simplifies data acquisition and broadens applicable scenarios. To achieve promising 3D reconstruction from sparse views with limited overlap, our method first recovers the depth map and semantic information for each view, and then fuses the depth maps into a 3D scene. To this end, we design an iterative deep architecture, named IterNet, to estimate the depth map and semantic segmentation alternatively. To obtain accurate alignment between views with limited overlap, we further propose a joint global and local registration method to reconstruct a 3D scene with semantic information. We also make available a new indoor synthetic dataset, containing photorealistic high-resolution RGB images, accurate depth maps and pixel-level semantic labels for thousands of complex layouts. Experimental results on public datasets and our dataset demonstrate that our method achieves more accurate depth estimation, smaller semantic segmentation errors, and better 3D reconstruction results over state-of-the-art methods.

Index Terms—3D reconstruction, deep learning, semantic segmentation, indoor scenes, sparse views

I. INTRODUCTION

With the increasing demand for indoor navigation, home/office design, and augmented reality, indoor 3D reconstruction and understanding have become active topics in computer vision and graphics. Existing reconstruction methods can be broadly categorized into two groups. The first group scans indoor scenes with an integrated depth camera based on either time-of-flight (ToF) or structured light sensing. The pioneering KinectFusion [1] presents a detailed workflow using Kinect for indoor reconstruction. It was more recently extended by BundleFusion [2] which achieves state-of-the-art results in real-time 3D reconstruction. However, the quality of captured depth maps is affected by the limitations of sensing techniques, e.g., limited sensing distances, low resolution, invalid measurements for specular areas, and/or significant amount of noise. On the contrary, RGB cameras provide much higher quality of color images, and are far more available particularly on smart phones. It is therefore interesting to study 3D scene reconstruction using a color camera, which however is challenging due to lack of depth information. Simultaneous localization and mapping (SLAM) [3] and structure from motion (SFM) [4] are two popular approaches for feature-based point cloud 3D reconstruction on-line and off-line, respectively. However, these feature-based methods require rich textures in the scene, and are therefore difficult to obtain dense point clouds. All the above methods require consecutive frame tracking or dense view capturing.

In this paper, we propose a new indoor-scene 3D reconstruction and semantic segmentation method using color images at several uncalibrated sparse views. The first challenge is to reconstruct a dense 3D scene from sparse views with limited overlap, which is practically degenerated into the task of monocular depth estimation. The second challenge, hence, is the non-rigid transformation between sparse views brought in by the inaccurate depth estimated from the single color image for each view. To address these problems, we design an iteratively optimized deep framework, named IterNet, for simultaneous depth map recovery and semantic segmentation for each view, where the two tasks help improve each other. To estimate complex transformations between sparse views, we further develop a joint global and local alignment method.

Figure 1. An example of 3D reconstruction by the proposed IterNet from five color images at sparse views: (a) one of the input RGB images, (b) ground-truth depth map, (c) ground-truth semantic segmentation, (d) our reconstructed 3D model using (e) the estimated depth map and (f) semantic labels. This example is part of test data.
to fuse estimated depths with the help of semantic information, which integrates geometry, photometry and semantic information in a coarse-to-fine manner.

Depth recovery and semantic segmentation from images are ill-posed, and it is essential to learn from high-quality training data. For indoor scene understanding, a number of datasets have been made publicly available. Real-world datasets, such as NYUv2 [5], SUN RGB-D [6], and ScanNet [7], need a lot of manual labor to annotate the labels and contain unavoidable noise in depths assumed as ground-truths, while synthetic datasets [8], [9] are difficult to generate photorealistic RGB images and usually have limited layouts and image resolution. To our best knowledge, no existing dataset can provide photorealistic RGB images, accurate depth maps, pixel-level semantic labels, and thousands of complex layouts at the same time. To address this, we build our IterNet RGB-D dataset with these features.

Experimental results on both public datasets and our dataset demonstrate that our method outperforms state-of-the-art methods on depth estimation, semantic segmentation, and sparse-view 3D reconstruction. Figure 1 gives an example of our IterNet RGB-D dataset and the reconstructed 3D model with estimated semantics using our IterNet. We will make the code and the dataset available online for research purposes.

In summary, our method is an integrated work that includes 1) a depth estimation method from a single color image; 2) a semantic segmentation method from a single view; and 3) a multi-view reconstruction method from sparse views. They jointly solve a challenging problem of 3D reconstruction and understanding from uncalibrated sparse views. As a result, our method is applicable to more scenarios than previous approaches that rely on textures and/or geometries of dense datasets. This is particularly useful for the training and evaluation of data-hungry learning-based approaches.

