2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Self-Guided Instance-Aware Network for Depth Completion and Enhancement

Zhongzhen Luo[1], Fengjia Zhang[2], Guoyi Fu[1], Jiajie Xu[2]

***Abstract*— Depth completion aims at inferring a dense depth image from sparse depth measurement since glossy, transparent or distant surface cannot be scanned properly by the sensor. Most of existing methods directly interpolate the missing depth measurements based on pixel-wise image content and the corresponding neighboring depth values. Consequently, this leads to blurred boundaries or inaccurate structure of object. To address these problems, we propose a novel self-guided instance-aware network (SG-IANet) that: (1) utilize self-guided mechanism to extract instance-level features that is needed for depth restoration, (2) exploit the geometric and context information into network learning to conform to the underlying constraints for edge clarity and structure consistency, (3) regularize the depth estimation and mitigate the impact of noise by instance-aware learning, and (4) train with synthetic data only by domain randomization to bridge the reality gap. Extensive experiments on synthetic and real world dataset demonstrate that our proposed method outperforms previous works. Further ablation studies give more insights into the proposed method and demonstrate the generalization capability of our model.**

I. INTRODUCTION

Depth sensor plays a crucial role in many robotics applications that require an interpretation of the scene. For example, [1] use semantic keypoints as object representation to plan robot trajectories by using one of high-end commodity-level depth sensors. In spite of the recent advances in depth sensing technology, object surfaces such as shiny, glossy, transparent, and distant pose challenges for depth measurements as shown in Fig. 1. One promising attempt is to alternatively capture the images sequentially in different orientations, combining multiple views, and manually annotate ground truth data. However, such solution requires prohibitively expensive cost and suffers from dynamic objects and high latency, which makes it a less practical solution for real-time robotics applications. Towards this end, the goal of our work is to develop an affordable and efficient solution for depth completion that only train on synthetic data, and test with a single view of a commodity-level RGB-D camera.

Traditional methods to estimate the missing depth values can be achieved by explicitly using the hand-tuned methods [2], [3], or finding the local affinity or discontinuity based on interpolation and diffusion schemes [4]. With the recent explosive growth of deep learning techniques, several methods have been introduced into this task and reveal promising improvements [5]–[7]. However, they basically share the same scheme by directly using the element-wise addition operation for feature fusion. Applying such a simple way leads to blurred boundaries and inaccurate shape reconstruction. Recent approaches [8] aim to mitigate these failures with local representation prediction from color images, and solve for depth completion via a traditional optimization problem. But we observe that such multi-stage algorithms often fail easily in cluttered scenarios where edges and instances are ambiguous.

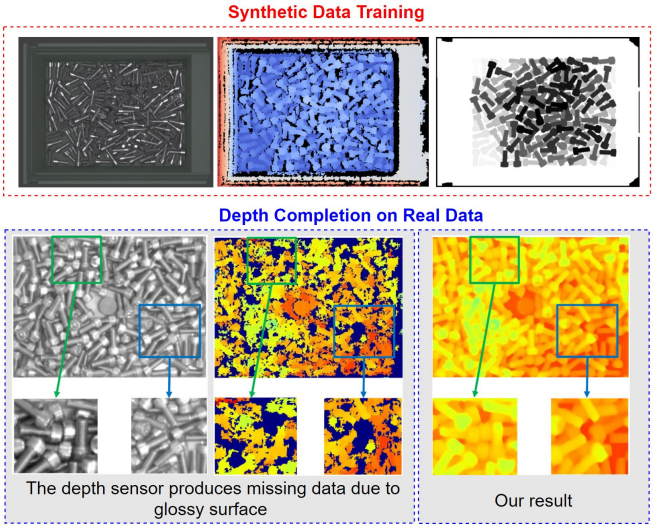

Fig. 1: Sample illustration of our approach. Our approach is trained on synthetic data only. At test time, the same model is used in real world with no additional training.

In contrast, as convolutional neural network (CNN) have been successful in learning effective representations in object detection [9]–[11], and pose estimation [12], [13]. These works demonstrate that CNN are very powerful for understanding semantic context, which inspire us the study of instance level features. One of our observations from this perspective is that the geometric property that position relationship between points of a rigid instance in 3D space is fixed. Inspired by the benefits of keypoint voting in [14], we argue that to address occlusion problems requires pixel-wise estimation of directions from each pixel towards the instance keypoints (e.g., center, corner, distinguished feature). Secondly, inspired by the success of guided image filtering [15], we introduce an unsupervised self-guided module, to learn the dynamic feature for each channel and each spatial location, hence the proposed methods will pay more attention on

[1]Epson Research, Markham, Ontario, Canada. {zhongzhen.luo, gary.fu}@ea.epson.com
[2]University of Toronto, Toronto, Ontario, Canada. {fuuka.zhang, jeremy.xu}@mail.utoronto.com

relevant information. Furthermore, we leverage the negative log-likelihood of the observed depth as uncertainty into the cost term, which not only force the network to find semantic clues from color information, but also enable the network to attenuate the effort on the observed region. In addition, we handle the issues of vague structures and multiple objects with context information concepts and global geometric constraint. Instead of predicting depth values only, we equip our network to concentrate on instance-aware consistency via joint tasks learning. By jointly optimizing with instance-aware tasks, we find that it can boost the performance of each other.

