IEICE
TRANSACTIONS

on Information and Systems

VOL. E106-D NO. 3
MARCH 2023

A PUBLICATION OF THE INFORMATION AND SYSTEMS SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Learning Multi-Level Features for Improved 3D Reconstruction

Fairuz SAFWAN MAHAD[†a)], ***Nonmember***, **Masakazu IWAMURA**[†*b)], ***Senior Member***,
and **Koichi KISE**[†*c)], ***Fellow***

SUMMARY 3D reconstruction methods using neural networks are popular and have been studied extensively. However, the resulting models typically lack detail, reducing the quality of the 3D reconstruction. This is because the network is not designed to capture the fine details of the object. Therefore, in this paper, we propose two networks designed to capture both the coarse and fine details of the object to improve the reconstruction of the detailed parts of the object. To accomplish this, we design two networks. The first network uses a multi-scale architecture with skip connections to associate and merge features from other levels. For the second network, we design a multi-branch deep generative network that separately learns the local features, generic features, and the intermediate features through three different tailored components. In both network architectures, the principle entails allowing the network to learn features at different levels that can reconstruct the fine parts and the overall shape of the reconstructed 3D model. We show that both of our methods outperformed state-of-the-art approaches.

key words: *computer vision, 3D reconstruction, deep learning, multi-view*

1. Introduction

Three-dimensional (3D) reconstruction using RGB images is a widely researched computer vision topic. 3D reconstruction has a wide range of applications in various fields including medical imaging [1], archaeology [2], and civil engineering [3]–[5]. Current research typically addresses either multi-view 3D reconstruction or single image 3D reconstruction. One of the main differences between these two topics is the number of viewpoints required. Conventionally, 3D reconstruction was performed with multiple images taken from different viewpoints. With the surge of popularity of neural networks, single image 3D reconstruction [6]–[14] was introduced and quickly became a trend. However, this comes with a limitation: single image 3D reconstruction is an ill-posed problem. The convenience of obtaining just one viewpoint for single image 3D reconstruction is tempered by the reduced quality. This occurs because a 2D image provides us with minimal information about a particular object or scene, thereby forcing the network to compensate for the unobserved viewpoints by inferring shape from the learned data. Thus, single image 3D reconstruction is appropriate in situations that do not require high-quality results.

Under certain circumstances, multi-view 3D reconstruction produces better results than single image 3D reconstruction. There is a trade-off between the number of viewpoints and the quality of the reconstructed 3D model: more viewpoints attain a more accurate the reconstructed 3D model. Research addressing multi-view 3D reconstruction [15]–[18] is motivated by this tradeoff. Using multiple 2D images theoretically yields better results than using a single 2D image to reconstruct a 3D model. However, reconstructing a highly accurate 3D model remains a challenging task even with state-of-the-art multi-view 3D reconstruction methods. Soltani et al. proposed a method to reconstruct a high resolution 3D model using multiple depth maps or silhouettes [18], which is shown to outperform other 3D reconstruction methods in terms of multi-view point cloud-based methods [19]. However, similar to most state-of-the-art single-image 3D reconstruction methods, the method proposed in [18] ignores an important aspect: the reconstruction of the detailed parts of the 3D models. In most cases, the method failed to reconstruct the fine parts of the 3D model, as shown in Fig. 1. The fine parts are either not reconstructed at all or only slightly reconstructed, where the latter indicates incomplete or sparse reconstruction. Specifically, the method failed in reconstructing details such as the legs of a chair and the handle and tip of a rifle. Our research aims to improve reconstruction quality, with a particular focus on reconstructing the detailed parts of the 3D models. This issue has typically been overlooked in most single-image and multi-view 3D reconstruction methods.

In this paper, we propose a simple but effective approach to improve the reconstruction of the detailed parts of 3D models using multiple viewpoints. To achieve this, we propose two methods. Both of our proposed methods are based on the state-of-the-art [18]. One is a multi-scale layered network involving a sequence of downsampling and upsampling[**]. The network architecture is based on the pyramidal hierarchical-based network concept from [22], originally designed for object detection. [22] builds a multi-scale feature map where each feature map consists of high-level semantic features with different spatial resolutions. Leveraging the semantically strong features extracted

Manuscript received September 4, 2022.
Manuscript revised October 28, 2022.
Manuscript publicized December 8, 2022.

[†]The authors are with Dept. of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University, Sakai-shi, 599–8531 Japan.

[*]Presently, with Dept. of Core Informatics, Graduate School of Informatics, Osaka Metropolitan University.

a) E-mail: fsafwan88@gmail.com
b) E-mail: masa.i@omu.ac.jp
c) E-mail: kise@omu.ac.jp

DOI: 10.1587/transinf.2020ZDP7508

[**]This approach is based on the method in [20], [21].

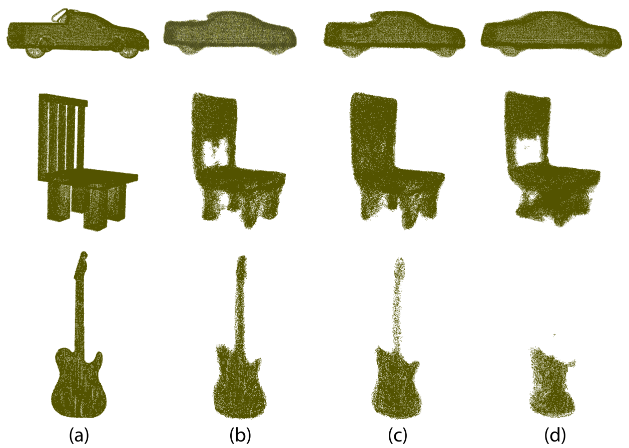

Fig. 1 A comparison of the reconstructed 3D models achieved by our proposed method 1, our proposed method 2, and state-of-the-art method from Soltani et al. [18]. (a) Ground truth. (b) Proposed method 1. (c) Proposed method 2. (d) Soltani et al. [18]. Our methods improve the reconstruction of the detailed parts by using both local and generic features.

at different levels further improves the reconstruction quality of the detailed parts of the 3D model. We demonstrate that the mechanism of the pyramidal hierarchical-based network is effective in extracting useful features at different scales to enhance the quality of the reconstructed 3D model. Our second network is a multi-branch deep generative network with specifically designed components. We adopt the network architecture concept from [23], which introduces the use of multi-level features in their colorization network. Inspired by [23], we include three components in the network: global net, mid net, and low net. These different components obtain features at different levels. The global net captures generic features across the image, while the low net captures local features. The mid net captures intermediate features between the generic and local features. Concatenating all of these features allows the network to learn the various useful features obtained from different levels. This helps improve the reconstruction of the detailed parts while preserving the overall quality and shape, leading to increased overall accuracy.

