Recovering Dense 3D Point Clouds from Single Endoscopic Image

Long Xi[a,1], Yan Zhao[a,2], Long Chen[b,3], Qing Hong Gao[a,4], Wen Tang[a,*,5], Tao Ruan Wan[c,6] and Tao Xue[d,7]

[a]Bournemouth University, Poole, Dorset, BH12 5BB, UK
[b]Lyft Level 5, London, EC2A 3AH, UK
[c]University of Btadford, Bradford, BD7 1DP, UK
[d]Xian Polytechnic University, Xian, Shaanxi, 710048, China

ARTICLE INFO

Keywords:
3D Point Clouds
Monocular Endoscopic Scenes
Artificial Intelligence/ Deep Learning
Augmented Reality
Virtual Reality
Minimally Invasive Surgery

ABSTRACT

Background and Objective: Recovering high-quality 3D point clouds from monocular endoscopic images is a challenging task. This paper proposes a novel deep learning-based computational framework for 3D point cloud reconstruction from single monocular endoscopic images.

Methods: An unsupervised mono-depth learning network is used to generate depth information from monocular images. Given a single mono endoscopic image, the network is capable of depicting a depth map. The depth map is then used to recover a dense 3D point cloud. A generative Endo-AE network based on an auto-encoder is trained to repair defects of the dense point cloud by generating the best representation from the incomplete data. The performance of the proposed framework is evaluated against state-of-the-art learning-based methods. The results are also compared with non-learning based stereo 3D reconstruction algorithms.

Results: Our proposed methods outperform both the state-of-the-art learning-based and non-learning based methods for 3D point cloud reconstruction. The Endo-AE model for point cloud completion can generate high-quality, dense 3D endoscopic point clouds from incomplete point clouds with holes. Our framework is able to recover complete 3D point clouds with the missing rate of information up to 60%. Five large medical *in-vivo* databases of 3D point clouds of real endoscopic scenes have been generated and two synthetic 3D medical datasets are created. We have made these datasets publicly available for researchers free of charge.

Conclusions: The proposed computational framework can produce high-quality and dense 3D point clouds from single mono-endoscopy images for augmented reality, virtual reality and other computer-mediated medical applications.

1. Introduction

Augmented reality (AR) information can help surgeons overcome the limited field of view and the lack of depth information during minimally invasive surgery. The higher the quality of the underlying 3D point cloud is, the more accurate the augmented information becomes [3]. During an endoscopic surgery, endoscopes are used to visualize organ surfaces in the body and the data acquired is the so-called endoscopic images. Constructing 3D point data from the endoscopic image is challenging due to occlusions of instruments, the change of brightness of organ surfaces, and the surface smoothness for feature extractions [16, 4]. Processing an extensive amount of endoscopic image sequences in real-time is a high computational cost, making it difficult to generate high-quality 3D point clouds [24, 5].

Missing data or information from the initially recovered point cloud is common, and it is a shared problem in many applications that rely on high-quality 3D point clouds, for example, AR information augmentation, robotic manipulation [31] and scene understanding [7]. 3D point cloud completion refers to a process that repairs data flaws by filling holes and parts of the dataset. To the best of our knowledge, no prior work has been reported on monocular endoscopic 3D point cloud completion, and the vast majority of point cloud completion researches have been focused on objects, for which specialized 3D shapes (e.g. aircraft, furniture) are learned or manually designed. Large medical *in-vivo* databases of 3D point clouds of real endoscopic scenes are scarcely publicly available. We believe that the availability of such databases will significantly assist in research innovations. We generate seven new medical datasets and make them freely available to research communities.

In this paper, a novel computational framework has been proposed to recover dense 3D point clouds from single endoscopic images using two deep learning neural networks. One is for monocular depth learning, and the other is for 3D point cloud completion to recover the missing data from the initially generated point clouds. Figure **??** shows the workflow of our proposed framework. The experimental results indicate that our 3D reconstruction method outperforms the state-of-the-art learning-based method and non-learning based stereo 3D reconstruction algorithms with an average Chamfer distance 0.01514 mm on our synthetic medical datasets. 3D point cloud Completion results also show a better performance of our Endo-AE compared with the state-of-the-art learning-based methods. An average Chamfer distance 0.00236 mm has been obtained when the missing input data rate is 20% on the testing datasets. Even if the missing rate reaches 60%, the quality of the completion result is still high,

*Corresponding Author: Wen Tang
**Wen Tang
lxi@bournemouth.ac.uk (L. Xi); zhaoy@bournemouth.ac.uk (Y. Zhao); alwaysunny@gmail.com (L. Chen); qgao@bournemouth.ac.uk (Q.H. Gao); wtang@bournemouth.ac.uk (W. Tang); t.wan@bradford.ac.uk (T.R. Wan); xt73@163.com (T. Xue)
ORCID(s): 0000-0003-0472-0876 (L. Xi)

with the average Chamfer distance 0.00804 mm.

Main contributions of this paper are:

- We propose a novel computational framework to recover high-quality 3D point clouds from single endoscopic images by combining two deep learning neural networks. One is for monocular depth learning, and the other is for 3D point cloud completion.
- Five large medical *in-vivo* databases of 3D point clouds are generated from public Laparoscopic/Endoscopic video datasets [12, 22], and two synthetic 3D medical datasets are also created. Our 3D point clouds are extracted from every frame of the video datasets. Our datasets are publicly available at[1].

2. Related Work

Our approach is closely related to two categories of prior works: 1) Monocular Depth Estimation and 2) 3D Point Cloud Completion.

3D Monocular Depth Estimation: Depth estimation is an integral part of 3D point cloud reconstruction. The state-of-the-art camera tracking and reconstruction systems (structure for motion systems) that estimate detailed depth maps with textures at selected keyframes can produce dense surface maps with millions of points [24]. Some of these systems rely on powerful commodity GPU processors for real-time performance and stereo visions. On the other hand, monocular SLAM (Simultaneous Localization and Mapping) systems that operate with limited processing resources only generate and track sparse feature-based models [23, 5].