To summarize, the main contributions of our work are the followings: (1) We develop a novel end-to-end network for depth completion and enhancement by jointly considering the instance-level features, and utilizing geometric constraint and context information to deals with occlusion and vague structures. (2) We also propose to use unsupervised self-guided mechanism together with negative log-likelihood function to learn a dynamic feature selection for each channel at each spatial location of layers to enhance the depth completion performance. (3) We present domain randomization to bridge the gap between synthetic and real so that we can train our model by using synthetic data only. (4) We demonstrate improvements of our approach compared to the state-of-the-art (SOTA) approaches on both synthetic and real KITTI datasets. Because our proposed method train on synthetic data only, we also evaluate on our real world in-house robotics dataset to demonstrate the practical ability for robustness and generalization.

II. Related Work

Depth completion has been intensively studied since the emergence of active depth sensors. We briefly review these techniques and other literatures relevant to our network design.

A. Image-only method

The topics gained popularity recently by using neural networks to learn depth information from pictorial cues. Eigen et.al was the first to use multi-scale neural network to achieve coarse-to-fine prediction [16]. Since then, various learning-based approaches were developed based on monocular color images [17]–[19]. Yang et al. [20] introduced surface normal representation for geometric constrain. Built upon this, they then further introduced edge consistency in parallel with surface normal consistency for fine detailed structures recovery [21]. Recent work in this field focus on improving the prediction accuracy and reducing pixel relative error through various methods, such as depth discretization (DORN) [22], 3D geometric constraints (VNL) [23], and local planar guidance (BTS) [24]. These methods share some ideas with our work. However, the motivation of these works is usually to compensate the constraints on camera size, cost and image quality. And therefore, they are not suitable for predicting accurate pixel level values on semi-dense depth measurement, specifically on the pixels that are missing from raw depth measurement.

B. Depth with image method

Several approaches attempted to use guidance from additional image input. Eigen et al. [25] proposed to use CNN for multitasks: depth, surface normal and semantic segmentation. Laina et al. [26] used fully CNNs, encompassing residual learning architecture to model the mapping between monocular images and depth maps. Another sub-topic in this area focused on reconstructing dense depth map by augmenting sparse depth map with single color image. Ma et al. [27] first introduced this method and achieved good performance on KITTI dataset. Ma et al. [28] further improved this method to train without semi-dense annotations, while still outperformed previous supervised training solutions. Recent depth completion research focus on fixing the missing area from relatively dense RGB-D data. Zhang et al. [8] extracted surface normal and occlusion boundary as the feature representation from RGB data. This effectively conquer the issue that the traditional deep regression method simply learns to copy or interpolate values from the context. Build upon this work, Huang et al. [29] further emphasize the depth reconstruction on object boundaries, by adding boundary consistency into the pipeline. However, their work still lack of global features, and fail easily in cluttered scenarios.

III. Proposed Method

Given the raw depth input D, the learning-based depth completion can be summarized as the process of seeking to find a predictor f_θ such that the predicted depth map $\hat{D}$ is as close as possible to the ground truth depth map D_{gt}. Formally, the depth completion process is formulated to minimize the objective function $\mathcal{L}$ of the form as: $\mathcal{L} = \sum_{k=1}^{n} C(f_\theta(I_k, D_k), D_{gt})$, where I_k and D_k are the pixel-wise image and depth value respetively. $C(,)$ is a certain measure of distance between the ground truth depth map and the predicted depth map, and n is the sets of image pixels of the same scene.

A. Network Overview

To tackle the task as fomulated above, we propose an end-to-end framework that takes an RGB image and a depth image as inputs and produce a complete depth image. The overall architecture is illustrated at Fig. 2. Inspired by recent methods [30], the whole network mainly consists of three branches, the image branch, depth branch and stack branch, as described in the following subsections.

*1) **image branch**:* The goal of the image branch is to compute an increasingly coarse but high-dimensional feature representation of the image by a series of ResNet blocks. This is followed by an Atrous Spatial Pyramid Pooling (ASPP) module [31] with dilated convolution for effective incorporation of hierarchical context information. Then the up-sampling part is equipped with skip-connection layers and shared pooling masks for corresponding max pooling and un-pooling layers. To learn a function over the prediction error,