Our contributions are listed below:

- We propose two different networks to improve the 3D reconstruction quality.
- We introduce a modified version of a pyramidal hierarchical-based network to the encoder.
- We introduce multiple network components with a multi-branch VAE structure to learn features at different levels.
- We improve the 3D reconstruction of details, especially the thin/fine parts of the 3D model.

2. Related Work

We categorize 3D reconstruction methods into two types: single image 3D reconstruction [6]–[14] and multi-view 3D reconstruction [15]–[18].

Reconstructing 3D models with a single image has become increasingly popular, and entails particular advantages and disadvantages. It requires only a single image to reconstruct a 3D model but results in reduced accuracy. This approach is trained with large-scale repositories of 3D CAD models such as ShapeNet [24]. Using only a single viewpoint forces the network to infer the unobserved viewpoints by using learned information to estimate a complete but inaccurate representation of the query image. [6]–[8], [11] addressed some single-image 3D reconstruction problems, but in most cases the overall shape of the reconstructed 3D models does not resemble that of the intended query images. In addition, this approach neglects the detailed parts of the model, specifically the fine parts (e.g., the legs of a chair or table, the spout and lever of a faucet) and thus precludes successful reconstruction. Moreover, the reconstructed 3D models have low resolution. [10], [25], [26] reconstruct high-resolution 3D models in mesh representation and are able to reconstruct the fine parts of the models. However, the major drawback of this approach is that the quality and accuracy of the final 3D reconstructed model relies heavily on the initial model selection. Their methods involve selecting an initial 3D model before further reconstructing the initially selected model to obtain a final 3D model. However, the method fails at reconstructing complex 3D models or 3D models that do not share a similar shape to those within the initial selection.

Multi-view 3D reconstruction methods use two to an arbitrary number of viewpoints. More viewpoints offer more information, resulting in a more accurate and complete 3D reconstructed model. Choy et al. [15] and Kar et al. [16] produced 3D reconstructed models in the form of voxels but with low resolution. Although voxel dimensions can be increased, this also increases memory consumption. The immense resource overhead makes it impractical to reconstruct the detailed parts. Ji et al. [17] proposed an end-to-end learning framework based on a multi-view stereo method as in [27]. It has similar limitations to [15], [16] because it reconstructs 3D models in the form of voxels. Moreover, because it is based on the conventional 3D reconstruction method, it also inherits the original limitations, resulting in an incomplete 3D reconstructed model. In contrast, Soltani et al. [18] proposed a multi-view reconstruction method focusing on synthesizing 3D shapes. However, in most cases, the method failed to reconstruct the fine parts of the 3D model. According to a recent survey paper [19], [18] outperformed other 3D reconstruction methods in terms of multi-view point clouds-based methods. For this reason, we specifically selected [18] for implementation purposes to show the effectiveness of our proposed method.

3. Approach

3.1 Soltani's Method [18]

Soltani et al. proposed a network with three different set-

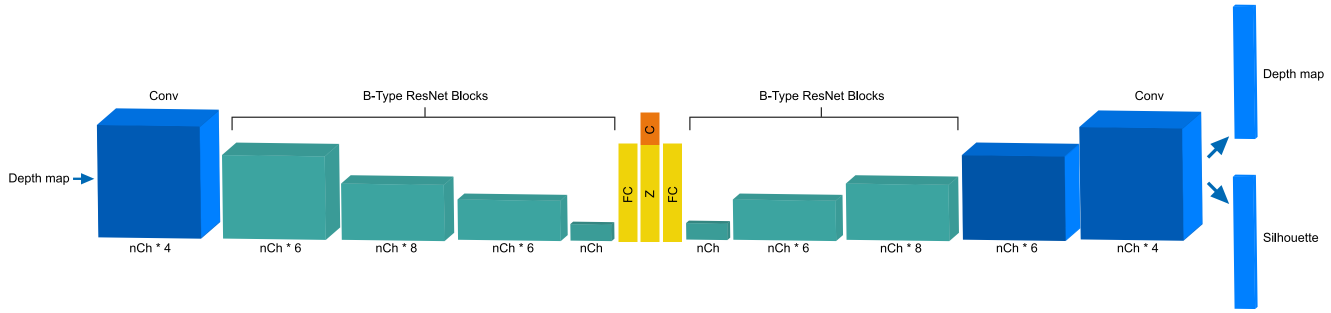

Fig. 2 The pipeline structure of the method proposed by Soltani et al. [18] nCh denotes the number of channels, which is 74. FC denotes a fully-connected layer while Z is the latent variable. C denotes "conditional", but is not used in this paper.

tings. First, the encoder accepts either 20 depth maps or 20 silhouette images. Second, it can also accept 20 depth maps or 20 silhouette images but randomly nullifies 15 to 18 images for each input model. Last, it can also work with either a single depth map or silhouette image. The network accepts either depth maps or silhouette images as input and produces a set of 20 depth map images and 20 silhouette images. The output from the network is a total of 40 images that are used to render a final 3D model. The 20 depth map images are projected back into 3D space, creating an initial 3D model. The 3D model is further refined by using the silhouette images to filter outliers. According to their paper, despite the easiest setting (i.e., the first), the detailed parts in the reconstructed 3D models were unsuccessfully recovered. Therefore, in this paper, we focus only on using the first setting which is using 20 depth map images as the input.

Figure 2 provides an overview of the pipeline proposed by Soltani et al. The core of its network structure is a deep generative network, using the variational autoencoder (VAE) [28] with B-type ResNet blocks [29] for both its encoder and decoder, producing a high resolution 3D model. However, its network is designed to only learn features at a single level. Thus, it fails to reconstruct thin and fine parts of the 3D models, resulting in an incomplete and inaccurate 3D model.

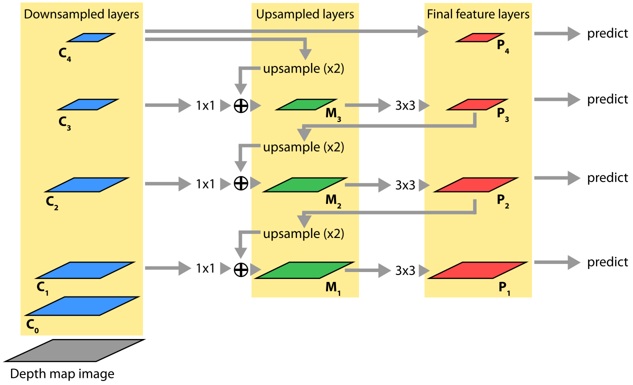

Fig. 3 Proposed network architecture of Lin et al. [22]

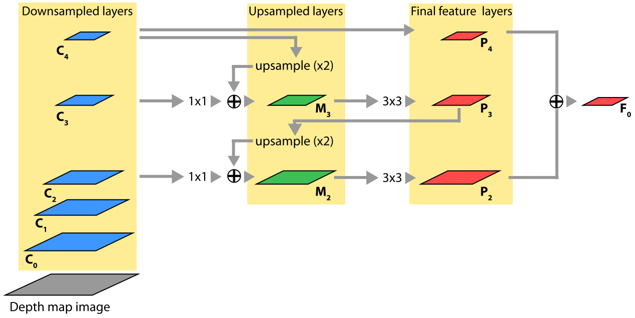

Fig. 4 Network architecture of our proposed method 1.