Recent advances in monocular depth estimation have shown results of predicting the depth from a single image [10, 9], which can be used for understanding the shape of a scene from a single image, a fundamental problem in machine vision. These methods pose the monocular depth estimation as a learning problem by training models offline [18, 10, 19]. Among these methods, supervised learning [10, 9] needs to train models on large collections of the ground truth. Novel unsupervised learning methods explore easier-to-obtain binocular stereo footage without the need for explicit depth data during the training [13, 6]. In our work, since the ground truth of depth information is unavailable for monocular endoscope scenes, we build on our previous unsupervised learning framework [6] to develop a monocular depth estimation for 3D point cloud reconstruction from single endoscopic images. The novelty of our approach is a fully differential patch-based cost function, and we propose to use the Zero-Mean Normalized Cross-Correlation that takes multi-scale patches as a matching strategy. This approach significantly increases the accuracy and robustness of depth learning. However, this method has only been tested with non-medical public datasets. We further extend the method to extract the dense 3D endoscopic point cloud based on the estimated depth and introduce a colour extraction method onto a reconstructed 3D point cloud from a single endoscopic image.

3D Point Cloud Completion: Real endoscopic 3D point clouds present incomplete data (e.g., missing data, holes), due to limited field of view and occlusions during minimally invasive surgery where surgical instruments interact with the organs, as well as the illumination variations caused by the endoscopic light, tissue hemorrhaging and or surgical smoke [16]. Hence, we cast the task of filling missing holes and information for reconstructed 3D point clouds as the task of 3D shape completion. We devise a new computational method for 3D endoscopic point cloud completion and evaluate the effectiveness of the proposed method [32, 1, 15].

Traditional geometry-based approaches use geometric cues to complete 3D shapes from a partial input [21], while data-driven based methods rely on the assumption that the database must include a very similar shape [29]. Recently emerged deep learning-based methods have achieved superior performance on shape completion using voxel-based techniques [8] or directly operating on point clouds through generative models based on Auto Encoder (AE) [17, 1, 15, 30] and Generative Adversarial Net (GAN) [14].

An optimization method has been proposed to select the best seed for the latent GAN to improve the performance for point cloud completion [15]. Structural point cloud decoder [30] can only generate sparse 3D point clouds since the decoder consists of most multi-layer perception (MLP) networks. Each MLP needs to generate a recovered point cloud, which limits the number of points to be processed. Achlioptas et al. [1] proposed an auto-encoder architecture for 3D point cloud processing. However, this method focuses on 3D point cloud representation learning and has only been tested with non-medical public datasets. We apply the auto-encoder architecture to recover dense endoscopic 3D point clouds, and our approach is the first attempt of its kind applied to endoscopic 3D point cloud completion. We conduct experiments to compare our Endo-AE with raw GAN and l-GAN to evaluate the completion results (see Section 7.2.2).

3. Overview of the Framework

Figure **??** illustrates the entire computational framework with three modules: Monocular Image Depth Learning, 3D Point Extraction and 3D Point Cloud Completion.

Monocular Image Depth Learning Module: In this module, an unsupervised learning network is developed. Public Laparoscopic/Endoscopic stereo video datasets are used for the network training. Our unsupervised depth learning method treats the monocular depth estimation as error minimization in image synthesis. During the training, the depth is estimated from the left image of stereo pairs. The depth is then converted into a disparity map to synthesize the right image of stereo pairs. The loss function is used to minimize the error between the reconstructed right image and the original right image. Once trained, the depth information is generated from monocular endoscopic images in the depth learning module.

[1]We make the datasets publicly available to researchers at https://github.com/LONG-XI/Endoscopic-3D-Point-Clouds-Datasets/

3D Point Cloud Extraction Module: In the 3D point cloud extraction module, the depth estimated from the depth learning module is converted into a dense 3D point cloud. A coordinate conversion method is used to transform the pixel coordinates into the 3D world coordinates. To obtain the colour information, colour attributes of the corresponding input monocular endoscopic image is extracted and applied to the 3D point cloud.

The effectiveness of the proposed 3D point cloud reconstruction framework is evaluated by comparing our method with a state-of-the-art learning-based method [13], as well as with two non-learning based stereo image reconstruction methods [2]. The detailed evaluation is described in section 7.1.1.

3D Point Cloud Completion Module: In the 3D point cloud completion module, a generative Endo-AE network based on an auto-encoder is performed for the task of 3D endoscopic point cloud completion. We split 3D point clouds generated in the 3D point cloud extraction module into the training data and the testing data. The auto-encoder, as an unsupervised network, uses the training data itself as the ground truth. During the training, the input of the network is complete 3D point clouds without any missing data, as shown in Figure **??**. The network learns global features of training datasets through an encoder and converts global features into an original 3D point cloud through a decoder. During the testing mode, by randomly deleting consecutive points in the testing data, our trained model can generate a complete 3D point cloud from the partial 3D point cloud input, as shown in Figure **??**. The colour attributes are also extracted from the corresponding 3D point cloud in the testing data.

The effectiveness of the proposed generative model for 3D point cloud completion is evaluated by comparing our method with a generative adversarial network [1]. The detailed comparison is presented in section 7.2.2.

4. Methods

4.1. Unsupervised Monocular Depth Learning

Building on our previous unsupervised mono-depth network [6], we estimate the per-pixel depth from single image input. We incorporate the patch matching theory and achieve unsupervised training. The mono-depth network is based on a VGG-like fully convolutional neural network architecture [20], as shown in Figure **??**.

During the training, the single left images I_l of stereo pairs are used as the input data for our DepthNet model to synthesize per-pixel depth D. The depth D is transformed into a disparity map $d = \frac{b \times f}{D}$, where b and f are the camera baseline and focal distance, respectively. The disparity map d is then used to reconstruct the right view of the stereo pairs I_{r_syn} and the sampling of patches $I_r(N_{x-d,y})$. Finally, a fully differential loss function L_{total} is proposed to train our mono-depth network. L_{total}, as illustrated in Equation 1, consists of a Patch Matching Loss L_{PM}, a View Reconstruction Loss L_{VR}, a Disparity Smoothness Loss L_{DS}, and a Disparity Consistency Loss L_{DC}. In addition, another parallel ConfidenceNet, as shown in Figure **??**, is trained by using the proposed L_{PM} to evaluate the performance of the monocular depth estimation. The ConfidenceNet produces a confidence map that gives a real-time assessment of the reliability of the predicted depth.