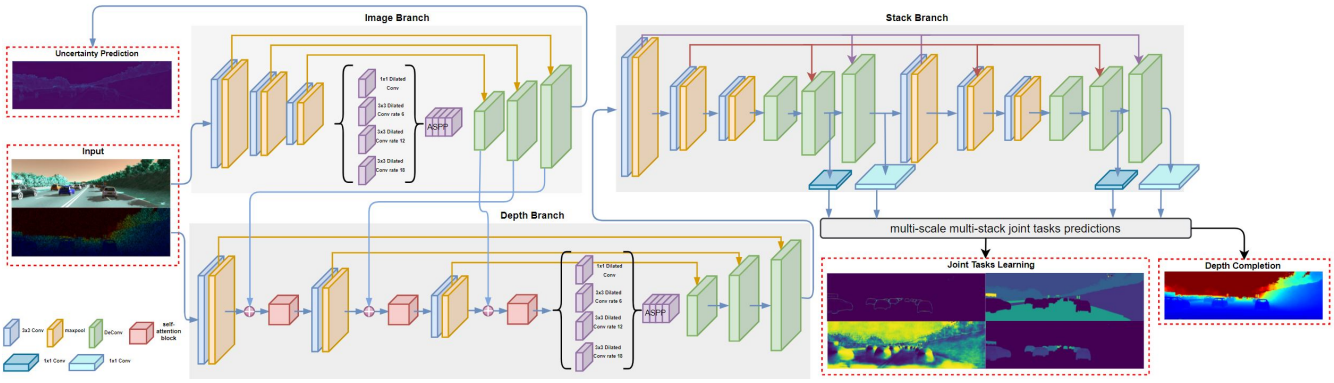

Fig. 2: Overall pipeline of our proposed method: The image branch takes image as input, and the depth branch takes depth as input. The stack branch takes the fusion features from depth branch and performs coarse-to-fine multi-scale joint tasks learning.

we employ an uncertainty map, which consists of training the network to infer mean and variance of the distribution $\mathbf{P}(D_{gt}|D,I)$ of parameters Θ. Hence the network is trained by log-likelihood maximization:

$$\log(\mathbf{P}(D_{gt}|W)) = \frac{1}{n}\sum_{k=1}^{n}\log(\mathbf{P}(D_{gt}|\Theta(I,D,W))) \quad (1)$$

where W is the training weights. As presented in [30], the predictive distribution can be modelled as Laplace distribution, because it allows the model to attenuate the cost of non-missing regions and to focus on reconstructing on missing region via image-to-depth fusion. Therefore, we derive the cost term $\mathcal{L}_u$ by minimizing the following:

$$\mathcal{L}_u = \frac{|\mu(D) - D_{gt}|}{\sigma(D)} + \log(\sigma(D)) \quad (2)$$

where $\mu(D)$ and $\sigma(D)$ are the outputs of the model encoding mean and variance of the distribution, while being subject to the additional logarithmic term to discourage infinite predictions. Moreover, the learned uncertainty map can also serve to gauge the reliability of depth completion at run time.

*2) **depth branch**:* The depth branch is to fuse the features from depth and image features together. Commonly used copy or interpolation operation can easily fall into local minima instead of predicting precise depth values. To avoid this problem, we proposed to use self-guided module, as shown in red box of Fig. 2, in depth branch, such that the network is self-guided to forward important features by paying more attention not only to global features, but also to instance level features. Therefore, the fusion module allows the depth branch to learn dynamic feature selection for each channel and each spatial location. The self-guided module is formulated as:

$$F_{out} = \sigma(\sum_{k=1}^{n} W_g \otimes (D \uplus I)) \odot \delta(\sum_{k=1}^{n} W_f \otimes (D \uplus I)) \oplus D \quad (3)$$

where D and I are features from depth and image branch. $\uplus$ represents concatenation operation of layers, $\otimes$ represents convolution operation, $\oplus$ represents element-wise summation and $\odot$ denotes the element-wise multiplication. While σ is sigmoid function, δ can be any other activation functions, such as ReLU, Linear or Tanh. W_g and W_f are two different convolutional filters for each activation function.

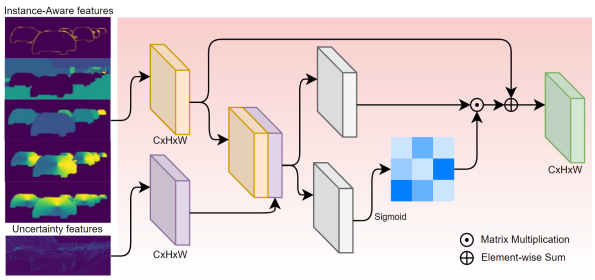

Fig. 3: Sample of instance-aware learning with self-guided module

*3) **stack branch**:* The stack branch is designed to consist of repeated encoder-decoder architecture with intermediate output. This allows the network for re-evaluation and re-assessment of initial estimates and helps to maintain precise local information while considering and then reconsidering the overall coherence of the features. With such stacked hourglass architecture, our network performance has been improved compared to baseline as shown in the ablation study in Section IV-D.

B. Instance-Aware Learning

Leveraging the complementarity properties of geometry and context information for instance-aware features, we propose to jointly solve these tasks so that one boosts the performance of another as shown in Fig. 3. Instead of a distillation multi-module proposed by [32], we propose to split the tasks at the last layers so that all the tasks share knowledge through the network and guided by the self-guided module.