3.2 Proposed Method 1

Our first proposed method improves on Soltani et al. [18]. Although the latter method reconstructs a high-resolution 3D model, the network is designed for completion and generalization and only learns features at a single scale. This results in a failure to reconstruct most of the detailed parts of the 3D model. To enhance the quality of the reconstructed 3D model, our strategy focuses on capturing features related to small and thin parts of the object. This improves the overall quality of the reconstructed 3D model. We upgrade the encoder part of the VAE network structure of [18] by implementing the multi-scale layered network with skip connections, inspired by the method in [22], as shown in Fig. 3. According to [22], the multi-scale upsampled layers are made up of semantically stronger features than the downsampled layers, which is the main advantage of using pyramidal-based networks. Figure 3 shows that [22] uses every output level $\{P_1, P_2, P_3, P_4\}$ to make predictions independently. Our proposed network architecture, shown in Fig. 4, adapts a similar concept but with the following two distinct differences.

1. [22] considers every output level $\{P_1, P_2, P_3, P_4\}$ independently, while our proposed network concatenates the final feature maps $\{P_2, P_4\}$. This produces a final merged feature map followed by a fully-connected layer.
2. Our proposed network is implemented in the encoder of the VAE structure.

Table 1 Ablation study results for different final layers for proposed method 1.

Combination of Layers	Accuracy (%)
$\{P_2\}$ and $\{P_4\}$ (Proposed method 1)	81.5 ±0.0008
$\{P_2\}$ and $\{P_3\}$	81.3 ±0.001
$\{P_3\}$ and $\{P_4\}$	81.3 ±0.0006
$\{C_2\}$ and $\{C_4\}$	81.1 ±0.0005
$\{C_2\}$ and $\{C_3\}$	81.2 ±0.001
$\{C_3\}$ and $\{C_4\}$	81.2 ±0.0005

3.2.1 Network Architecture

Figure 4 shows the network architecture of our first proposed method, which we adapted from [22]. It features bottom-up and top-down pathways with skip connections. All of the blocks in both the bottom-up and top-down pathways are consist of ResNet blocks [29]. The network starts with a bottom-up pathway by scaling down the input image to feature maps of sizes $\{110^2, 53^2, 25^2, 11^2, 4^2\}$, denoted as $\{C_0, C_1, C_2, C_3, C_4\}$, respectively. The bottom-up pathway ends with layer C_4.

The top-down pathway begins by first upsampling the last layer (C_4) back to a feature map of size 11^2. The layer C_3 (which has the same spatial size as the upsampled layer from C_4) from the bottom-up pathway is associated with the upsampled layer using a 1x1 convolution. Similar to [22], the features in the upsampled layer and the skip connection layer are then concatenated, producing a feature map denoted as M_3. We apply a 3x3 convolution to layer M_3 as an anti-aliasing measure to reduce the aliasing effect caused by sampling the layers. This produces a final feature map for this level, denoted as $\{P_3\}$. We iterate until we obtain the final features maps, denoted as $\{P_2, P_3, P_4\}$. Note that $\{P_4\}$ is the same layer as $\{C_4\}$. Last, we concatenate the final feature maps $\{P_2, P_4\}$ to produce a final merged feature map, denoted as $\{F_0\}$, followed by a fully connected layer. To concatenate the final feature maps $\{P_2\}$ and $\{P_4\}$, we further downsample layer $\{P_2\}$ to match the size of layer $\{P_4\}$. To reduce the number of parameters, we disregard $\{P_3\}$ in the final merged feature map. For the same reason, we do not further upsample layer $\{P_2\}$ to a spatial size of $\{110^2, 53^2\}$. We implement our proposed network architecture only in the encoder of the VAE structure of [18]. We used the same decoder structure as [18].

The network in [18] can be trained either in an unsupervised manner or conditionally. For our purpose, we train our network using the unsupervised method. Therefore, we use the same loss function as Eq. (1) in [18].

An ablation study was performed to underpin our selection of final layers, which is $\{P_2\}$ and $\{P_4\}$. Table 1 shows the accuracy (average and standard deviation) for different combinations of layers: $\{P_2\}$ and $\{P_3\}$, and $\{P_3\}$ and $\{P_4\}$. These experiments were run five times. In addition, we also examined layers in the bottom-up pathway. That is, $\{C_2\}$ and $\{C_4\}$, $\{C_2\}$ and $\{C_3\}$, and $\{C_3\}$ and $\{C_4\}$. These experiments were run three times. The result shows that $\{P_2\}$ and $\{P_4\}$ achieved the highest accuracy as compared to the other combinations of layers. Considering the standard deviation, its difference is considered sufficient. Due to this, we selected $\{P_2\}$ and $\{P_4\}$ as our final layers.

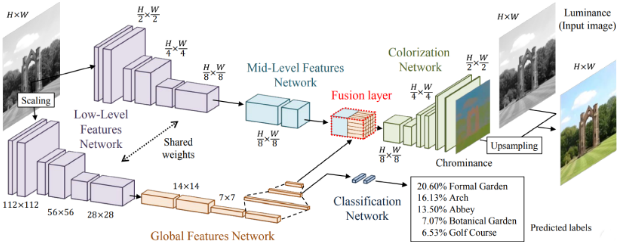

Fig. 5 Proposed network architecture of Iizuka et al. [23]

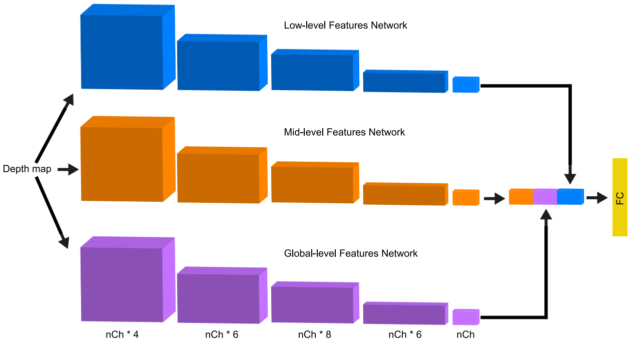
Low-level Features Network
Depth map
Mid-level Features Network
Global-level Features Network
FC
nCh * 4
nCh * 6
nCh * 8
nCh * 6
nCh

Fig. 6 Encoder structure of our proposed method 2. nCh denotes the number of channels, which is 74. FC denotes a fully-connected layer.