We design a novel iterative learning-based method for joint depth estimation and semantic segmentation. Unlike previous learning-based methods that require two-way input (color+depth or color+semantic), our IterNet needs only a single color image for each view, and the depth-estimation branch and semantic segmentation branch are iteratively concatenated to help improve each other. The parallel context layer and atrous spatial pyramid pooling are incorporated to improve the LSTM (Long Short-Term Memory) module for better semantic segmentation. This architecture is not restricted to these tasks we address here, and is also applicable to other related tasks such as object/part parsing.

We propose a joint global and local registration method to fuse different sparse perspectives. A seven-dimensional (7D) feature descriptor consisting of 3D location, photometric, and semantic information is designed for global registration. To improve the performance, a local alignment strategy is applied to semantic objects. This coarse-to-fine alignment is robust to sparse views and the errors of monocular depth estimation.

II. RELATED WORK

Indoor Datasets. Naseer et al. [12] gave a comprehensive overview of indoor scene understanding in 2.5D/3D. NYU-Depth dataset [5] captured by Microsoft Kinect is the first dataset of this category. SUN RGB-D dataset [6] captured by four different RGB-D sensors contains 10,335 indoor images with dense annotations. Armeni et al. [10] provided Building Parser dataset with instance level semantic and geometric annotations. Matterport3D [11] contains 10,800 panoramic images covering 360° viewpoints captured by a Matterport camera. ScanNet [7] is a 3D reconstruction dataset with 2.5 million frames obtained from 1,513 scans. The depth maps in these real-world datasets usually contain noise and missing areas and need a lot of manual effort for label annotation. Hence, synthetic datasets are constructed for easy generation and accurate ground truth. SUNCG [8] is a densely annotated large-scale indoor dataset, but the rendered RGB images are not photorealistic and RGB-D videos are not available. SceneNet RGB-D [9] provides pixel-level annotations and photorealistic RGB images, but the number of layouts is limited. Table I lists the main features of various publicly available 2.5D indoor datasets including our one. Our IterNet RGB-D dataset provides a total of 12,856 photorealistic images for thousands of layouts, and has higher image resolutions: 1280 × 960 or 1280 × 720, covering more indoor scenes. Moreover, our dataset provides absolute depth maps and pixel-level semantic segmentation that are more precise and accurate. Compared with other datasets, the indoor scenes covered by our dataset are more general and more complex.

Monocular Depth Estimation. In computer vision, monocular depth estimation has been a long-standing topic in last decades. Previous approaches mainly relied on handcrafted features [13], statistical priors [14] or graphical models [15]. With the development of deep learning, more recent approaches were based on Convolutional Neural Networks (CNNs). For instance, Eigen et al. [16] proposed a multi-scale CNN for depth estimation, which achieves promising results. Considering the correlation between tasks, Wang et al. [17] introduced a CNN for joint depth estimation and semantic segmentation. Xu et al. [18] proposed a multi-task approach for depth estimation via cross-modal interactions. Recently, the attention mechanism has been extensively applied to various learning tasks, and Xu et al. [19] proposed a structured attention scheme for depth estimation to fuse features of different scales. Xu et al. [20] utilized a continuous Conditional Random Field (CRF) to combine multi-scale features. Our method also uses CRFs as basic building blocks for depth inference, but further integrates semantic information in an iterative manner.

Semantic Segmentation. As an extension of image classification, semantic segmentation assigns pixel-wise labels of object categories for the input image. It is challenging due to randomness of object distribution, color ambiguity,
Table I

Comparison between various indoor datasets. IterNet RGB-D is our proposed dataset. ×: not included, ✓: included, -: relevant information not available.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Synthetic</td>
<td>Synthetic</td>
<td>Synthetic</td>
</tr>
<tr>
<td>Images/Scans</td>
<td>1449</td>
<td>10K</td>
<td>70K</td>
<td>194K</td>
<td>1513</td>
<td>130K</td>
<td>5M</td>
<td>12,856</td>
</tr>
<tr>
<td>Layouts</td>
<td>464</td>
<td>-</td>
<td>270</td>
<td>90</td>
<td>1513</td>
<td>45,622</td>
<td>57</td>
<td>3214</td>
</tr>
<tr>
<td>Object Classes</td>
<td>894</td>
<td>800</td>
<td>13</td>
<td>40</td>
<td>50</td>
<td>84</td>
<td>255</td>
<td>333</td>
</tr>
<tr>
<td>RGB</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Depth</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Semantic Label</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>RGB Texturing</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Real</td>
<td>Not Photorealistic</td>
<td>Photorealistic</td>
<td>Photorealistic</td>
</tr>
<tr>
<td>Image Resolution</td>
<td>640×480</td>
<td>640×480</td>
<td>1080×1080</td>
<td>1280×1024</td>
<td>640×480</td>
<td>640×480</td>
<td>320×240</td>
<td>1280×960, 1280×720</td>
</tr>
</tbody>
</table>

The table above provides a comparison between various indoor datasets, including NYUv2, SUN RGB-D, Building 3D, Matterport 3D, ScanNet, SUNCg, SceneNet RGB-D, and IterNet RGB-D. The datasets vary in terms of year, type (real or synthetic), images/scans, layouts, object classes, RGB, depth, semantic label, RGB texturing, and image resolution. The IterNet RGB-D dataset is our proposed dataset, which includes new features and improvements compared to existing datasets.