During the testing, the trained mono-depth model does not need the original right image of stereo pairs to calculate the loss anymore. Thus, the trained model can generate per-pixel depth only from the monocular image as shown in Figure **??**.

Loss Function: L_{PM} is proposed to maximize the similarities between patches in the input left image and shifted patches in the reconstructed right image by using the Zero-Mean Normalized Cross-Correlation that takes multi-scale patches as a matching strategy. L_{VR} minimizes the differences between the original input left image and its reconstruction using the $L1$ norm. L_{DS} regularizes our mono-depth network to produce more smooth depth by calculating the sum of the $L1$ norm of disparity gradients along x and y directions. L_{DC} attempts to make the left-view disparity map to be equal to the reconstructed right-view disparity map using the $L1$ norm. The loss function L_{total} is defined by Equation 1.

$$L_{total} = \omega_p L_{PM} + \omega_v L_{VR} + \omega_d L_{DS} + \omega_c L_{DC} \quad (1)$$

where ω is the corresponding weight to balance the effect of gradients of the back propagation.

The back propagation is defined to update parameters θ of our mono-depth learning network to minimize the L_{total} using Equation 2.

$$\begin{aligned} \frac{\partial L_{total}}{\partial \theta} = & \frac{\partial L_{PM} + \partial L_{VR} + \partial L_{DS} + \partial L_{DC}}{\partial F_{warp}(I_l, d) + \partial F_{sample}(I_r, d)} \\ & \times \frac{\partial F_{warp}(I_l, d) + \partial F_{sample}(I_r, d)}{\partial d} \\ & \times \frac{\partial d}{\partial D} \times \frac{\partial D}{\partial F_{depth}(I_l, \theta)} \times \frac{\partial D = F_{depth}(I_l, \theta)}{\partial \theta} \end{aligned} \quad (2)$$

We refer readers to our paper [6] for more information about the unsupervised monocular depth learning network.

4.2. 3D Point Cloud Extraction

3D point cloud extraction is the second module of our proposed framework shown in Figure **??**. A 3D point cloud is extracted from the generated depth D, as described in section 4.1. A coordinate conversion method from the pixel coordinates to the world coordinates is applied to 3D point cloud extraction. Based on the generated depth D, 3D point clouds can be extracted using Equation 3.

$$\begin{aligned} x_w &= (u - u_0) * D / f_x \\ y_w &= (v - v_0) * D / f_y \\ z_w &= D \end{aligned} \quad (3)$$

where (x_w, y_w, z_w) is the coordinates of a point in the world coordinate system and (u, v) is each pixel in the depth D.

(u_0, v_0) are the centre coordinates of the depth D in pixel coordinate system. f_x and f_y are the focal lengths of the left and right camera.

While reconstructing a 3D point cloud from the generated depth D, the colour attributes of each pixel are extracted from the corresponding left image and assigned to each point in the 3D point cloud, as shown in Figure **??**.

4.3. 3D Point Cloud Completion

For the point cloud completion task, we train an Endo-AE network based on Auto-encoder (AE) that includes an encoder and a decoder to generate complete 3D point clouds from partial 3D point clouds with missing data. Since our Endo-AE is an unsupervised network, the ground truth is the input training 3D point cloud itself. The encoder of our Endo-AE network is based on PointNet [27], a state-of-the-art deep learning method on 3D point cloud classification. PointNet combines point wise multi-layer perceptions with a symmetric aggregation function that is invariant to permutation, which is essential for effective feature learning on 3D point clouds. The main differences between PointNet [27] and our Endo-AE network are the loss function, the ground-truth and the output of the two networks. PointNet focuses on 3D point cloud classification, and the loss function of the PointNet classification network is softmax, which can be considered a multi-classes classifier. Every 3D point cloud has a label for classification, and each label is the ground-truth for PointNet. Thus, the PointNet classification network outputs the label of an input 3D point cloud. Whereas the loss function of our Endo-AE network is the Chamfer distance, as illustrated in Equation 4, which minimizes the distance between the input and the output of 3D point clouds. The ground truth of our Endo-AE network is the input training 3D point cloud itself, and the output is the complete 3D point cloud.

During the training, the input of our Endo-AE network is the complete 3D point cloud without missing data, and the output 3D point cloud is the reconstruction of the input. The input and output 3D point clouds in training mode are shown in Figure **??**. A 3D point cloud with N points is represented as a $N \times 3$ the matrix, and each row of matrix is the 3D coordinates of a point defined as $P_i = (x, y, z)$. The encoder compresses an input 3D point cloud of N points into a k dimensional feature vector $v \epsilon \mathbb{R}^k$. Specifically, a shared multi-layer perception (MLP) with an activation function ReLU and a batch-normalization are used to transform each point P_i into a point feature vector F. A point wise max pooling is placed after all MLPs, ensuring the global features are invariant to any permutations of a 3D point cloud and producing a k-dimensional feature vector. The decoder aims to generate the reconstruction of the input 3D point cloud based on the learned k-dimensional feature vector by using three fully connected layers. Thus, the learned global features can represent the 3D point cloud for the point cloud completion task.

During the testing, the input of the trained completion model is partial 3D point clouds, and our trained model can output complete 3D point clouds based on learned global features extracted from the encoder. The input and output 3D point clouds in testing mode are shown in Figure **??**.