3.3 Proposed Method 2

Our second proposed method also improves on Soltani et al. [18]. As with our first proposed method, we aim to improve the 3D reconstruction accuracy by focusing on the reconstruction of the fine parts of the 3D model. This time, we improve the VAE network structure used in [18] by adapting the concept of the network structure from [23], as shown in Fig. 5. Similar to [23], our proposed network architecture, as shown in Fig. 6, comprises three components: the global net, the mid net, and the low net. However, our proposed network architecture differs from [23] in two distinct ways:

1. [23] implements a shared weight for the low net, where the weights are passed onto the mid net and global net in a separate branch. In contrast, we separate all three components into different branches, as illustrated in Fig. 6. The features of the global net, the mid net, and the low net are then concatenated before further processing with a fully-connected layer.
2. In contrast to [23], which exclusively uses 3x3 convolutional kernels across all three components, the components in our network architecture do not share the same kernel size. Concatenating all of the features from the three components allows the network to learn features at different levels. This enables the network to capture both coarse and fine details from each image.

Each component of the proposed method plays a vital role in extracting features at different levels: ranging from local to generic features. The following paragraphs describe the implementation details of each of the three components of the proposed network architecture: the low, global, and mid nets.

(1) Low net

The low net extracts local features directly from the input. To extract local features, the low net uses a filter size of 2x2 pixels. The 2x2 filter allows the low net to learn local features by capturing smaller details. We selected a 2x2 filter instead of a 1x1 filter because the latter treats a single pixel as a feature, thereby ignoring any information about neighboring pixels.

(2) Global net

The purpose of the global net is to extract generic features using a filter size of 5x5 pixels. Our preliminary experiment shows that a filter larger than 5x5 degrades the accuracy. In our case, a filter larger than 5x5 is unsuitable because it extracts features that are too generic.

(3) Mid net

The purpose of the mid net is to obtain intermediate features between the generic features and the local features with a filter size of 3x3 pixels. Features from the mid net are concatenated with features from the global net and the low net.

We use this proposed architecture as the encoder and reuse the same decoder as [18].

3.3.1 Roles of Features at Different Scale Levels

In Fig. 7, we illustrate how the local, intermediate, and generic features contribute to reconstructing the different parts of the image. The figure shows the estimated depth map side-by-side with a heatmap generated by obtaining the difference between the ground truth and the estimated depth map. Seven dedicated networks were trained separately resulting in seven different results (b) to (h). The first three models are those trained individually, which are the low net (b), mid net (c), and the global net (d). As described in Sect. 3.3, these differences are in the size of the receptive fields. Models (e), (f), (g), and (h) are variations acquired by training several combinations of the three branch models (b), (c), and (d). For example, model (e) is trained using a combination of (b) and (c) and so on. Among them, Model (h) is trained using a combination of (b), (c), and (d), which is our proposed method 2. The extracted features from different scale levels affect the estimated depth map differently. The heatmap in (b), representing the low net, shows that the depth map is estimated on a lower level, only capturing a rough shape of the gun. While the heatmap in (c), representing the mid net, estimates the stock (the rear part of the gun) and also the magazine of the gun. This indicates that the mid net learns the shape of the gun better than the low net because the mid net operates on a higher level. The global net (d), estimates the stock and also the muzzle (the tip) of the gun. This demonstrates that by learning the generic features, the global net better learns the overall shape of the object than the mid net and the low net. The low net (b) trained individually on its own does not show its effectiveness. However, its effectiveness shines when paired with other branches (c) and/or (d). Results from (e), (f), and (g) show that combining at least two of the main models (b), (c), and (d) yields a better result as compared to the main models trained individually. Result in (h) shows that combining all main models (b), (c), and (d) yields an even better result as compared to the others. For those who have an interest in the quantitative comparison between the models (b), (c), and (d), we report they achieved an accuracy of 80.8%, 80.8%, and 80.9% in one run, respectively, following the manner of Sect. 4.3.

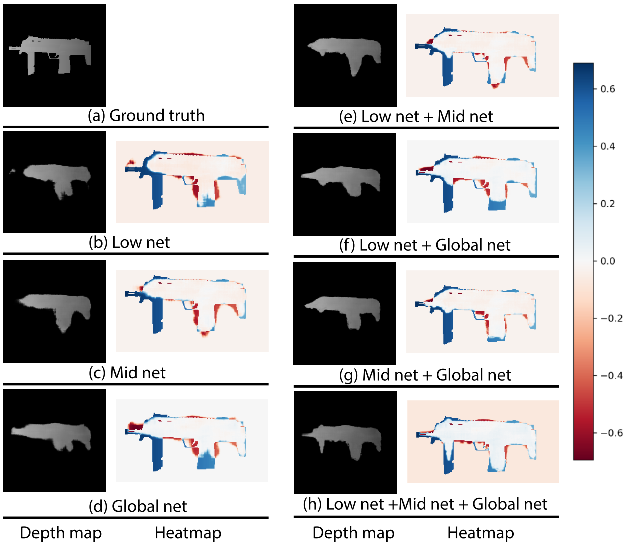

Fig. 7 A heatmap visualization of the depth map accuracies obtained using features at different scale levels. (a) represents the ground truth. (b)–(h) represent the results of seven dedicated networks trained separately; they represent all combinations of with/without the low, mid, and global nets. Models (b), (c), and (d) solely use the low, mid, and global nets, respectively. Models (e), (f), and (g) use two of them. Model (h) uses all three networks, which is our proposed method 2. The heatmap shows the difference between the ground truth and the estimated depth map using features from different scale levels. The white region shows the parts of the depth map that were successfully estimated. The blue region shows the parts where the estimation failed. The red region shows the parts outside of the bounds as compared with the ground truth.

3.3.2 Efficient Implementation

Using a multi-leveled network with three components corresponding various receptive fields increases the number of parameters. In addition, the B-type residual blocks also significantly increase the number of parameters. Therefore, to suppress the effect of the increase in the number of parameters on the computational cost, we implement a filter decomposition. That is, instead of a 5x5 filter for the global net, we use two 3x3 filters. Two 3x3 filters yield a similar result to one 5x5 filter but are more computationally efficient. Two 3x3 filters have fewer weights but more layers, leading to more complex non-linear features.

To further reduce the computational cost, we implement a separable filter as proposed in [30]. Instead of a 3x3 filter for the mid net and global net, we further decompose it into 3x1 and 1x3 filters. Thus, the global net has two layer sets comprising 3x1 and 1x3 filters. Similarly, instead of a 2x2 filter for the low net, we use 2x1 and 1x2 filters. This further reduces the computational cost from $O(d^2H^{'}W^{'})$ to $O(2dH^{'}W^{'})$, where H and W refers to the height and weight of the output feature map, respectively [30].

4. Experiments

In this section, we evaluate our methods against the original state-of-the-art method proposed by Soltani et al. [18]. We present both qualitative and quantitative evaluations of our methods.