Poor illumination, and occlusion. Deng et al. [21] proposed a transformation function and a discriminative classifier that maximize the mutual information of data and their labels in the latent space to reduce the uncertainties, i.e., missing and noisy labels. Alternative approaches are typically based on CNNs. Long et al. [22] chose to stand on the popular Fully Convolutional Network (FCN) architecture, which inspired most of the subsequent approaches on semantic segmentation. Using the publicly available RGB-D datasets, some methods attempted to use depth information for better segmentation, but limited to a single RGB image. Li et al. [23] constructed HHA images [24] for the depth channel through geometric encoding before feeding them to the network and used the Long Short-Term Memory (LSTM) network to fuse two different types of features. Ma et al. [25] predicted semantic segmentation from RGB-D sequences, but the method is inapplicable to sparse views. Cong et al. [26] presented saliency detection with comprehensive information, e.g., depth cue and inter-image correspondence. Our method exploits depth information to help improve semantic segmentation, but the depth is estimated from the input color image instead of directly captured by a dedicated depth sensor. We propose an iterative method for joint estimation of the depth and semantic segmentation, which benefit each other.

Indoor Scene 3D Reconstruction. Indoor Scene 3D Reconstruction from a color video or multi-view color images is a challenging and active topic. Given a color video, most structure from motion (SFM) methods [27] recovered the 3D structure by estimating the motion of the cameras corresponding to the frames. However, it is difficult to achieve dense and accurate reconstruction. Given multi-view color images with calibrated camera parameters, multi-view stereo (MVS) methods [28] can achieve more accurate 3D reconstruction. But they require adjacent views to have sufficient overlap and cannot work well with sparse views. COLMAP [29], [30] provides a pipeline containing both SFM and MVS with graphical and command-line interfaces. When the views of images are very sparse, the depth of each image can be estimated and fused together using iterative closest point (ICP) like registration methods [31]. However, it is difficult to achieve accurate depth estimation from individual color images which increases the difficulties of ICP fusion. Saxena et al. [32] proposed a novel method for 3D reconstruction from sparse views, but it only works well for building-like outdoor scenes and cannot generate semantics. Learning-based methods, e.g., MVSNet [33] and DeepMVS [34], output the depth of a specific frame based on a color multi-view sequence, but they cannot deal with sparse views. In this paper, we design IterNet to estimate a more accurate depth map with the help of semantic segmentation, and propose a joint global and local registration method to better achieve indoor scene 3D reconstruction from sparse views.

III. PROPOSED METHOD

In this section, we first introduce our IterNet RGB-D dataset in Section III-A, and then describe the technical details of IterNet for iterative joint depth estimation and semantic segmentation in Section III-B. The joint global and local multi-view reconstruction method is presented in Section III-C. Figure 2 illustrates the workflow of our method.

A. Dataset

Different from other synthetic datasets [8], [9], our dataset is generated by a third-party platform which includes various real-life house styles, real prototype rooms designed by professional designers, and rich material profiles. For fast photorealistic rendering, we use image splitting and recombinative to achieve distributed rendering on a cluster of 32 servers. Each server has two CPUs with a total of 32 cores and 64 threads. The average rendering time of a 1280×960 image is about 90 seconds. To render 12,856 images, it takes about 321 hours. In terms of rendering quality, in addition to modeling the direct illumination of the light source in the scene, the illumination reflected by other objects, known as Global Illumination (GI), is also taken into consideration. The Brute Force (BF) algorithm [35] based on path tracking is adopted. The number of samples per pixel is up to 512, and varies for different scenes. The noise level (standard deviation) is set below 0.05. A lower noise level would yield better...
Figure 2. Illustration of the proposed method for indoor 3D reconstruction and understanding. The blue Module refers to our IterNet for iterative joint depth estimation and semantic segmentation (Section III-B). With the help of semantic segmentation, we use our proposed joint global and local registration method to reconstruct a 3D scene with semantic information from sparse views (Section III-C).

Figure 3. Some examples of different scenarios in our dataset. From top to bottom: color images, ground-truth depth maps, and ground-truth semantic segmentations.