Loss Function: The loss function for 3D point cloud completion measures the difference between the generated 3D point cloud S_{pred} and the ground-truth S_{gt}. The loss is defined to be invariant to any permutation of 3D point clouds in both S_{pred} and S_{gt}. We use the Chamfer distance [11] (CD) to measure the difference between S_{pred} and S_{gt}, as shown in Equation 4.

$$CD(S_{pred}, S_{gt}) = \frac{1}{S_{pred}} \sum_{p \epsilon S_{pred}} \min_{q \epsilon S_{gt}} ||p - q||_2 + \frac{1}{S_{gt}} \sum_{q \epsilon S_{gt}} \min_{p \epsilon S_{pred}} ||q - p||_2 \quad (4)$$

The Chamfer distance calculates the average nearest point distance between S_{pred} and S_{gt} by finding the closest neighbour with $O(nlogn)$ complexity. In addition, S_{pred} and S_{gt} can be 3D point clouds with different sizes.

5. Implementation Details

We conduct our experiments in two stages. We train a mono-depth network to predict depth for 3D point cloud reconstruction. We then achieve the 3D point cloud completion based on reconstructed point clouds with a trained Endo-AE network. Our unsupervised mono-depth network and Endo-AE network are trained on an Nvidia Titan X GPU with 12G memory and a CPU with 32G memory. The implementation details for the two networks are explained as follows:

Hyper Parameters: In terms of training mono-depth network, all input images are resized to 512×256 with a batch size of four. Adam optimizer with an initial learning rate of 0.0001 and 50 epochs are used for the training process. The weights defined in our total loss are $\omega_p = 0.5$, $\omega_v = 1$, $\omega_d = 0.1$ and $\omega_c = 1$, respectively. In addition, 6 skip connections are implemented, preserving intermediate information during training to ensure the high quality of per-pixel depth estimation. The first four kernel sizes of the encoder are 7, 7, 5, and 5, followed by ten kernel sizes of 3. The kernel size of the decoder in each layer is the reverse order in the encoder.

The encoder of our Endo-AE network consists of five layers of shared multi-layer perception (MLP) with 64, 128, 128, 256 and 128 filters, respectively. The decoder consists of three fully connected layers with 256, 256 and 4096×3 filters, respectively. We also use Adam optimizer with an initial learning rate of 0.0005, a batch size of 50 and 500 epochs. Our point cloud completion network is trained with the input size of $M_1 \times 3$ and the ground truth size of $M_2 \times 3$, generating the output point cloud with the size of $M_3 \times 3$, where M_1, M_2 and M_3 can be any number. In our experiments we set $M_1 = M_2 = M_3 = 4096$.

Data Augmentation: To increase the robustness of our mono-depth network and prevent over-fitting, we randomly

flip images and change the brightness and colour of images. During the Endo-AE network training, we augment 3D point clouds by applying a random rotation matrix.

6. Evaluation Metrics

We evaluate the 3D point cloud reconstruction method with Chamfer distance (Equation 4). We evaluate the 3D point cloud completion model with three evaluation metrics [1], which are minimum matching distance (MMD), coverage (COV) and Jensen-Shannon Divergence (JSD), respectively.

MMD: MMD calculates the average distance in the matching between two 3D point clouds. MMD_CD is based on the Chamfer distance (Equation 4), and MMD_EMD is about the Earth Mover's distance (EMD) [28].

$$EMD(S_{pred}, S_{gt}) = \min_{\phi} \sum_{p \epsilon S_{pred}} \|p - \phi(p)\|_2 \tag{5}$$

where $\phi : S_{pred} \rightarrow S_{gt}$ is bijection. The EMD distance minimizes the distance between S_{pred} and S_{gt} with $O(n^2)$ complexity. Note that EMD requires the same sizes of S_{pred} and S_{gt}.

A major difference between MMD_CD and MMD_EMD is that calculating MMD_EMD is too expensive with $O(n^2)$ complexity and takes more time than calculating MMD_CD with $O(nlogn)$ complexity. Another major difference between them is that MMD_CD can calculate the average distance between two point clouds with different sizes, whereas calculating MMD_EMD requires the two 3D point clouds to have the same sizes.

Coverage: Coverage measures the rate of bijection relationship between the output point cloud set and the ground-truth set. For each 3D point cloud x_i in output point cloud set X, we find its closest ground truth point cloud y_j in the ground truth set Y based on the minimum CD or EMD between them. Assume there is a bijection $g : x_i \epsilon X \rightarrow y_j \epsilon Y$, where x_i is the output point cloud in the output set X, and y_j is the ground truth point cloud in the ground truth set Y. $G = \{\langle x_i, g(x_i)\rangle\}$ is a set that includes all the pairs that meet the bijection condition, where each $g(x_i)$ is the corresponding y_j with respect to x_i. Coverage is defined by Equation 6.

$$Coverage = \frac{CardG}{CardY} \tag{6}$$

where $Card$ means the number of elements in one set.

The range of the coverage result is between 0 and 1, and the higher coverage value indicates the higher matching result.

JSD: JSD is a probability value that measures the similarity between two probability distributions and is based on the Kullback-Leibler (KL) divergence $D_{KL}(P_A||P_B)$.

$JSD(P_A||P_B)$ and D_{KL} are defined by Equation 7 and 8:

$$JSD = \frac{1}{2} D_{KL}(P_A||M) + \frac{1}{2} D_{KL}(P_B||M) \tag{7}$$

$$D_{KL}(P||M) = \sum Plog\frac{P}{M} \tag{8}$$

where $M = \frac{1}{2}(P_A + P_B)$, A and B are the generated point cloud and the ground truth, respectively. P_A and P_B are two probability distributions of the generated point cloud and the ground truth. To calculate P_A and P_B, we firstly set a 3D grid with the size of 21952×3 and normalize the 3D grid and 3D point cloud between -1 and 1. We then calculate the K-Nearest Neighbor to find the corresponding points between the 3D grid and 3D point cloud, where K is 1. We consider indices of matched points in the 3D grid as the probability distribution of a 3D point cloud. P in Equation 8 can be P_A or P_B. The result of JSD will become 0, if $P_A = P_B$. Thus, the smaller result of JSD indicates the better performance of the completion result.

7. Results and Discussions

7.1. 3D Point Cloud Reconstruction

In this section, we firstly compare our 3D point cloud reconstruction method with a state-of-the-art learning-based method Godar et al. [13] and two non-learning based stereo image reconstruction methods [2]. Secondly, we generate 3D endoscopic point cloud datasets based on our 3D point cloud reconstruction method.