4.1 Experimental Settings

We trained our model on the ShapeNet dataset [24]. The ShapeNet dataset consists of 57 object categories spanning a total of 56,652 3D models. To use the dataset for training, we rendered depth maps from all of the 3D models in the dataset with fixed camera angles. The rendered size was 224 × 224 pixels. To evaluate our method against [18], we used the same dataset distribution: 92.5% for training and the remaining 7.5% for testing. We used the pre-trained dataset, which was the exact model used to produce the result in [18], along with their original source code. We trained our model on a system using NVIDIA GeForce GTX TITAN V.

4.2 Qualitative Evaluation

We demonstrate our results against Soltani et al. [18] in Fig. 8. The 3D models were reconstructed for the test dataset — the 7.5% that was not used for training. Figure 8 illustrates two main points. First, both of our methods better capture the overall shape of the object compared with the method of Soltani et al. This is attributed to the fusion of features across multi-scale layers in our proposed method 1 and the merging of the local, intermediate, and generic features in our proposed method 2. Second, our methods better preserve the thin parts of the object. In most cases, the method of Soltani et al. failed to reconstruct the thin parts of objects.

We designed the network architecture of our proposed method 1 to use features across multiple scale levels. The features in upsampled layers are semantically stronger than those in downsampled layers. Thus, upsampled layers provide meaningful information for both global and local features. However, the features in upsampled layers are weak in terms of localization. In contrast, the features in downsampled layers are semantically weaker than those in upsampled layers, but they have better localization. By associating the semantically stronger features with weak localization (from the upsampled layers) with the semantically weaker features with strong localization (from the downsampled layers) at each level, the network simultaneously learns both the overall shape and detailed parts of an object. These factors lead to a much more complete reconstructed 3D model compared with that obtained by the method of Soltani et al.

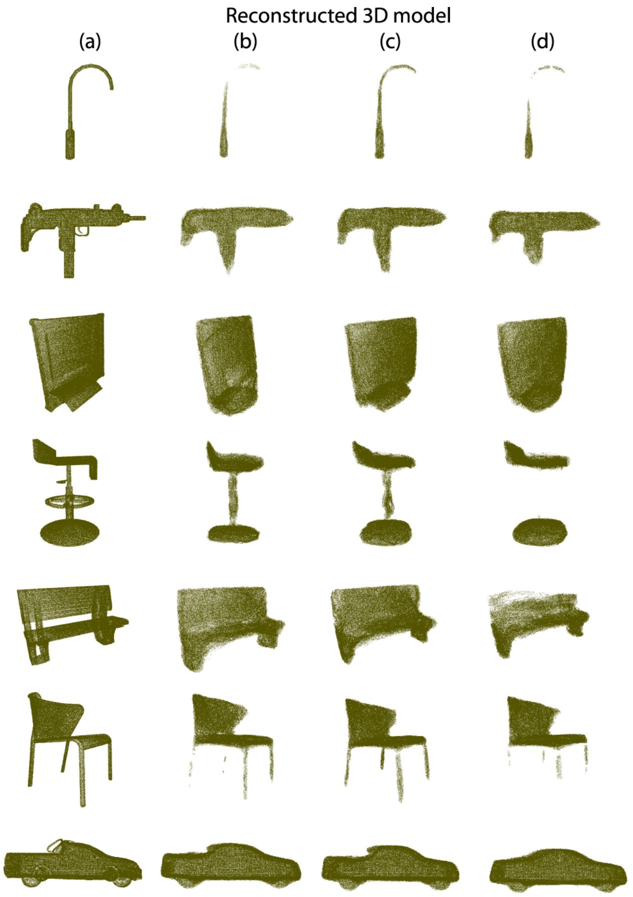

Fig. 8 Qualitative results for successful reconstructions. Comparison of 3D reconstructed models of our proposed method 1, proposed method 2, and Soltani et al. [18]. (a) 3D model ground truth. (b) 3D model reconstructed by proposed method 1. (c) 3D model reconstructed by proposed method 2. (d) 3D model reconstructed by Soltani et al. [18].

We achieve a similar effect from the network architecture of our proposed method 2 by extracting the local, intermediate, and generic features using three dedicated branches. Merging all of these branches at the end of the network fuses all of the feature maps, allowing the network to learn both the overall shape and the detailed parts of an object.

Because of these two main points, the 3D models reconstructed by our methods are much more complete and closer to the ground truth than those reconstructed by Soltani et al. However, our method failed to reconstruct complex shapes, as shown in Fig. 9. This may have been caused by a lack of training data. For example, the topmost lamp in the figure has a specific shape that was not present in the training data.

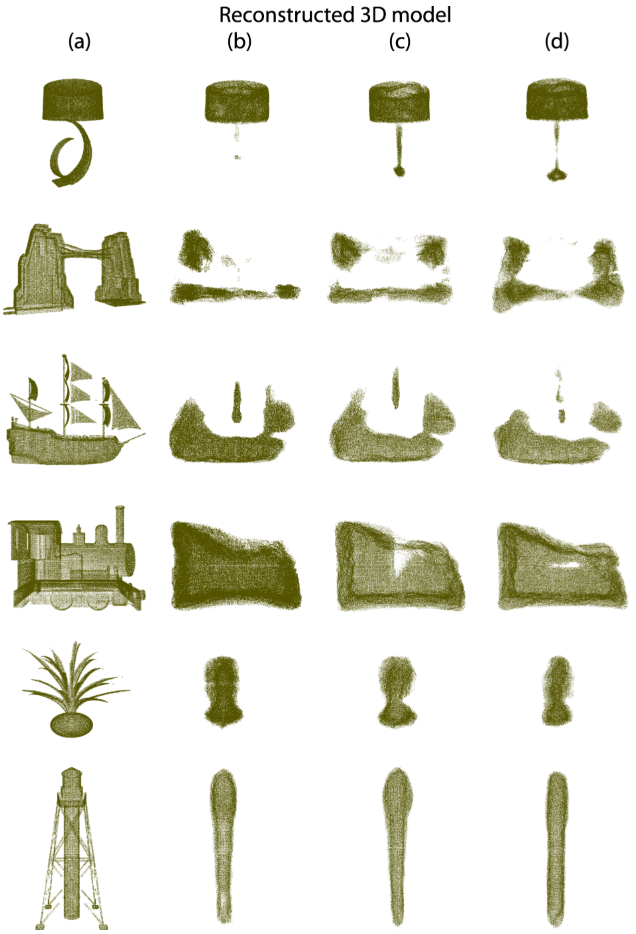

Fig. 9 Qualitative results for failed reconstructions. Comparison of 3D reconstructed models of our proposed method 1, proposed method 2, and Soltani et al. [18]. (a) 3D model ground truth. (b) 3D model reconstructed by proposed method 1. (c) 3D model reconstructed by proposed method 2. (d) 3D model reconstructed by Soltani et al. [18].