B. IterNet: Iterative CNN for Joint Depth Estimation and Semantic Segmentation

Network Architecture. The proposed IterNet is a multi-task deep CNN mainly consisting of two parts: the depth estimation sub-network and the semantic segmentation sub-network, as shown in Figure 4.

Our depth estimation sub-network is inspired by the monocular depth estimation method [20] which uses a continuous conditional random field (CCRF) to combine multi-scale features. Different from [20], we add a semantic branch with an encoder-decoder structure to extract semantic features. Then, multi-scale RGB features and semantic features are integrated by a CCRF module to improve boundary accuracy. The RGB branch consists of a front-end base network and a refinement network combined with several CCRF modules. Together with semantic information, the output of the RGB branch is fed into a CCRF module to generate the depth map which in turn serves as the input of the semantic segmentation sub-network in the next iteration.

In the semantic segmentation sub-network, we use the Long Short-Term Memorized Context Fusion (LSTM-CF) [23] model, which is capable of fusing contextual information from multiple sources (i.e. photometric and depth channels). Instead of the original serial vertical and horizontal context layers, we adopt a parallel context layer and a direct fusion scheme to take advantage of the estimated depth map from the RGB branch. An Atrous Spatial Pyramid Pooling (ASPP) [37] is added as a multi-scale feature extractor using multiple parallel filters with different sampling rates. In addition, the depth map is first encoded into an HHA image [24] using geocentric encoding before passing through the segmentation sub-network.

Training and Testing. Given a dataset of RGB-Depth-Semantic triplets, our aim is to train the designed network for joint depth and semantic estimation. The depth estimation sub-network and semantic segmentation sub-network are applied alternately to boost the performance. Instead of jointly training the two sub-networks, we train the depth estimation and semantic segmentation sub-networks sequentially for flexible boosting. Taking the depth estimation sub-network as an example, we train the upper branch and the lower branch with RGB-Depth pairs and Semantic-Depth pairs, respectively. The depth estimation sub-network is then fine-tuned with the RGB-Depth-Semantic triplets. The semantic segmentation sub-network is trained in a similar way.

At the test stage, since each sub-network expects the output of the other sub-network as part of input, we use the following strategy. We need an initialized semantic segmentation or depth estimation which can be easily obtained by disabling one of the branches in the original network structure. For example, if we want to obtain an initial depth estimation for semantic segmentation, we disable the semantic segmentation branch...
in the depth estimation sub-network and then extract features from RGB branches as an initial depth. We then alternately run the two sub-networks, with the output of one sub-network used as input for the other sub-network. The additional depth information helps improve semantic segmentation, and the semantic segmentation in turn contributes to improved depth estimation. In practice, we find that there is no significant improvement after 3 iterations, which shows quick convergence.

Implementation Details. The proposed approach is implemented on the Caffe framework [38] and runs on a computer with an Nvidia GTX 1080 Ti graphics card (11GB). For the depth estimation sub-network, the learning rate is initialized at $10^{-11}$ and decreases by 10% for every 30 epochs. The batch size is set to 16. The momentum and the weight decay are set to 0.9 and 0.0005, respectively. The semantic segmentation sub-network follows the same training rules, but the learning rate is initialized at $10^{-5}$. The parameters of the learning rate decreases by $10^{-5}$. The learning rate is initialized at 0.005, respectively. The learning rate decreases by 10% for every 20 epochs. For the fine-tuning of each sub-network, the initial learning rates are set to $10^{-12}$ and $10^{-5}$ for depth estimation and semantic segmentation, respectively. The batch size, momentum and weight decay remain the same as the pretraining.

C. Joint Global and Local Reconstruction

After depth estimation and semantic segmentation for the image of each view, we reconstruct the whole 3D scene by fusing the depth maps of different views. The straightforward way is to use the ICP algorithm to align the partial 3D models from different perspectives, but would be difficult to achieve satisfactory alignment. First, the depth maps are obtained by a monocular depth estimation network instead of depth sensors like Kinect, and contain significant amount of non-statistical errors. It is therefore insufficient to align two depth point clouds with just one rigid transformation. Second, for sparse perspectives, two adjacent views have limited overlap, and are difficult to align by the standard ICP algorithm. Hence, we propose a new joint global and local registration method by exploiting photometric and semantic information to improve the reconstruction quality.

Before 3D fusion, we filter the noisy 3D points based on the plane constraint similar to [39]. After depth estimation and semantic segmentation, each view now contains three components: color image $C_i$, depth map $D_i$, and semantic segmentation $S_i$, where $i$ denotes view index. Let $X = \{X_i\} = \{(C_i, D_i, S_i)\}_{i=1}^{N}$ be the set of color-depth-semantic triplets for $N$ sparse views. We align all the depth point clouds sequentially with the previous registration result used as the next target model. Each alignment has two stages, namely the global alignment and local alignment.