7.1.1. Comparison experiments

Evaluation Datasets: There are some endoscopic datasets generated by previous researcher projects, such as EndoVis 2019 Sub-challenge dataset[1], Laparoscopic Image to Image Translation Dataset [26] and EndoAbS dataset [25]. EndoVis 2019 Sub-challenge dataset may not be publicly downloadable. Laparoscopic Image to Image Translation Dataset includes simulated monocular images and the corresponding depth. However, this dataset does not provide camera parameters and ground truth 3D point clouds or ground-truth stereo correspondences that are required to train our mono-depth learning network. EndoAbs dataset consists of images of kidney, liver and spleen, captured under different lighting conditions and smoke, and it also contains 20 ground truth 3D point clouds that are captured by a laser scanner and camera parameters. We thank the authors of the EndoAbS dataset, who have kindly provided us with 120 stereo images of phantoms and 20 ground truth 3D point clouds. To minimize the error during the evaluation, we manually transform (translate and rotate) the ground truth 3D point clouds in the EndoAbS dataset to match the views of the images.

We also create a synthetic medical dataset for the network training by capturing stereo images and the corresponding ground truth 3D point clouds under the same view using a 3D computer modeling software[2]. Specifically, we capture 200 stereo images and their corresponding depth images of each left frame from a 3D liver model and a heart model. The ground truth 3D point clouds are extracted from 200 left frames and 200 corresponding depth images using Equation 3.

[1] https://endovissub2019-scared.grand-challenge.org/
[2] https://www.autodesk.com/products/maya/

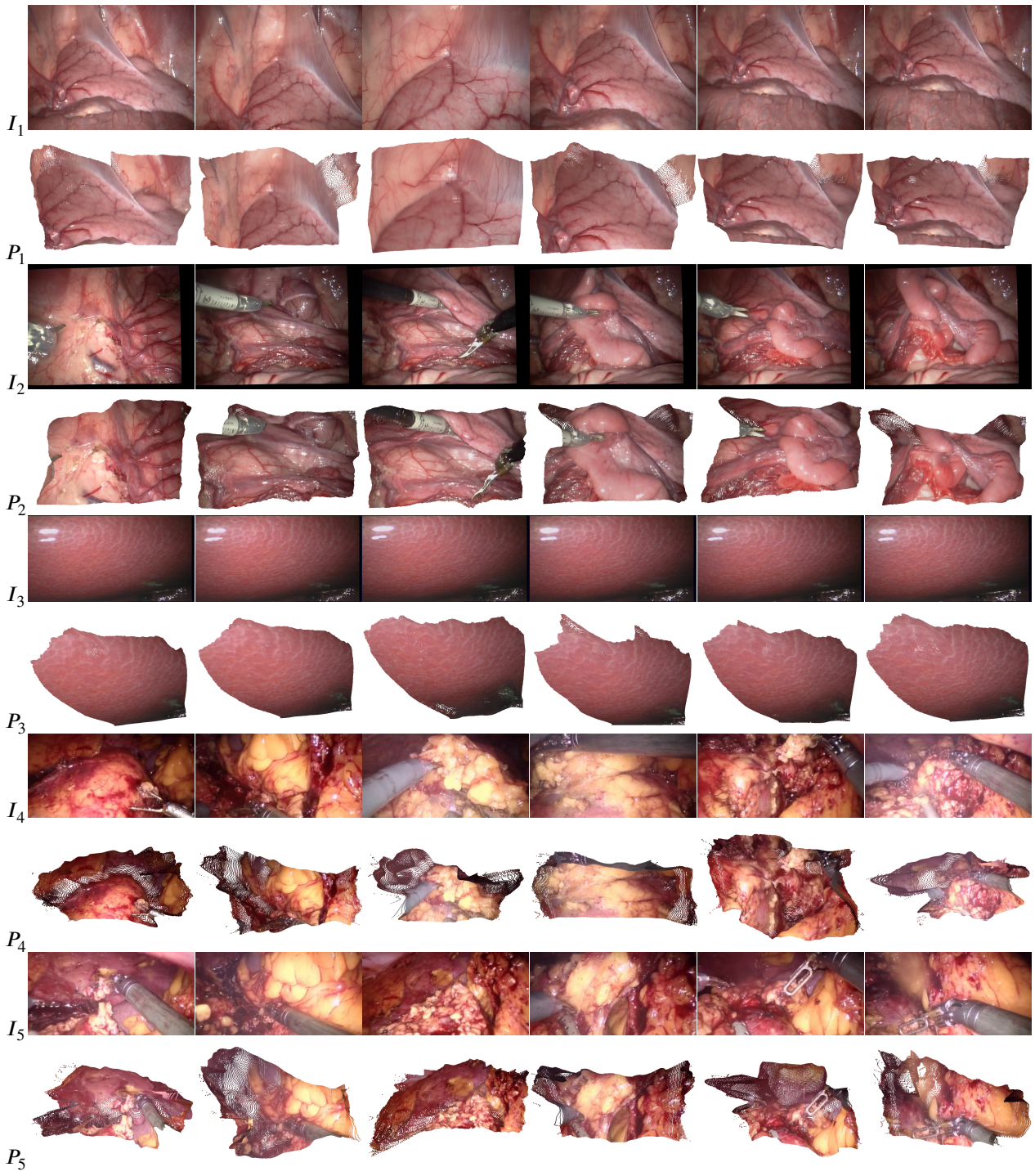

Figure 1: Results of 3D point cloud reconstructions of public Laparoscopic/Endoscopic video: The 3D point cloud is extracted from every frame of the video. I_i represents Images, and P_i is the corresponding extracted 3D point clouds related to I_i. Note that I_4 and I_5 are from the same video stream, but the scene changes from the middle of the video. Thus, we divided it into two separate datasets.