Firstly, we propose to learn the instance center consistency by localizing the object center in the image and estimating

object distance from the camera. We predict pixel-wise unit vectors that represent the direction from each pixel to center of the object. Predicting pixel-wise directions alleviates the influence of cluttered background since the invisible part of the object can be correctly located from other visible parts in terms of the directions. In addition, we use object distance label to strengthen the network ability to handle object with similar shape but different depth values. Therefore, our first task is to preserve center consistency $\mathbf{T}$ such that: $\mathbf{T} = (v_x = \frac{p_x - c_x}{|\mathbf{p}-\mathbf{c}|}, v_y = \frac{p_y - c_y}{|\mathbf{p}-\mathbf{c}|}, T_z)$, where p_x and p_y are 2D pixel position in image. c_x and c_y are instance center position, and T_z is the semantic label for instance distance. To handle sensor noise or inaccurate depth measurement, we adopt smooth L1 loss terms $\mathcal{L}_1$ for the task of center consistency.

Secondly, our model predict semantic information to handle the foreground and background differentiation. Given the pixel-wise extracted feature, semantic segmentation task $\mathcal{L}_{ss}$ is supervised by the binary cross entropy loss. Thirdly, we also introduce object boundary consistency to enforce model to preserve the clear boundaries in the output depth, making the depth image to be more structured and conformed to realistic situation. The boundary consistency task $\mathcal{L}_{bc}$ is also supervised by binary cross entropy loss. Because depth and surface normal are two strongly correlated factors, the locally linear orthogonality between them can be utilized to regularize the depth completion. Motivated by [33], we also estimate the surface normal and incorporate a depth-normal consistency term for the normal guided depth completion into a loss function. The depth-normal consistency loss is defined as: $\mathcal{L}_{sn} = \sum_{p,q \subseteq n} || < V(p,q), N(p) > ||^2$, where $<,>$ denotes an inner product. The $\mathcal{L}_{sn}$ measures the consistency such that the inner product of the vector from point p to its neighbor q should be zero with surface normal vector $N(p)$.

Therefore, in order to guide the update of the network weights to measure how far the estimated depth map $\hat{D}$ is from the ground truth depth map D_{gt}, we design the cost terms in our pipeline as: $\mathcal{L}_{total} = \lambda_1 \mathcal{L}_u + \lambda_2 \mathcal{L}_1(\mathbf{T}_{gt}, \hat{\mathbf{T}}) + \lambda_3 \mathcal{L}_1(D_{gt}, \hat{D}) + \lambda_4 \mathcal{L}_{ss} + \lambda_5 \mathcal{L}_{bc} + \lambda_6 \mathcal{L}_{sn}$. The first two terms measure the errors between the ground truth and the estimated depth while the rest instance-aware task terms are used to incorporate various geometry and context constraints.

C. Domain Randomization

To bridge the reality gap and make our model to transfer to the real world, we used domain randomization [34] by randomly perturbing the following aspects of the scene: (1) texture of background images taken from the Flickr 8K [35] dataset; (2) position and intensity of point lights, in addition to a planar light for ambient illumination; (3) position and orientation of camera with respect to the scene (pan, tilt, and roll from $-30°$ to $30°$, azimuth from $0°$ to $360°$, elevation from $5°$to $30°$); (4) object material and color with respect to the scene; (5) number and types of depth missing region with respect to input depth map; and (6) number and types of sensor noise with respect to input images. While (1)-(6) are applied to our Epson inhouse dataset, only (5) and (6) are applied on virtual KITTI Dataset.

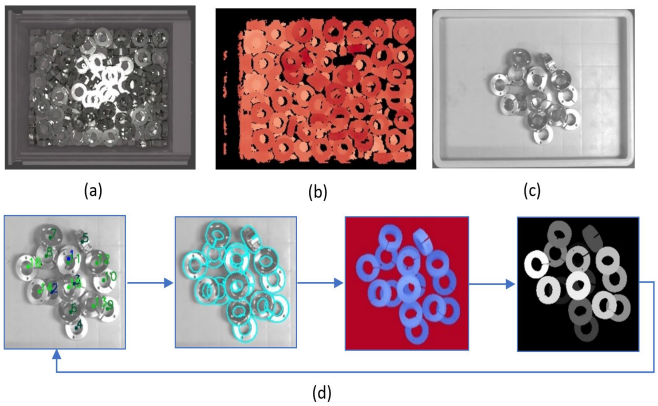

Fig. 4: (a) Synthetic image (b) Synthetic depth (c) Real world image (b) Iterative ground truth labeling process on Epson robotics dataset.

IV. Experiments

We conduct extensive experiments on benchmark dataset: virtual KITTI [36] and real KITTI [37]. Moreover, to demonstrate the generalization ability, we also perform experiments on our Epson in-house dataset in different scenarios. Note that the best results are marked with **bold**. ↓ means smaller is better while ↑ mean larger is better.