Global Alignment. Taking the point cloud reconstructed from the previous $i-1$ views as the target, our goal for global alignment is to find an optimal global rigid transformation $T_i = \{R_i, t_i\}$ for view $i$, where $R_i$ and $t_i$ denote the rotation and translation, respectively. Specifically, we first convert the depth map $D_i$ into a point cloud denoted by $P_i = \{p_j\}_{j=1}^{n_i}$, where $p_j$ denotes a 3D point and $n_i$ represents the total number of points at view $i$. Our global registration stands on the ICP-type framework which alternates two steps until convergence. The transformation is initialized with a $4 \times 4$ identity matrix. Let $P'_{i-1} = \{p'_j\}_{j=1}^{n'_{i-1}}$ be the target point cloud containing all the $n'_{i-1}$ fused points from previous $i-1$ views, the first step finds for each point $p_k \in P_i$ its corresponding point $p'_k \in P'_{i-1}$ if possible, and the second step optimizes the transformation $T_i$ by aligning the point cloud $P_i$ at view $i$ to the fused point cloud $P'_{i-1}$.

In the first step, we incorporate the additional photometric and semantic information for more accurate matching. To this end, we lift each point $p_k \in P_i$ from the 3D space to a 7-dimensional (7D) point $\tilde{p}_k \in \tilde{P}_i$ in a 7D space by including color and semantic information. Specifically, $\tilde{p}_k \triangleq \{\tilde{p}_k(s)\}_{s=1}^{7}$ include the 3D position $\{\tilde{p}_k(s)\}_{s=1}^{3}$, RGB color $\{\tilde{p}_k(s)\}_{s=4}^{5}$, and semantic label $\tilde{p}_k(s)$. Similarly, the point $p'_j \in P'_{i-1}$ is lifted to a 7D point $\tilde{p}'_j \in \tilde{P}'_{i-1}$. Our global registration method first finds the corresponding point $\tilde{p}'_k \in \tilde{P}'_{i-1}$ for each point $\tilde{p}_k$ in $\tilde{P}_i$ by the following optimization:

$$\tilde{k} = \arg \min_{j \in \{1, 2, \ldots, n'_{i-1}\}} \|\tilde{p}_k - \tilde{p}'_j\|_2^2.$$  \hspace{1cm} (1)

The weighted Euclidean distance, $\|\tilde{p}_k - \tilde{p}'_j\|_w$, between $\tilde{p}_k$ and
\( \hat{p}_j \) in the 7D space is defined as
\[
\| \hat{p}_k - \hat{p}_j' \|_w^2 = \sum_{s=1}^{3} (\hat{p}_k - \hat{p}_j'(s))^2 \\
+ w_1 \sum_{s=4}^{6} (\hat{p}_k - \hat{p}_j'(s))^2 \\
+ w_2 (\hat{p}_k(7) - \hat{p}_j'(7))^2,
\]
where \( w_1 \) and \( w_2 \) are weights to balance the importance of geometric, photometric and semantic information. They are set to be \( w_1 = 0.1 \) and \( w_2 = 10 \) in our experiments.

Due to limited overlap, not all the points in \( \mathcal{P}_k \) have their corresponding points in \( \mathcal{P}_k' \). We reject \( \hat{p}_k' \) if the matching error is larger than a threshold. In our implementation, this threshold is set to 5cm. Let \( \mathcal{C}_k = \{ p_k, \hat{p}_k' \} \) be the set of retained correspondences. In the second step, since photometric and semantic matching errors are independent of rigid transformations, we use a standard ICP algorithm [31] to find the transformation between the two point clouds:
\[
(R, t) = \arg \min_{R,t} \sum_{(p_k, \hat{p}_k') \in \mathcal{C}_k} \| \hat{p}_k' - R \hat{p}_k - t \|_2^2.
\]

Local Alignment. The global registration on the 7D feature space is able to achieve coarse alignment, but still cannot handle non-rigid local deformation, e.g., due to non-statistical errors in monocular depth estimation. To address this problem, we further propose a local registration strategy to refine the previous coarse estimation, similar to coarse-to-fine refinement. Specifically, we first extract local point sets from the original point cloud according to their semantic labels, and then register each of them using the above method. Note that in this case, a subset of points from one view is only matched to subsets of points with the same semantic label. Therefore, when finding the matched point, the semantic difference term in Eq. (2) is always zero. For each local set, once it is aligned, we fuse the registered parts from different views by averaging
3D positions of overlaps to mitigate the influence of noise. The key for our joint global and local registration method is to use multiple transformations to register sparse views with coarse-to-fine refinement, rather than just one single transformation, which is more robust to the noise and outliers in the monocular depth estimation.