Comparison with Learning-Based Method: We train the Godar [13] and our mono-depth networks on the public available Laparoscopic/Endoscopic video datasets [12] [22]. We first compare the performance of these two methods on the EndoAbS dataset. The reconstructed 3D point clouds are shown in (b) and (e) in Figure **??**. The Chamfer distance (CD) is calculated between reconstructed 3D point clouds and transformed ground truth 3D point clouds. The CD results on the kidney, liver and spleen for our 3D reconstruction method are 0.54533mm, 0.41444mm and 0.07512mm, showing better performance than Godar's with CD results 0.76253mm, 0.49351mm and 0.09610mm.

We also compare our 3D reconstruction method with Godar's on our synthetic medical dataset, as shown in (b) and (e) in Figure **??**. The CD results in Figure **??** show that our method also outperforms Godar's method on our synthetic medical dataset. In addition, we also calculate the average CD on our synthetic medical dataset for our 3D reconstruction method (0.01514mm), which outperforms Godar's method (0.01902mm).

Comparison with Stereo Image Reconstruction Methods: To assess the effectiveness of learning-based methods with non-learning based stereo-image reconstruction methods, we compare our method with non-learning based block matching method[3] (BM) and semi-global block matching (SGBM) [2]. BM and SGBM directly use low-level image features to search for matched pixels in the left and right images of stereo pairs. As a result, the quality of the generated 3D point cloud is usually poor, as shown in (c) and (d) in Figure **??** and Figure **??**. The CD results show that our method outperforms BM and SGBM on both the EndoAbS dataset and our synthetic medical dataset. In addition, the average CD result on the synthetic medical dataset for our method is 0.01514mm, which shows better performance than BM and SGBM with 0.29784mm and 0.49407mm.

7.1.2. Generated Endoscopic Datasets

We generate 3D endoscopic point cloud datasets based on depth information generated from public available Laparoscopic/Endoscopic video datasets [12] [22] using our 3D point cloud reconstruction method.

Five stereo endoscopic videos from the datasets [12] [22] are chosen to generate depth information, including Abdomen Wall, Uterine Horn, Liver, Nephrectomy scene 1 and Nephrectomy scene 2, respectively. The stereo videos are divided into 35,000 left frames and 35,000 right frames. We randomly select around 10,000 Nephrectomy left frames and 10,000 Nephrectomy right frames as the input to train our mono-depth network. Once the model has been trained, our mono-depth network can generate the per-pixel depth from the monocular image. We generate approximately 35,000 depth images from 35,000 left frames, as shown in Figure **??**. The five left frames in Figure **??** are randomly selected from Abdomen Wall, Uterine Horn, Liver, Nephrectomy scene 1 and Nephrectomy scene 2, respectively.

Based on the generated depth images, we generate approximately 35,000 *in-vivo* 3D point clouds by using Equation 3. Our datasets consist of five endoscopic point cloud categories, including Abdomen Wall, Uterine Horn, Liver, Nephrectomy scene 1 and Nephrectomy scene 2. Our datasets are made publicly available for researchers, which can be used for learning-based methods as training datasets and evaluating 3D reconstruction methods. Each dense 3D point cloud contains approximately 100,000 points on average. We divide each category into six parts and display the first frame of each part, as shown in Figure 1. The margins of endoscopic images, i.e., the black margins in I_2, are due to the movement of the instrument that is not horizontally and vertically positioned. Thus, we remove the margins when extracting 3D point clouds. As shown in Figure **??**, we also synthesize two additional datasets using synthetic 3D models of a heart and a liver by applying different affine transformation and rotation matrices. Each synthetic dataset contains 2,000 point clouds, while each point cloud is randomly down-sampled to 4096 points. Thus, our datasets contain five classes of *in-vivo* datasets and two classes of synthetic datasets, including approximately 39,000 3D point clouds in total.

[3] https://opencv.org

7.2. 3D Point Cloud Completion

7.2.1. Evaluation of Completion Performance

For the 3D point cloud completion task, we train five class-specific Endo-AE networks separately with our five classes of endoscopic point cloud datasets generated in section 7.1.2. We also train a two-classes Endo-AE network with two synthetic 3D models, as mentioned in section 7.1.2.

For each class, we randomly select 90% of 3D point clouds as the training data and the remaining 10% as the testing data. To evaluate our trained Endo-AE model on partial 3D point clouds, we use the remaining 10% of testing data to create partial 3D point clouds with different missing rates. First, we randomly select a point from each testing 3D point cloud with the size of $N \times 3$, where N is the total number of points in a 3D point cloud. Second, we delete the nearest $N * delete\_rate$ points around that selected point to create partial 3D point clouds with different missing rates, where $delete\_rate$ is the rate of deletion, i.e., 0.2, 0.4, 0.7, etc. Third, we randomly sub-sample each partial 3D point cloud to 4096 points. Finally, for each class of testing datasets, we generate 7 groups of partial 3D point clouds testing data with various missing rates of [20%, 30%, 40%, 50%, 60%, 70%, 80%]. The examples of partial 3D point clouds with 60% and 20% missing data are shown in (c) and (e) in Figure **??**.

The visualizations of 3D point cloud completion results with 60% and 20% of missing data for our five class-specific Endo-AE networks are shown in Figure **??**. All completion results of MMD_CD and MMD_EMD in (f) are smaller than that in (d), indicating that the less missing input data, the more accurate the recovered 3D point clouds. P_1, P_2, P_3, P_4 and P_5 show the effectiveness of our method with the 20% and 60% of missing rates. The completion result is unstable in the dataset of P_2 when the missing rate reaches 60%. The unstable recovery is due to the large deformation of the organ in the video, which is caused by surgical instruments. Thus, it is difficult for a neural network to extract common global features for all different deformation degrees. In addition, we visualize an example of a completion result on a partial point cloud with multiple missing areas and a 20% of the missing rate, as shown in Figure **??**, which indicates that our Endo-AE can process various partial 3D point clouds with multiple missing areas. Figure **??** and Figure **??** show that our Endo-AE model can deal with the partial input 3D point cloud with various missing areas and rates.