A. Dataset

Virtual and Real KITTI Dataset The Virtual KITTI dataset has 21260 stereo pairs in total, including: scene1 (crowded urban area): 4470, scene2 (road in urban area then busy intersection): 2330, scene6 (stationary camera at a busy intersection): 2700, scene18 (long road in the forest with challenging imaging conditions and shadows): 3390, and scene20 (highway driving scene): 8370. In our experiments, the plan to select the train-test split is the following: 1) choose the entire scene6 as test set, 2) choose first 88% frames from each scene category among the remaining ones for training set, the next 6% frames for validation set and hence the last 6% frames for test set, that is resulting in a training set of 16334 images, a validation set of 1113 images, and a testing set of 3813 images. We only use the Real KITTI Datase for evaluation purpose. As there are rare measurement at many areas of images due to LiDAR scans, so the evaluation is set to only count on measurement points.
Epson Robotics Dataset is composed of synthetic data for training and real world data for testing. We use the PyBullet [38] as our physics simulator and Blender [39] as our rendering engine to generate synthetic training data as shown in Fig. 4(a) and Fig. 4(b). To be able to evaluate in our real world data, we manually conduct iterative ground truth generation process until the error is less than 1mm, as shown in Fig. 4(d). For each object, the synthetic data contains over 5k training and 1k validation images, and the real test images have about 500 scenes, which is captured

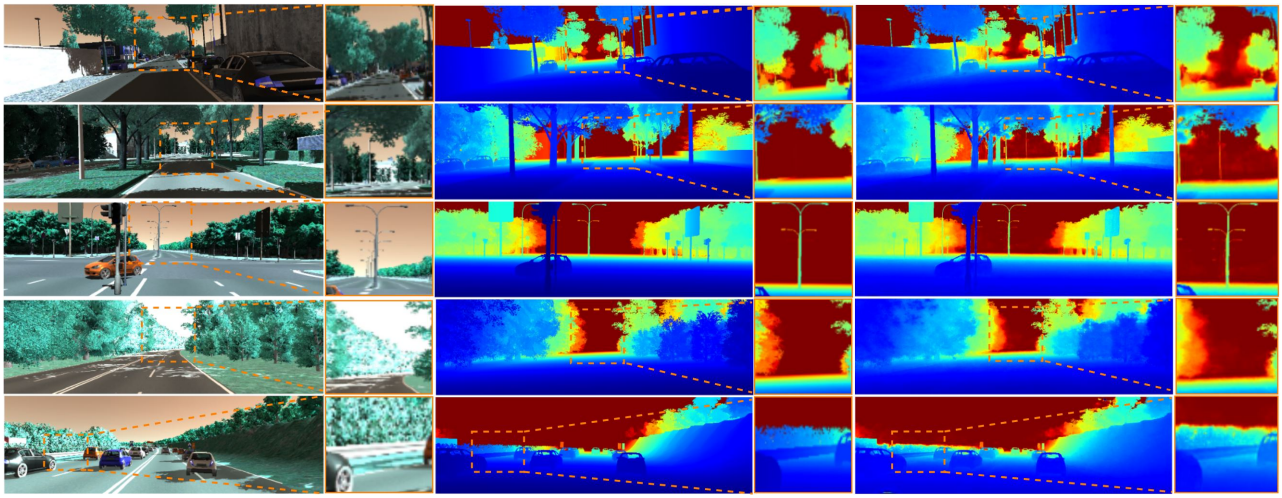

Fig. 5: Qualitative results on virtual KITTI datase (5% sparse depth input). From left to right: image, ground truth depth, and our depth prediction. Compared to ground truth depth, our method is able to predict fine-grained details, learned from fusion features, that even ground truth has not included.

in different lighting conditions, bin background colors, bin positions and orientations with a commodity-level RGB-D camera. In our experiment, we have chosen 4 distinct objects that represent shinny, thin and deformable shape scenarios.

B. Evaluation Metrics

We adopt the standard metrics for the evaluation on virtual KITTI dataset: Root Mean Square Error (RMSE), Mean Absolute Error (MAE). Following the real KITTI benchmark, we also adopt two additional metrics: root mean squared error of the inverse depth (iRMSE) and mean absolute error of the inverse depth (iMAE).

C. Evaluation Performance

Table I shows the quantitative results for depth completion on virtual KITTI dataset. Because the sparsity of depth points increase with the range, the results present the performance in the range of 20 m, 50 m and 100 m, respectively. Table I shows our method outperforms SOTA in both range of 0-50m and 0-100m, while it is comparable to SOTA in 0-20m range. As shown by Fig. 5, our method enhance depth estimation with fine-grained details, that even ground truth has not included. Therefore, the predicted depth value at fine-grained region is expected to be different from ground truth value. Consequently, the average error between ground truth and predictions has been increased in these regions.