IV. EXPERIMENTAL RESULTS
A. Ablation Study

We compare the full model with the one without semantic segmentation and the one without depth estimation in Table II. It can be seen that the full model has achieved the best performance. Figure 5 shows the fusion results of an ICP matching method [31], 4PCS [40], the global alignment using the estimated depth without the help of semantic branch, and our proposed joint global and local registration method. The red boxes highlight the areas that are difficult to align with other methods, and the green boxes indicate features lost by our method. The results show that some misalignments occur in local areas for standard ICP methods. Our joint global and local alignment method cannot ensure that all the features are captured and aligned, such as the curtain on both sides of the window (in the green boxes). However, compared with other alignment methods, our method achieves better fusion result in terms of both global structure and local details.

Our iterative scheme in IterNet usually converges to promising results after three iterations and is stable for various images. Figure 6 shows the rapid decreasing of average RMS (root mean squared) errors of depth estimation over all the test images with respect to the number of iterations. The average pixel accuracy of semantic segmentation over all the test images also increases sharply to the stationary point in three iterations. No significant improvement for both depth estimation and semantic segmentation is observed beyond three iterations.

To study and verify the effectiveness of IterNet in depth estimation, we compare it with two recent backbone archi-
Figure 7. Comparison of depth estimation with two different network architectures.

Figure 8. Depth estimation results on NYUv2 dataset (top two rows) and our dataset (bottom row). From left to right are the input RGB images, the ground-truths depth and the depth results estimated by Eigen et al. [16], Xu et al. [20], Xu and Wang [19], and our method.

Table II
Ablation results on our dataset. F-S: full model without semantic; F-D: full model without depth; F: full model.

<table>
<thead>
<tr>
<th>Method</th>
<th>F-S</th>
<th>F-D</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>rel (lower is better)</td>
<td>0.176</td>
<td>-</td>
<td>0.136</td>
</tr>
<tr>
<td>log10 (lower is better)</td>
<td>0.088</td>
<td>-</td>
<td>0.062</td>
</tr>
<tr>
<td>rms (lower is better)</td>
<td>1.012</td>
<td>-</td>
<td>0.507</td>
</tr>
<tr>
<td>P-acc. (%) (higher is better)</td>
<td>-</td>
<td>67.35</td>
<td>75.54</td>
</tr>
<tr>
<td>M-acc. (%) (higher is better)</td>
<td>-</td>
<td>68.29</td>
<td>74.49</td>
</tr>
<tr>
<td>IoU(%) (higher is better)</td>
<td>-</td>
<td>54.21</td>
<td>63.98</td>
</tr>
</tbody>
</table>

Figure 6. Convergence curves of the proposed IterNet for NYUv2 dataset [5] and our dataset (averaged over all test images in each dataset).

B. Depth Estimation

We compare our approach with several state-of-the-art methods on NYUv2 dataset [5] in Table IV. We use 795 images for training and other 654 images for testing as other methods did. We also use the same raw data as other methods and adopt data augmentation (finally 4770 images for training) to avoid the over-fitting problem. Let \( d_i \) and \( d_i^* \) denote the predicted depth value and the ground-truth value for pixel \( i \), respectively, and \( P \) represent the total number of pixels. Referring to previous work [16], [17], [41], we evaluate the depth estimation results with the following metrics: (1) mean relative error (rel): \( \frac{1}{P} \sum_i \frac{|d_i - d_i^*|}{d_i^*} \); (2) root mean squared error (rms): \( \sqrt{\frac{1}{P} \sum_i (d_i - d_i^*)^2} \); (3) mean log10 error (log10):
Figure 9. Depth estimation results on NYUv2 dataset [5] (a, b, c) and SUN RGB-D dataset [6] (d, e, f) using our model trained by our dataset. From top to bottom are the input color images, the ground truths, and our estimated depths.

Figure 10. Semantic segmentation results on NYUv2 dataset (top two rows) and our dataset (bottom row). From left to right are the input RGB images, the ground-truths and the results estimated by FCN [22], Chen et al. [37], Li et al. [23], Zhao et al. [50], DANet [51] and our method.

\[ \frac{1}{n} \sum \left| \log_{10}(d_i) - \log_{10}(d^*_i) \right|; \] and (4) accuracy with threshold \( t \): percentage (%) of \( d_i^* \) subject to \( \delta = \max(\frac{d_i}{d^*_i}, \frac{d^*_i}{d_i}) < t \). Our results are averaged over three test trials. The results of the compared methods are quoted from their papers. As shown in Table IV, our method outperforms thirteen competing methods in all metrics, and is comparable to PAD-Net [18] which has a more complex network structure and requires ground-truth contours and normals as part of labels.