We calculate the K-nearest neighbour (KNN) to find the corresponding points between the ground-truth 3D point cloud

and our generated 3D point cloud, where K is 1. After finding the nearest neighbour between them, we extract the colour of each point from the ground-truth and duplicate it to the corresponding point in our generated 3D point cloud, as shown in Figure **??** and Figure **??**.

We also visualize two-classes Endo-AE network completion results with 50% of missing regions on simulated 3D heart and liver models, as shown in Figure **??**, which presents superior completion results.

The evaluation results on our datasets with 50% of missing data are reported in Table 1. The results of MMD_CD on the synthetic heart (D6) and liver (D7) are relevantly smaller than others, and the values of COV are still high with 50% of missing data, showing a better performance of our completion model on synthetic data due to the completion result contains less noise. Abdomen Wall (D1) and synthetic heart (D7) in Table 1 have similar MMD_CD and show different JSD. There are two possible reasons. First, the scale of the completed point cloud is relatively small compared with the original ground-truth when the missing rate reaches 50%, such as the last image of the completed liver model shown in Figure **??**. Since the JSD is sensitive to the scale between two point clouds, D1 and D7 show different results of JSD. Second, the model has been trained with two-classes synthetic liver and heart, which indicates that the scale problem of the generated point cloud is also caused by the multi-classes training.

Table 1
Three evaluation metrics on seven datasets with the missing rate of 50%.

	MMD_CD (mm)	COV (%)	JSD (%)
D1	0.00336	49.0	9.950
D2	0.01464	55.0	15.869
D3	0.00391	41.0	11.557
D4	0.00873	42.0	10.259
D5	0.00765	52.0	12.398
D6	0.00318	66.0	18.992
D7	0.00358	78.0	27.079

Figure **??** shows the average values of each evaluation metric calculated between the generated point cloud and the ground truth on our 7 groups of partial 3D point clouds testing data. MMD_CD, MMD_EMD, COV and JSD show that the error of missing data repairing increases gradually as more regions are occluded. The average MMD_CD on our seven datasets is 0.00236mm when the missing rate of input data is 20%. Even if the missing rate reaches 60%, the quality of our completion result is still good with the average MMD_CD 0.00804mm. The results of MMD_EMD on Nephrectomy scene 1 (D4) show better performance when the missing rate is 40% than 20% and 30%. The reason is that some partial inputs in testing sets are similar to each other due to the missing data of incomplete point clouds is randomly removing consecutive points from original point clouds. The results of COV show that the completion errors are increasing as the missing rate increases from 20% to 80%. From the results of JSD, Liver (D3) shows better performance when the missing rate is less than 50%, but the completion error increases rapidly after the missing rate reaches 60%. The reason is that the scale of the complete point cloud is relatively small compared with the ground-truth when the missing rate reaches 60%, and the JSD is sensitive to the scale between two point clouds.

7.2.2. Comparison With GANs

We compare our generative auto-encoder based Endo-AE with another generative network, generative adversarial network (GAN), by training a raw GAN and a latent-space GAN (l-GAN) [1], respectively.

Raw GAN operates directly on 3D point clouds, and the generator of the raw GAN consists of five fully connected layers with ReLU, producing a 4096×3 point cloud. The architecture of the discriminator is identical to the encoder of our Endo-AE with leaky ReLUs and without any batch normalization, and a sigmoid activation function is placed after the discriminator.

We train a raw GAN on a single class of Liver and show one of the completion results in Figure **??**. The MMD_CD and MMD_EMD of raw GAN are 0.00138 mm and 0.14223 mm, showing lower performance than our Endo-AE with MMD_CD 0.50478 mm and MMD_EMD 0.65328 mm. The average values are also calculated to evaluate the raw GAN on the testing dataset of the Liver in terms of MMD, COV and JSD, respectively. When the missing rate of testing data is 20%, the average values of MMD_CD, COV and JSD are 0.01086 mm, 14.0% and 28.940%, respectively. Whereas the average values of these three evaluation metrics based on our Endo-AE are 0.00142 mm, 78% and 5.6%, showing better performance than raw GAN. In addition, our Endo-AE also outperforms raw GAN on single-class training when the missing rate of testing data reaches 30%, 40%, 50%, 60%, 70% and 80% based on the average values of these three evaluation metrics.

Raw GAN is not able to generate a specific 3D point cloud with respect to the input data for multi-classes training, which means the output of raw GAN can be similar to any data in multi-classes datasets. Figure **??** shows the output of raw GAN and Endo-AE, respectively. The input data (b) is a 3D point cloud of the Abdomen Wall. Endo-AE can generate the complete 3D point cloud (d) according to the input (b), but the output of raw GAN (c) is similar to the data in Uterine Horn. We also train an l-GAN, passing data through a pre-trained Endo-AE, the same problem with raw GAN occurs. Thus, although raw GAN and l-GAN can produce complete 3D point clouds, it is not reliable for endoscopic 3D point cloud completion.

In addition, even raw GAN and l-GAN are trained with a single class of endoscopic 3D point clouds, the same problem still occurs. One major reason is that GAN randomly generates data based on the whole training set and does not find the best representation of the corresponding partial input 3D point cloud. Another reason is the specialist of endoscopic data. Because the *in-vivo* endoscopic data is varied

and deformable, differences can be large even between the closest frames in the same dataset.

7.3. Discussion

Figure **??** and Figure **??** indicate that our 3D reconstruction method outperforms the state-of-the-art learning-based method (Godar) and non-learning based stereo 3D reconstruction algorithms (BM and SGBM) on our synthetic medical datasets and the EndoAbS dataset. The limitation of our mono-depth model is that it will produce inconsistent depth on low-light and textureless surfaces.

Figure **??** shows the high-quality completion results of our Endo-AE model on our seven datasets in terms of various missing rates. Figure **??** and Figure **??** also illustrate that our Endo-AE outperforms GANs. The limitation of our proposed Endo-AE network is that the output of the complete 3D point cloud is regenerated rather than repaired by increasing the number of points in missing areas.