Method	0-20 (cm)		0-50 (cm)		0-100 (cm)	
	MAE ↓	RMSE↓	MAE ↓	RMSE↓	MAE ↓	RMSE↓
MRF [40]	56.67	116.776	131.03	312.41	209.45	575.20
TGV [41]	41.85	114.57	113.38	323.97	205.78	621.48
STD [42]	258.98	386.91	653.54	1066.55	1072.52	1892.04
SCNN [37]	56.44	137.34	153.01	384.96	258.23	681.13
Ours	22.46	**84.77**	**61.03**	**211.68**	**78.19**	**271.21**

TABLE I: Quantitative comparison with methods on virtual KITTI dataset.

To demonstrate the generalization of our proposed method quantitatively in real world scenario, we also test our method on the real KITTI dataset as presented in Table II. Different from other methods which train on real KITTI dataset, we train on virtual KITTI dataset only. As can be seen from Table II, our proposed method has achieved very close to SOTA, with 4.01% below for MAE, 4.91% below for iMAE, and 2.5% below for iRMSE. Nevertheless, we outperform SOTA for RMSE by 9.81%. This is because we enforce our model for instance aware consistency by joint tasks learning in order to regularize the depth estimation and mitigate the impact of noise. Our model also recover fine-grained details for depth completion as shown in Fig. 6. Because the ground truth of real KITTI dataset also has not demonstrated this fine-grained details due to sparse measurement of LiDAR scans, higher error between ground truth and predictions in these areas are also expected.

Method	MAE (cm) ↓	RMSE (cm) ↓	iMAE (1/km) ↓	iRMSE (1/km) ↓
DFuse [43]	120.66	429.93	3.62	1.79
MorNet [44]	104.54	310.49	3.84	1.57
CSPN [45]	101,96	279.46	2.93	1.15
HMSNet [46]	93.75	258.48	2.93	1.14
NConv [47]	82.99	233.26	2.60	**1.03**
STD [48]	81.47	249.95	2.80	1.21
DNorm [49]	**77.71**	235.17	**2.42**	1.13
Ours	81.01	**214.13**	2.55	1.16

TABLE II: Quantitative comparison with methods on real KITTI dataset. Note that our method train with synthetic data only

To show the benefits for our robotics manipulation tasks, we also evaluate the performance on our Epson in-house robotics dataset as shown in Fig. 7. From the Table III, our network improve the sensor depth quality significantly. Overall, we improve MAE by 24.8% and RMSE by 16.8% in non-missing region, while improve 1.96% for MAE and 9.95% for RMSE in missing region.

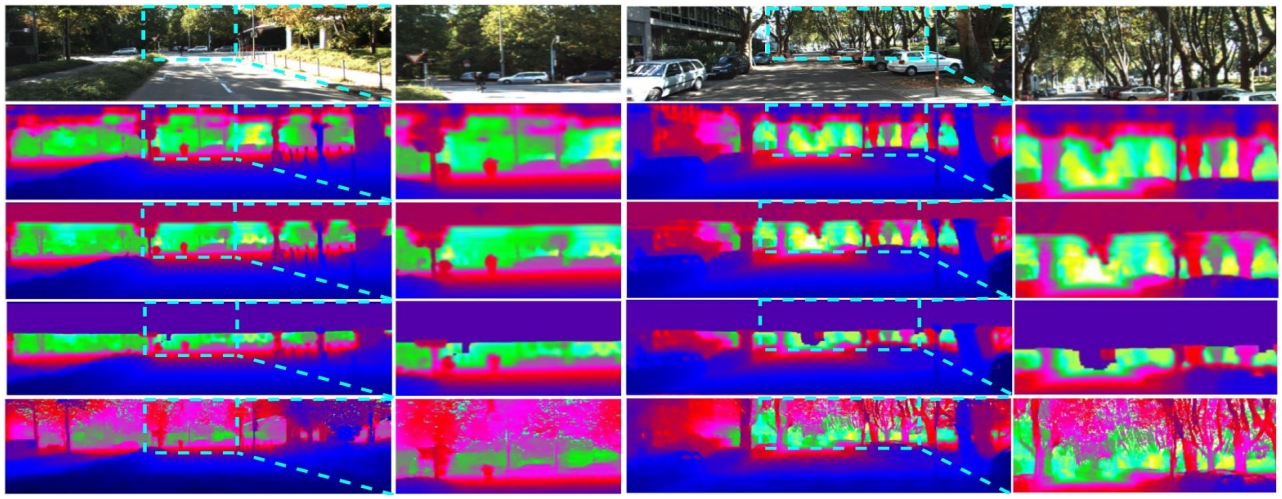

Fig. 6: Qualitative results on real KITTI dataset. From top to bottom: DFuseNet [43] Morph-Net [44], SparseConv [37] and Ours. As we can see from the zooming region, our method is able to recover depth values for fine-grained details even at the far end.