4.3 Quantitative Evaluation

In this section, we discuss our results quantitatively. We evaluate our results by using the mean intersection-over-union (IoU) along with the breakdown of each category. We reconstructed all of the 3D models in the test dataset in point cloud form and computed the IoU by converting the point clouds into 3D voxels. We trained the neural network models on all 57 object classes five times. Table 2 lists our results for each class and compares them with those of the method of Soltani et al. [18]. We also present the standard deviation for each category over the five runs to indicate that the results obtained were stable and not coincidental. Note that the average IoU was reported as 84.0% in [18]. However, after training and running it for five times using the original source code provided by the authors, we achieved an average IoU of 81.1%. Therefore, we used this value as our benchmark for the method of Soltani et al. [18].

Table 2 shows the best results for each method. The table shows that our proposed methods 1 and 2 outperformed the method of Soltani et al. in 45 and 43 out of 57 categories, respectively. Our methods performed better in categories such as guitar, table, pistol, microphone, and chair because most of the 3D models in those categories contained thin structures, lending a clear advantage to our methods. In particular, in categories such as lamp, bookshelf, camera, and vase, our method coped better with the shape complexity than the method of Soltani et al. [18].

Additionally, to measure the significance of the evaluations of our proposed methods against that of the method of Soltani et al. [18], we ran a statistical test: the student's T-test to produce a P-value; a P-value of less than 0.05 indicates that the experiment is statistically significant. Our experiment obtained a P-value of 0.008 for proposed method 1 and a P-value of 0.023 for proposed method 2, proving that our evaluation of our two proposed methods is statistically significant.

4.4 Discussion

Both proposed methods utilize upsampling and downsampling in order to extract and learn features at multi-scale layers. However, a distinct difference between the two proposed methods is that proposed method 1 utilizes skip connections to associate features extracted from the bottom-up layers to the top-down layers in conjunction with several merging of layers. On the other hand, proposed method 2 features a much simpler architecture where features are extracted by going through a series of downsampling (encoder) and upsampling (decoder) without any skip connection. The merging of layers from all three branches are only done once right before the fully-connected layer. The quantitative evaluation in Table 2 revealed their similarity and dissimilarity. The similarity appeared in the fact that their gains had a high correlation of 0.90. This indicates that the categories in which they excel and those in which they do not excel are similar. However, the variances of the gains show the dissimilarity: it was smaller (i.e., 1.47) in proposed method 1 and larger (1.64) in proposed method 2. This indicates that proposed method 1 is more stable than proposed method 2. On the other hand, in some cases, proposed method 2 can obtain a much better result than proposed method 1, while in other cases proposed method 2 can obtain a much worse result than proposed method 1. For example, the faucet, rifle, monitor, and chair in Fig. 8 show that proposed method 2 reconstructed better than proposed method 1.

Generally, in deep learning, features closer to the end of the network are semantically stronger, while features closer to the image source have high localization accuracy. In proposed method 1, the skip connections associate the semantically stronger features (features from upsampling) with the highly accurate localized features (features from downsampling). This explains why the 3D reconstructed models are much more stable and consistent, especially in dealing with thin and tight spots thus making it better suited for objects that have thin and tight spots (such as bicycle and

Table 2 Category-wise comparison between Soltani et al. [18] and our proposed methods 1 and 2 (PR1 and PR2). "#M" refers to the number of models. The columns labeled "Accuracy" for [18], PR1, and PR2 represent the accuracy in terms of IoU followed by its standard deviation value. The columns labeled "Gain" for PR1 and PR2 represent the difference between the IoU of PR1 and [18] and that between PR2 and [18], respectively.