8. Conclusion

We develop a novel framework to recover dense 3D point clouds from single endoscopic images. Our framework includes an unsupervised mono-depth network that generates the depth from a single endoscopic image. Based on the mono-depth learning network, a dense 3D point cloud can be extracted from an endoscopic image. We create *in-vivo* 3D endoscopic point cloud datasets and make the datasets publicly available to researchers. A generative Endo-AE network is then trained to complete 3D point clouds with various degrees of missing data. Our results show the capability of our computational framework for producing dense 3D endoscopic point cloud datasets and its effectiveness in repairing defects of real endoscopic point cloud datasets and synthetic medical models.

In future work, we would like to extend our 3D point cloud completion network to generate points only in missing areas instead of regenerating all points in the output. In addition, It would also be interesting to deal with multi-classes training for 3D point cloud completion.

References

[1] Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L., 2017. Learning representations and generative models for 3d point clouds. arXiv preprint arXiv:1707.02392 .

[2] Boykov, Kolmogorov, 2003. Computing geodesics and minimal surfaces via graph cuts, in: Proceedings Ninth IEEE International Conference on Computer Vision, pp. 26–33 vol.1.

[3] Chen, L., Day, T.W., Tang, W., John, N.W., 2017a. Recent developments and future challenges in medical mixed reality, in: 2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), IEEE. pp. 123–135.

[4] Chen, L., Tang, W., John, N.W., 2017b. Real-time geometry-aware augmented reality in minimally invasive surgery. Healthcare technology letters 4, 163–167.

[5] Chen, L., Tang, W., John, N.W., Wan, T.R., Zhang, J.J., 2018. Slam-based dense surface reconstruction in monocular minimally invasive surgery and its application to augmented reality. Computer methods and programs in biomedicine 158, 135–146.

[6] Chen, L., Tang, W., Wan, T.R., John, N.W., 2020. Self-supervised monocular image depth learning and confidence estimation. Neurocomputing 381, 272–281.

[7] Dai, A., Ritchie, D., Bokeloh, M., Reed, S., Sturm, J., Nießner, M., 2018. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4578–4587.

[8] Dai, A., Ruizhongtai Qi, C., Nießner, M., 2017. Shape completion using 3d-encoder-predictor cnns and shape synthesis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5868–5877.

[9] Eigen, D., Fergus, R., 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: Proceedings of the IEEE international conference on computer vision, pp. 2650–2658.

[10] Eigen, D., Puhrsch, C., Fergus, R., 2014. Depth map prediction from a single image using a multi-scale deep network, in: Advances in neural information processing systems, pp. 2366–2374.

[11] Fan, H., Su, H., Guibas, L.J., 2017. A point set generation network for 3d object reconstruction from a single image, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 605–613.

[12] Giannarou, S., Stoyanov, D., Noonan, D., Mylonas, G., Clark, J., Visentini-Scarzanella, M., Mountney, P., Yang, G., 2012. Hamlyn centre laparoscopic/endoscopic video datasets.

[13] Godard, C., Mac Aodha, O., Brostow, G.J., 2017. Unsupervised monocular depth estimation with left-right consistency, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 270–279.

[14] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets, in: Advances in neural information processing systems, pp. 2672–2680.

[15] Gurumurthy, S., Agrawal, S., 2019. High fidelity semantic shape completion for point clouds using latent optimization, in: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. pp. 1099–1108.

[16] Haouchine, N., Dequidt, J., Peterlik, I., Kerrien, E., Berger, M.O., Cotin, S., 2013. Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery, in: 2013 IEEE international symposium on mixed and augmented reality (ISMAR), IEEE. pp. 199–208.

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

[18] Ladicky, L., Shi, J., Pollefeys, M., 2014. Pulling things out of perspective, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 89–96.

[19] Liu, F., Shen, C., Lin, G., Reid, I., 2015. Learning depth from single monocular images using deep convolutional neural fields. IEEE transactions on pattern analysis and machine intelligence 38, 2024–2039.

[20] Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440.

[21] Mitra, N.J., Guibas, L.J., Pauly, M., 2006. Partial and approximate symmetry detection for 3d geometry. ACM Transactions on Graphics (TOG) 25, 560–568.

[22] Mountney, P., Stoyanov, D., Yang, G.Z., 2010. Three-dimensional tissue deformation recovery and tracking. IEEE Signal Processing Magazine 27, 14–24.

[23] Mur-Artal, R., Montiel, J.M.M., Tardos, J.D., 2015. Orb-slam: a versatile and accurate monocular slam system. IEEE transactions on robotics 31, 1147–1163.

[24] Newcombe, R.A., Lovegrove, S.J., Davison, A.J., 2011. Dtam: Dense tracking and mapping in real-time, in: 2011 international conference on computer vision, IEEE. pp. 2320–2327.

[25] Penza, V., Ciullo, A.S., Moccia, S., Mattos, L.S., De Momi, E., 2018. Endoabs dataset: Endoscopic abdominal stereo image dataset for benchmarking 3d stereo reconstruction algorithms. The Interna-

tional Journal of Medical Robotics and Computer Assisted Surgery 14, e1926.

[26] Pfeiffer, M., Funke, I., Robu, M.R., Bodenstedt, S., Strenger, L., Engelhardt, S., Roß, T., Clarkson, M.J., Gurusamy, K., Davidson, B.R.a., 2019. Generating large labeled data sets for laparoscopic image processing tasks using unpaired image-to-image translation .

[27] Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 652–660.

[28] Rubner, Y., Tomasi, C., Guibas, L.J., 2000. The earth mover's distance as a metric for image retrieval. International journal of computer vision 40, 99–121.

[29] Sung, M., Kim, V.G., Angst, R., Guibas, L., 2015. Data-driven structural priors for shape completion. ACM Transactions on Graphics (TOG) 34, 1–11.

[30] Tchapmi, L.P., Kosaraju, V., Rezatofighi, H., Reid, I., Savarese, S., 2019. Topnet: Structural point cloud decoder, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 383–392.

[31] Varley, J., DeChant, C., Richardson, A., Ruales, J., Allen, P., 2017. Shape completion enabled robotic grasping, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 2442–2447.

[32] Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J., 2016. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling, in: Advances in neural information processing systems, pp. 82–90.