Object	Non-missing Region				Missing Region	
	Sensor (mm)		Ours(mm)		Ours(mm)	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
Object 1	2.74	4.39	2.15	3.42	2.36	3.10
Object 2	1.43	2.54	0.77	2.17	1.54	2.60
Object 3	2.44	4.13	1.96	3.79	2.56	4.42
Object 4	0.53	1.11	0.49	0.74	0.54	0.83

TABLE III: Performance evaluation on Epson robotics dataset

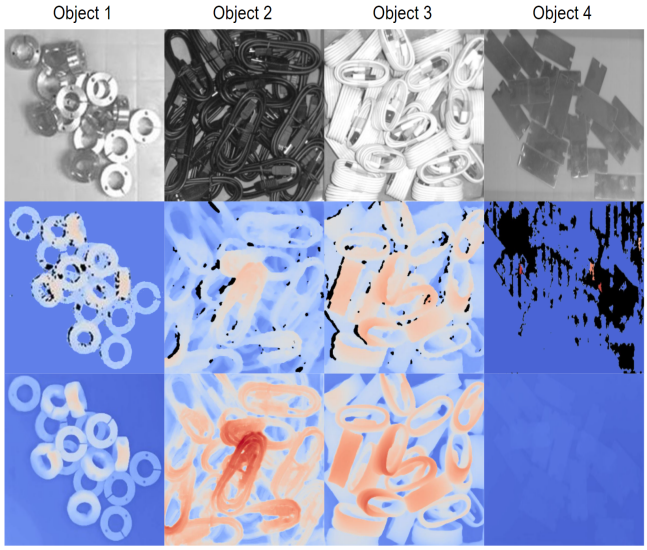

Fig. 7: Qualitative results on Epson robotics dataset. From top to bottom: grayscale image, original low resolution depth, and ours. From our results, our model not only predict missing depth, but also recover fine-grain details.

D. Ablation Study

For better understanding of our approach, we explore ablation studies on how each module boost the performance for depth completion task. We remove all modules and denote the depth prediction only as our baseline. The experimental results Table IV show that after integrating with all modules, the performance of depth completion present the best promising results. This highlights a major advantage of instance aware learning, which strengthen the model's ability to predict depth values based on pixel-wise semantic label of instance features.

Method	MAE (cm) ↓	RMSE (cm) ↓	iMAE (1/km) ↓	iRMSE (1/km) ↓
Baseline	224.31	338.96	3.68	1.86
+ stack	161.81	310.27	3.54	1.77
+ aspp	113.96	274.46	2.91	1.65
+ self-guided	97.21	261.64	2.73	1.36
Full w/o IA-L	92.31	248.07	2.65	1.33
Full w/ IA-L	**81.01**	**214.13**	**2.55**	**1.16**

TABLE IV: Ablation study: performance changes on real KITTI dataset with each component of our proposed model.

V. Conclusion

In this work, we present a novel end-to-end self-guided instance-aware network (SG-IANet) for depth completion and enhancement. We introduce geometry and context learning, especially for occluded or truncated objects, to predict clearer and sharper structures of object. In addition, we also introduce instance-aware learning and show mutual improvements. Incorporate with muti-scale levels of feature maps from image branch of model, we show that the self-guided module allows the network to learn to focus on meaningful features for accurate depth completion. Last but not least, we propose to utilize domain randomization to improve robustness and generalization to unseen environment. Extensive experiments demonstrate that our proposed method outperforms on synthetic dataset, and comparable performance on real dataset though our model is trained on synthetic only. Moreover, ablation studies validate the effectiveness of proposed modules.

REFERENCES

[1] W.Gao and R.Tedrake, "kpam-sc: Generalizable manipulation planning using keypoint affordance and shape completion," in *arXiv preprint arXiv:1909.06980*, 2019.

[2] K.Matsuo and Y.Aoki, "Depth image enhancement using local tangent plane approximations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, p. 3574–3583.

[3] D.Doria and R.Radke, "Filling large holes in lidar data by inpainting depth gradients," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 65–72.

[4] R. M. D.Ferstl, C.Reinbacher and H.Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *IEEE International Conference on Computer Vision*, 2013, p. 993–1000.

[5] C. D.Eigen and R.Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *In advances in neural information processing systems*, 2014, p. 2366–2374.

[6] R. J.Xie and A.Farhadi, "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks," in *In European Conference on Computer Vision*. Springer, 2016, p. 842–857.

[7] Y. X. S. B. J.Qiu, Z.Cui and M.Pollefeys, "Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image," 2018.

[8] Y.Zhang and T.Funkhouser, "Deep depth completion of a single rgb-d image," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, p. 175–185.

[9] P. K.He, G.Gkioxari and R.Girshick, "Mask r-cnn," in *IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[10] R. J.Redmon, S.Divvala and A.Farhadi, "You only look once: Unified, real-time object detection," 2015.

[11] D. C. S. C. W.Liu, D.Anguelov and A.Berg, "Ssd: Single shot multibox detector," vol. 9905, 10 2016, pp. 21–37.

[12] Q. H. S.Peng, Y.Liu and X.Zhou, "Pvnet: Pixel-wise voting network for 6dof pose estimation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4556–4565, 2018.