We also quantitatively evaluate some methods with their provided code on our IterNet RGB-D dataset. As shown in Table VI, our method achieves the most accurate depth estimation on all the metrics. Figure 8 gives some visual comparison results on NYUv2 dataset [5] and our dataset.

Figure 9 shows some depth estimation results for real indoor scenes from the NYUv2 dataset [5] and SUN RGB-D dataset [6] without finetuning. The results demonstrate that our model trained on our dataset has promising generalizability to other datasets.

Table III reports the running time results of our proposed depth estimation module and other competing methods. All the methods are run on the same desktop equipped with an Intel Xeon 2.10GHz CPU, 64GB RAM, and an NVIDIA GeForce GTX 1080Ti GPU. Requiring three iterations for convergence, our network takes longer time particularly at the GPU mode.

C. Semantic Segmentation

To evaluate the performance of semantic segmentation, we use NYUv2-40 dataset [22] in which all objects in the NYUv2 dataset [5] are divided into 40 categories. We use the same training and testing data as other methods and adopt three metrics in percentage (%): pixel accuracy, mean accuracy, and Intersection over Union (IoU). As shown in Table VII, our inferred semantic segmentation results outperform those state-of-the-art methods. We also quantitatively evaluate some recent work that provide source code on our IterNet RGB-D dataset. Results in Table VIII show that our method also achieves the best performance. Figure 10 presents some visual
comparison results on NYUv2-40 dataset and our dataset mapped into 87 categories. Being consistent with the quantitative results in Table VII and Table VIII, our approach generates more accurate semantic segmentation results on both real dataset (NYUv2) and synthetic dataset (IterNet RGB-D) than state-of-the-art methods. More visual comparison results for semantic segmentation in Figure 11 and Figure 12 show that our approach generates more accurate semantic segmentation on both real dataset (NYUv2) and synthetic dataset (IterNet RGB-D) than other four competing methods.

Table V reports the running times of different semantic segmentation methods. Similar to the depth estimation module, the semantic segmentation module also takes longer time, but still at the same order of magnitude. As future work, we will investigate more efficient network architectures (e.g. lightweight network modules) to reduce the running times.

**D. Multi-view Reconstruction**

In Figure 13, we evaluate multi-view 3D reconstruction performance of the proposed method on NYUv2 dataset [5] and our dataset using three wide-baseline views, compared with four state-of-the-art multi-view stereo methods: COLMAP [29], [30], PMVS2 [57], OpenMVS [58], and DeepMVS [34].
Figure 13. Comparison of scene reconstruction results of different methods on NYUv2 dataset (top two rows) and our dataset (bottom two rows). From left to right are the results of COLMAP [29], [30], PMVS2 [57], OpenMVS [58], DeepMVS [34] and our method.

Figure 14. Comparison of multi-view reconstruction results of different methods on NYUv2 dataset [5]. From left to right are the results of COLMAP [29], [30], PMVS2 [57], OpenMVS [58], DeepMVS [34] and our method.
We obtain the sparse views for NYUv2 dataset by selecting one frame per 30-40 frames, and use the camera parameters estimated by COLMAP [29] for OpenMVS [58], PMVS2 [57] and DeepMVS [34]. As shown in Figure 13, COLMAP [29], [30] fails to generate meaningful results on NYUv2 dataset from sparse views. Incorrect 3D points in results reconstructed by PMVS2 [57] and OpenMVS [58] are observed: some points gather together from side view and top view on NYUv2 dataset. Moreover, their obtained point clouds are too sparse to provide acceptable results even enhanced by linear interpolation. DeepMVS reconstructs more points compared with the traditional methods, but the reconstructed model contains a lot of noise and outliers. On the contrary, our method achieves the best results for sparse multi-view reconstruction by considering 7D information (geometry, photometry and semantics) and by using joint global and local registration. More results on NYUv2 dataset [5] and our dataset using three or four sparse views are given in Figure 14 and Figure 15, respectively. We observe that the multi-view stereo method in COLMAP [30] fails to generate 3D point clouds, and the point clouds reconstructed by OpenMVS [58] and PMVS2 [57] lack sufficient density and completeness. Although DeepMVS [34] achieves dense reconstruction, the reconstructed model contains many incorrect points. In contrast, our method achieves accurate and complete reconstruction from sparse views. Because COLMAP [30] fails for most scenes in NYUv2 dataset [5], we give quantitative evaluation on our dataset in Table IX. Two metrics are used to evaluate the results of MVS reconstruction: accuracy and completeness. Accuracy represents the average distance between the points on reconstructed model and the nearest points on the ground-truth model. Completeness measures the percentage of the points on the ground-truth model that can find corresponding points on the reconstructed model within a certain distance threshold (0.1). We generate the 3D ground-truth model by fusing multi-view ground-truth depth point clouds using ICP. As shown in Table IX, our method achieves the most complete 3D reconstruction, significantly outperforming other competing methods. Although traditional multi-view stereo methods [30], [57], [58] have higher accuracy, their reconstructed points are too sparse to provide acceptable results even enhanced by linear interpolation. Figure 16 shows our reconstructed models on NYUv2 dataset [5] and our dataset presented from five different views.