Category	#M	Soltani [18]	Proposed methods 1		Proposed methods 2	
		Accuracy (%)	Accuracy (%)	Gain	Accuracy (%)	Gain
aeroplane	304	71.8 ±0.002	**72.7** ±0.001	0.9	**72.7** ±0.001	0.9
bag	8	78.8 ±0.006	**80.6** ±0.008	1.8	**81.9** ±0.013	3.1
basket	11	83.6 ±0.002	**84.5** ±0.003	0.9	**84.1** ±0.001	0.5
bathtub	62	87.3 ±0.006	**88.3** ±0.002	1.0	**88.0** ±0.003	0.7
bed	10	76.3 ±0.007	**77.6** ±0.006	1.3	**77.7** ±0.001	1.4
bench	132	78.1 ±0.007	**78.6** ±0.001	0.5	**80.1** ±0.005	2.0
bicycle	6	43.1 ±0.005	**43.5** ±0.001	0.4	43.0 ±0.002	−0.1
birdhouse	2	82.8 ±0.009	**84.4** ±0.002	1.6	**84.4** ±0.003	1.6
bookshelf	39	73.3 ±0.007	**74.2** ±0.002	0.9	**74.0** ±0.001	0.7
bottle	35	92.5 ±0.006	**93.0** ±0.0008	0.5	**92.8** ±0.0008	0.3
bowl	13	93.5 ±0.005	**93.9** ±0.002	0.4	**93.7** ±0.003	0.2
bus	82	91.1 ±0.005	**91.4** ±0.002	0.3	**91.5** ±0.001	0.4
cabinet	115	89.2 ±0.011	**91.1** ±0.001	1.9	**91.1** ±0.001	1.9
camera	8	67.7 ±0.009	**69.2** ±0.005	1.5	**68.5** ±0.002	0.8
can	8	93.9 ±0.007	**94.8** ±0.004	0.9	**94.2** ±0.003	0.3
cap	3	**81.1** ±0.023	77.6 ±0.003	−3.5	76.9 ±0.003	−4.2
car	554	82.2 ±0.003	**82.7** ±0.000	0.5	**82.6** ±0.001	0.4
cellphone	45	91.8 ±0.005	**92.3** ±0.005	0.5	91.8 ±0.001	0.0
chair	491	77.0 ±0.003	**78.4** ±0.005	1.4	**78.1** ±0.001	1.1
clock	40	79.8 ±0.009	79.6 ±0.006	−0.2	**79.9** ±0.013	0.1
dishwasher	8	93.9 ±0.002	**94.1** ±0.009	0.2	**94.1** ±0.001	0.2
display	81	86.5 ±0.001	**86.6** ±0.001	0.1	**86.6** ±0.001	0.1
faucet	47	**66.1** ±0.007	65.6 ±0.002	−0.5	**66.1** ±0.012	0.0
filecabinet	18	**91.9** ±0.007	91.8 ±0.008	−0.1	90.6 ±0.012	−1.3
flowerpot	42	65.3 ±0.005	**65.7** ±0.001	0.4	**65.6** ±0.001	0.3
guitar	65	78.5 ±0.004	**81.5** ±0.006	3.0	**81.4** ±0.001	2.9
headphone	5	55.7 ±0.010	**60.7** ±0.006	5.0	**59.1** ±0.013	3.4
helmet	16	75.2 ±0.005	**75.6** ±0.0008	0.4	**75.6** ±0.0008	0.4
keyboard	5	87.6 ±0.005	**88.0** ±0.001	0.4	**88.0** ±0.0007	0.4
knife	42	**80.4** ±0.025	78.0 ±0.006	−2.4	77.9 ±0.002	−2.5
lamp	181	68.2 ±0.008	**68.3** ±0.004	0.1	68.0 ±0.001	−0.2
laptop	34	97.0 ±0.003	**97.1** ±0.001	0.1	**97.1** ±0.003	0.1
letterbox	7	**71.2** ±0.008	70.0 ±0.017	−1.2	70.6 ±0.012	−0.6
microphone	6	62.9 ±0.003	**63.6** ±0.001	0.7	**63.1** ±0.002	0.2
microwave	11	**93.7** ±0.002	93.2 ±0.005	−0.5	92.9 ±0.004	−0.8
motorcycle	28	**75.7** ±0.006	75.4 ±0.002	−0.3	74.8 ±0.002	−0.9
mug	17	**84.3** ±0.002	**84.3** ±0.003	0.0	**84.3** ±0.002	0.0
piano	13	79.4 ±0.004	**80.9** ±0.005	1.5	**81.0** ±0.006	1.6
pillow	6	86.7 ±0.001	86.6 ±0.007	−0.1	**87.3** ±0.014	0.6
pistol	19	84.9 ±0.004	**85.4** ±0.001	0.5	**85.2** ±0.002	0.3
printer	18	79.4 ±0.002	**80.9** ±0.009	1.5	**80.8** ±0.009	1.4
remote control	4	89.4 ±0.007	**89.7** ±0.001	0.3	**89.7** ±0.008	0.3
rifle	171	77.5 ±0.005	**77.8** ±0.005	0.3	**77.9** ±0.001	0.4
rocket	7	**73.2** ±0.002	71.3 ±0.006	−1.9	71.2 ±0.008	−2.0
ship	147	79.6 ±0.002	**79.7** ±0.001	0.1	77.5 ±0.012	−2.1
skateboard	18	80.7 ±0.005	**81.4** ±0.001	0.7	**81.5** ±0.001	0.8
sofa	242	87.1 ±0.003	**87.5** ±0.001	0.4	**87.3** ±0.002	0.2
speaker	121	84.2 ±0.002	**84.4** ±0.0008	0.2	**84.4** ±0.001	0.2
stove	8	88.5 ±0.002	**91.1** ±0.004	2.6	**91.4** ±0.002	2.9
table	652	84.8 ±0.002	**85.2** ±0.0009	0.4	**85.2** ±0.001	0.4
telephone	92	92.6 ±0.002	**92.7** ±0.001	0.1	**92.7** ±0.001	0.1
tower	12	**76.6** ±0.004	76.2 ±0.001	−0.4	76.3 ±0.002	−0.3
train	25	84.7 ±0.007	**85.1** ±0.002	0.4	**85.4** ±0.007	0.7
trashcan	28	85.3 ±0.002	**85.4** ±0.0005	0.1	85.3 ±0.001	0.0
vase	38	82.4 ±0.003	**83.2** ±0.004	0.8	**83.1** ±0.003	0.7
vessel	85	**81.5** ±0.01	80.6 ±0.004	−0.9	80.9 ±0.003	−0.6
washing machine	17	92.4 ±0.003	**92.7** ±0.002	0.3	**92.6** ±0.002	0.2
Average		**81.1**	**81.5**	0.4	**81.4**	0.3

motorcycle). On the other hand, proposed method 2 follows a different approach. Proposed method 2 does not utilize any skip connections to associate the features. In fact, the concept is to force the low net to learn the finer details of the object while the mid and global net learns more global features. This explains why proposed method 2 is not as good as proposed method 1 in dealing with thin parts and tight spots. However, in proposed method 2, the features from all three branches at the last layers are combined, which contains meaningful features from every aspect of the object. This explains why the reconstructed 3D models in proposed method 2 tend to be slightly denser than proposed method 1. This also shows that unlike proposed method 1, proposed method 2 is better suited for solid-filled objects with lesser or no gaps in between the object (such as bag, clock, or faucet).

4.5 Limitations

Although our methods largely improved the 3D reconstruction of the overall shape, especially with respect to detailed parts, they did not perform well in categories such as knife, rocket, and cap. These categories contained simple object shapes, with few differences between the samples. Our methods may fail for such categories because the local features may lose their effectiveness when there is little variation in shape, thereby jeopardizing the overall accuracy.

Furthermore, depth map-based 3D reconstruction is known to be unsuitable for capturing thin parts of objects; this claim was supported in [19]. Despite the improvements made in the reconstruction of certain thin and detailed parts, the fundamental limitations of depth map-based 3D reconstruction undermined the effectiveness and capabilities of our methods.

5. Conclusion

In this paper, we proposed two simple yet effective methods to improve the reconstruction of the detailed parts of objects, especially thin parts. To learn the detailed parts in addition to the overall shape of the object, we designed two networks. The first network uses multi-scale layers to learn and merge features of different scales. The second network separately learns highly local features, intermediate features, and generic features through tailored components. We compared our results with those of the state-of-the-art method proposed by Soltani et al. [18] using both qualitative and quantitative evaluations. Our results demonstrate that our methods outperformed the state-of-the-art [18] in most cases, and we show that these results are statistically significant. Our methods achieved improved reconstruction accuracy. Our qualitative results illustrate that the 3D models reconstructed by our methods were more complete and more closely resembled the ground truth than those reconstructed by the state-of-the-art method [18].

References

[1] L. Humbert, J.A. De Guise, B. Aubert, B. Godbout, and W. Skalli, "3D reconstruction of the spine from biplanar x-rays using parametric models based on transversal and longitudinal inferences," Medical engineering & physics, vol.31, no.6, pp.681–687, 2009.

[2] F. Bruno, S. Bruno, G. De Sensi, M.-L. Luchi, S. Mancuso, and M. Muzzupappa, "From 3D reconstruction to virtual reality: A complete methodology for digital archaeological exhibition," Journal of Cultural Heritage, vol.11, no.1, pp.42–49, 2010.

[3] E. Kwak, I. Detchev, A. Habib, M. El-Badry, and C. Hughes, "Precise photogrammetric reconstruction using model-based image fitting for 3D beam deformation monitoring," Journal of Surveying Engineering, vol.139, no.3, pp.143–155, 2013.

[4] X. Brunetaud, L.D. Luca, S. Janvier-Badosa, K. Beck, and M. Al–Mukhtar, "Application of digital techniques in monument preservation," European Journal of Environmental and Civil Engineering, vol.16, no.5, pp.543–556, 2012.