[13] Y. L.Shao and J.Bohg, "Clusternet: 3d instance segmentation in rgb-d images," 2018.

[14] H. J. H. Y.He, W.Sun and J.Sun, "Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation," 2019.

[15] J. X. X. J.Yu, Z.Lin and T.Huang, "Free-form image inpainting with gated convolution," 2018.

[16] C. D.Eigen and R.Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *In Advances in neural information processing systems*, 2014.

[17] V. F. I.Laina, C.Rupprecht and N.Navab, "Deeper depth prediction with fully convolutional residual networks." in *arXiv preprint arXiv:1606.00373*, 2016.

[18] M. R.Mahjourian and A.Angelova, "Geometry-based next frame prediction from monocular video," in *Intelligent Vehicles Symposium*, 2017.

[19] G. R.Garg, V.BG and I.Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *European Conference on Computer Vision*, 2016.

[20] W. L. Z.Yang, P.Wang and R.Nevatia, "Unsupervised learning of geometry with edge-aware depth-normal consistency," in *Association for the Advancement of Artificial Intelligence*, 2019.

[21] Y. W. Z.Yang, P.Wang and R.Nevatia, "Lego: Learning edge with geometry all at once by watching videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[22] C. K. H.Fu, M.Gong and D.Tao, "Deep ordinal regression network for monocular depth estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[23] C. W.Yin, Y.Liu and Y.Yan, "Enforcing geometric constraints of virtual normal for depth prediction," in *International Conference on Computer Vision*, 2019.

[24] D. J.Lee, M.Han and I.Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," 2018.

[25] D.Eigen and R.Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *IEEE International Conference on Computer Vision*, 2015.

[26] V. F. I.Laina, C.Rupprecht and N.Navab, "Deeper depth prediction with fully convolutional residual networks," in *International Conference on 3D Vision*, 2016.

[27] F.Ma and S.Karaman, "Sparse-to-dense: Depth prediction fromsparse depth samples and a single image," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 1–8.

[28] G. F.Ma and S.Karaman, "Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera," in *IEEE International Conference on Robotics and Automation*, 2019, pp. 3288–3295.

[29] Y. Y.Huang, T.Wu and W.Hsu, "Indoor depth completion with boundary consistency and self-attention," 2019.

[30] R. S. M.Bloesch, J.Czarnowski and A.Davison, "Codeslam — learning a compact, optimisable representation for dense visual slam," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[31] S. K.He, X.Zhang and J.Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[32] H. H. N. D.Xu, W.Wang and E.Ricci, "Structured attention guided convolutional neural fields for monocular depth estimation," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[33] S. B.Lee, H.Jeon and I.Kweon, "Depth completion with deep geometry and context guidance," in *The IEEE Conference on Robotics and Automation*, 2019.

[34] D. M. V. C. J.Tremblay, A.Prakash, S. T.To, E.Cameracci, and S.Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1082–10 828, 2018.

[35] P. M.Hodosh and J.Hockenmaier, "Framing image description as a ranking task: Data, models and evaluation metrics," in *Journal of Artificial Intelligence Research*, 2013, pp. 853–899.

[36] Y. Cabon, N. Murray, Humenberger, and Martin, "Virtual kitti 2," 2020.

[37] L. U. T. J.Uhrig, N.Schneider and A.Geiger, "Sparsity invariant cnns," 2017, p. 11–20.

[38] E.Coumans and Y.Bai, "Pybullet, a python module for physics simulation for games," in *robotics and machine learning*, 2016. [Online]. Available: http://pybullet.org

[39] B. O. Community, "Blender - a 3d modelling and rendering package," in *Blender Foundation, Blender Institute*, 2006.

[40] A.Harrison and P.Newman, "Image and sparse laser fusion for dense scene reconstruction," 2010, p. 219–228.

[41] R. M. D.Ferstl, C.Reinbacher and H.Bischof, "Image guided depth upsampling using anisotropic total generalized variation," 2013, p. 993–1000.

[42] F.Mal and S.Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," 2018, p. 1–8.

[43] S. S.Shivakumar, T.Nguyen and C.Taylor, "Dfusenet: Deep fusion of rgb and sparse depth information for image guided dense depth completion." 2019.

[44] P. M.Dimitrievski and W.Philips, "Learning morphological operators for depth completion." 2018, pp. 450–461.

[45] P. X.Cheng and R.Yang, "Learning depth with convolutional spatial propagation network." 2018.

[46] S. X. Z.Huang, J.Fan and H.Li, "Hms-net: Hierarchical multi-scale sparsity invariant network for sparse depth completion." 2018.

[47] M. A.Eldesokey and F.Khan, "Confidence propagation through cnns for guided sparse depth regression." 2018.

[48] E. X. M.Jaritz, R.Charette and F.Nashashibi, "Sparse and dense data with cnns: Depth completion and semantic segmentation," in *International Conference on 3D Vision*, 2018.

[49] J. G. H. Y.Xu, X.Zhu and H.sheng, "Depth completion from sparse lidar data with depth-normal constraints." 2019.