V. CONCLUSIONS

In this paper, we solve a challenging problem: reconstructing and understanding indoor 3D scenes based on several
Figure 15. Comparison of scene reconstruction results of different methods on our dataset. From left to right are the results of COLMAP [29], [30], PMVS2 [57], OpenMVS [58], DeepMVS [34] and our method.

Figure 16. Our reconstructed models on NYUv2 dataset [5] and our dataset presented from five different views as illustrated for each scene.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (lower is better)</th>
<th>Completeness (higher is better)</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLMAP [30]</td>
<td>3.74</td>
<td>2.33%</td>
</tr>
<tr>
<td>PMVS2 [57]</td>
<td>3.71</td>
<td>1.83%</td>
</tr>
<tr>
<td>OpenMVS [58]</td>
<td>3.68</td>
<td>1.25%</td>
</tr>
<tr>
<td>DeepMVS [34]</td>
<td>21.49</td>
<td>12.47%</td>
</tr>
<tr>
<td>Ours</td>
<td>17.72</td>
<td>31.55%</td>
</tr>
</tbody>
</table>

Iterative network, IterNet, is proposed to jointly estimate depth map and semantic segmentation from a single color image. We proposed a joint global and local registration method to reconstruct indoor 3D scenes from sparse views. We also build and make available the IterNet RGB-D dataset, a new dataset that simultaneously provides high-resolution photorealistic RGB images, accurate depth maps, and pixel-level semantic labels for thousand of layouts. Experimental results on both public datasets and our dataset demonstrate that our method achieves the best results on depth estimation, semantic segmentation and multi-view reconstruction, compared with state-of-the-art methods.


Jingyu Yang (M’10–SM’17) received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2003, and Ph.D. (Hons.) degree from Tsinghua University, Beijing, in 2009. He has been a Faculty Member with Tianjin University, China, since 2009, where he is currently a Professor with the School of Electrical and Information Engineering. He was with Microsoft Research Asia (MSRA) in 2011, and the Signal Processing Laboratory, EPFL, Lausanne, Switzerland, in 2012, and from 2014 to 2015. His research interests include image/video processing, 3D imaging, and computer vision.

Ji Xu received the B.E. degree from the School of Information Engineering, Yangzhou University, Yangzhou, China, in 2017. He is currently pursuing the M.E. degree at the College of Electrical and Information Engineering, Tianjin University, Tianjin, China. His research interests are mainly in 3D reconstruction and computer vision.

Kun Li received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2006, and the master and Ph.D. degrees from Tsinghua University, Beijing, in 2011. She visited École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2012 and from 2014 to 2015. She is currently an Associate Professor with the School of Computer Science and Technology, Tianjin University, Tianjin, China. Her research interests include dynamic scene 3D reconstruction and image/video processing.

Yu-Kun Lai received his bachelor and Ph.D. degrees in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Reader of Visual Computing in the School of Computer Science & Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing and computer vision. He is on the editorial boards of Computer Graphics Forum and The Visual Computer.

Huanjing Yue (M’17) received the B.S. and Ph.D. degrees from Tianjin University, Tianjin, China, in 2010 and 2015, respectively. She was an Intern with Microsoft Research Asia from 2011 to 2012, and from 2013 to 2015. She visited the Video Processing Laboratory, University of California at San Diego, from 2016 to 2017. She is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University. Her current research interests include image processing and computer vision.

Jianzhi Lu received the B.E degree from the College of Informatics, South China Agricultural University in 2012. He is currently serving for the Cloud rendering lab of 3VJ in Guangzhou, China. His research interests include distributed computing, system integration and rendering.

Hao Wu received the B.S. degree in electrical engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 2011. He is currently working toward the Ph.D. degree in electrical engineering in Tianjin University, Tianjin, China. His research interests include image/video compression and computer vision.

Yebin Liu received the BE degree from Beijing University of Posts and Telecommunications, China, in 2002, and the PhD degree from the Department of Automation, Tsinghua University, Beijing, China, in 2009. He has been working as a research fellow at the computer graphics group of the Max Planck Institute for Informatik, Germany, in 2010. He is currently an associate professor in the Department of Automation, Tsinghua University. His research areas include computer vision, computer graphics and computational photography.