[5] Y. Ham and M. Golparvar-Fard, "Three-dimensional thermography-based method for cost-benefit analysis of energy efficiency building envelope retrofits," Journal of Computing in Civil Engineering, vol.29, no.4, 2014.

[6] H. Fan, H. Su, and L.J. Guibas, "A point set generation network for 3D object reconstruction from a single image," Proc. IEEE conference on computer vision and pattern recognition, pp.605–613, 2017.

[7] S. Tulsiani, T. Zhou, A.A. Efros, and J. Malik, "Multi-view supervision for single-view reconstruction via differentiable ray consistency," Proc. IEEE conference on computer vision and pattern recognition, pp.2626–2634, 2017.

[8] Y. Sun, Z. Liu, Y. Wang, and S.E. Sarma, "Im2avatar: Colorful 3D reconstruction from a single image," arXiv preprint arXiv:1804. 06375, 2018.

[9] C.-H. Lin, C. Kong, and S. Lucey, "Learning efficient point cloud generation for dense 3D object reconstruction," Thirty-Second AAAI Conference on Artificial Intelligence, vol.32, no.1, 2018.

[10] J.K. Pontes, C. Kong, S. Sridharan, S. Lucey, A. Eriksson, and C. Fookes, "Image2mesh: A learning framework for single image 3D reconstruction," Asian Conference on Computer Vision, pp.365–381, Springer, 2018.

[11] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3D mesh models from single rgb images," Proc. European Conference on Computer Vision (ECCV), pp.52–67, 2018.

[12] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D mesh renderer," Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.3907–3916, 2018.

[13] H. Kato and T. Harada, "Learning view priors for single-view 3D reconstruction," Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.9778–9787, 2019.

[14] A. Kanazawa, S. Tulsiani, A.A. Efros, and J. Malik, "Learning category-specific mesh reconstruction from image collections," Proc. European Conference on Computer Vision (ECCV), pp.371–386, 2018.

[15] C.B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3D object reconstruction," European conference on computer vision, vol.9912, pp.628–644, Springer, 2016.

[16] A. Kar, C. Häne, and J. Malik, "Learning a multi-view stereo machine," Advances in neural information processing systems, pp.365–376, 2017.

[17] M. Ji, J. Gall, H. Zheng, Y. Liu, and L. Fang, "Surfacenet: An end–to-end 3D neural network for multiview stereopsis," Proc. IEEE International Conference on Computer Vision, pp.2307–2315, 2017.

[18] A.A. Soltani, H. Huang, J. Wu, T.D. Kulkarni, and J.B. Tenenbaum, "Synthesizing 3D shapes via modeling multi-view depth maps and silhouettes with deep generative networks," Proc. IEEE conference

on computer vision and pattern recognition, pp.1511–1519, 2017.

[19] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era," IEEE Trans. Pattern Anal. Mach. Intell., vol.43, no.5, pp.1578–1604, 2021.

[20] F.S. Mahad, M. Iwamura, and K. Kise, "Leveraging pyramidal feature hierarchy for 3D reconstruction," International Workshop on Frontiers of Computer Vision, vol.1212, pp.347–362, Springer, 2020.

[21] F.S. Mahad, M. Iwamura, and K. Kise, "Learning pyramidal feature hierarchy for 3D reconstruction," IEICE Trans. Inf. & Syst., vol.E105-D, no.2, pp.446–449, 2022.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," Proc. IEEE conference on computer vision and pattern recognition, pp.2117–2125, 2017.

[23] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," ACM Transactions on Graphics (TOG), vol.35, no.4, pp.1–11, 2016.

[24] A.X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," arXiv preprint arXiv:1512.03012, 2015.

[25] C. Kong, C.-H. Lin, and S. Lucey, "Using locally corresponding cad models for dense 3D reconstructions from a single image," Proc. IEEE conference on computer vision and pattern recognition, pp.4857–4865, 2017.

[26] J.K. Pontes, C. Kong, A. Eriksson, C. Fookes, S. Sridharan, and S. Lucey, "Compact model representation for 3D reconstruction," Proceedings of 2017 International Conference on 3D Vision (3DV), 2017.

[27] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.8, pp.1362–1376, 2009.

[28] D.P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. IEEE conference on computer vision and pattern recognition, pp.770–778, 2016.

[30] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," Proceedings British Machine Vision Conference 2014, 2014.

Fairuz Safwan Mahad is currently a Ph.D. student at Osaka Prefecture University and also a full-time senior machine learning engineer at a tech company. He received his Bachelor of Computer Science majoring in software engineering from University of Technology of Malaysia and Master of Engineering in Electrical Engineering and Information Science from Osaka Prefecture University. His current research interests are 3D reconstruction via active and passive methods such as LiDAR, structured light and stereo vision, deep learning with 3D reconstruction, unsupervised and supervised depth map estimation with deep learning, and machine learning with robotics for industrial use.

Masakazu Iwamura is an Associate Professor at the Department of Core Informatics, Graduate School of Informatics, Osaka Metropolitan University. He received the B.E., M.E., and Ph.D degrees in engineering from Tohoku University, Japan, in 1998, 2000 and 2003, respectively. His research interests include text and object recognition, and visually impaired assistance. He received awards including IAPR/ICDAR Young Investigator Award in 2011, best paper awards of IEICE in 2007 and 2022, IAPR/ICDAR best paper awards in 2007, IAPR Nakano award in 2010, the ICFHR best paper award in 2010, and MVA best paper award in 2017. He worked as the vice-chair of the IAPR technical committee 11 (reading systems) in 2016-2018. He has been an Associate Editor of the International Journal of Document Analysis and Recognition since 2013, and an Associate Editor of the IEICE Transactions on Information and Systems in 2017-2021 and Associate Editor-in-Chief since 2021.

Koichi Kise received the B.E., M.E. and Ph.D. degrees in communication engineering from Osaka University, Osaka, Japan in 1986, 1988 and 1991, respectively. From 2000 to 2001, he was a visiting professor at German Research Center for Artificial Intelligence (DFKI), Germany. He is now a Professor of the Department of Core Informatics, Graduate School of Informatics, Osaka Metropolitan University, Japan. He received awards including the best paper award of IEICE in 2006 and 2022, the IAPR/ICDAR best paper awards in 2007 and 2013, the IAPR Nakano award in 2010, the ICFHR best paper award in 2010 and the ACPR best paper award in 2011. He worked as the chair of the IAPR technical committee 11 (reading systems), a member of the IAPR conferences and meetings committee. He is an editor-in-chief of the international journal of document analysis and recognition. His major research activities are in analysis, recognition and retrieval of documents, images and human activities. He is a member of IEEE, ACM, IPSJ, IEEJ, ANLP and HIS.