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

MDANet: Multi-Modal Deep Aggregation Network for Depth Completion

Yanjie Ke[1], Kun Li[2], Wei Yang[1*], Zhenbo Xu[1], Dayang Hao[2], Liusheng Huang[1] and Gang Wang[2]

***Abstract*— Depth completion aims to recover the dense depth map from sparse depth data and RGB image respectively. However, due to the huge difference between the multi-modal signal input, vanilla convolutional neural network and simple fusion strategy cannot extract features from sparse data and aggregate multi-modal information effectively. To tackle this problem, we design a novel network architecture that takes full advantage of multi-modal features for depth completion. An effective Pre-completion algorithm is first put forward to increase the density of the input depth map and to provide distribution priors. Moreover, to effectively fuse the image features and the depth features, we propose a multi-modal deep aggregation block that consists of multiple connection and aggregation pathways for deeper fusion. Furthermore, based on the intuition that semantic image features are beneficial for accurate contour, we introduce the deformable guided fusion layer to guide the generation of the dense depth map. The resulting architecture, called MDANet, outperforms all the state-of-the-art methods on the popular KITTI Depth Completion Benchmark, meanwhile with fewer parameters than recent methods. The code of this work will be available at https://github.com/USTC-Keyanjie/MDANet_ICRA2021.**

I. Introduction

Accurate depth perception is essential in many computer vision applications, such as autonomous driving, robot navigation and augmented realities (AR). Depth cameras can easily provide accurate depth information in indoor scenes, but perform poorly in outdoor scenes due to the interference of ambient light. Up to now, the outdoor depth perception mainly relies on four methods: lidar scanning [1], depth prediction [2], stereo matching [3] and depth completion [4]. lidar provides accurate but sparse (density only 5.9%) depth perception, as shown in Fig. 1 (Row 2). Algorithms based depth prediction or stereo matching not only depend on large computing and storage resources, yet are barely satisfactory on distant objects and untextured areas. In comparison, depth completion provides a reliable solution for dense depth map and has attracted intensive attention in recent years.

Many recent works [5]–[13] on this topic investigate to aggregate multi-modal data in an appropriate way to achieve better performance. For instance, Qiu et al. [13] predict the surface normal from image and sparse depth as the intermediate representation to guide depth completion. Chen et al. [12] design a 2D-3D fuse block to extract image and depth features by convolutional operation and continuous convolutions [11] respectively. Zhao et al. [10] exploit the graph propagation on the two modalities respectively to extract multi-modal representations. Nevertheless, most of these works construct network by stacking layers or modules linearly, and adopt shallow aggregation scheme, such as concatenation, addition or adaptive weighting. Architecture analysis [14]–[16] and representation visualization [17], [18] demonstrate that more intensive aggregation helps to deepen feature representation and extract more robust features.

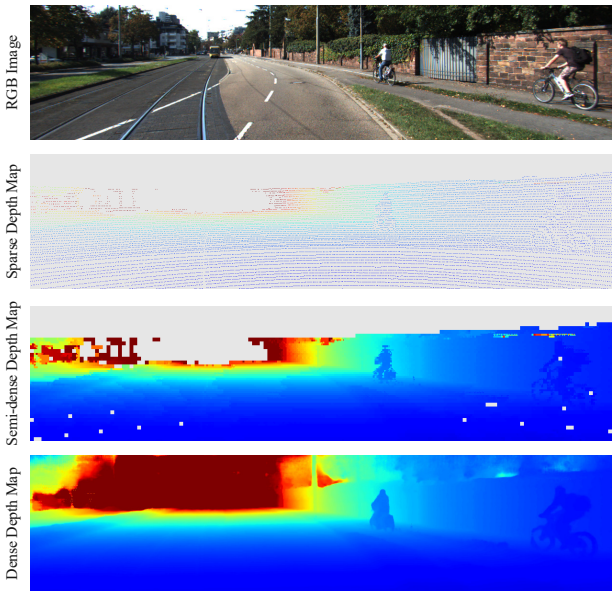

Fig. 1. **Illustration of our system**. The model takes an RGB image (Row 1) and a sparse depth map (Row 2) as input, and output a dense depth map (Row 4). Rather than directly extracting features from sparse depth map, our model infers the Semi-dense Depth Map (Row 3) which is helpful to generate accurate depth.

In order to aggregate multi-modal features effectively and improve the accuracy of depth results extremely, we propose **M**ulti-Modal **D**eep **A**ggregation Network (MDANet), which contains three stacked **M**ulti-Modal **D**eep **A**ggregation Block (MDA Block), as shown in Fig. 2. Inspired by the core concept of recent works [9], [12] about multi-scale fusion and deep aggregation, we extend the typical hourglass architecture [19] of existing methods to incorporate more semantic and spatial information. Semantic aggregation, namely aggregating across modality, helps to utilize complementarity between different modal data. Spatial aggregation, namely aggregating across scale, helps to fit various patterns across the whole image. Both of these aggregation strategies are applied in MDANet to boost the depth completion accuracy.

Furthermore, current methods [4], [20], [21] rarely take into account the inherent heterogeneity on density between

*Corresponding author: qubit@ustc.edu.cn
[1]University of Science and Technology of China
[2]Damo Academy, Alibaba Group
This work was supported by the Anhui Initiative in Quantum Information Technologies (No. AHY150300).

image and sparse depth, which prevents the vanilla convolutional neural network (CNN) based methods from extracting synchronous features. For better aggregation between multi-modal representations, we put forward the **P**re-**c**ompletion **A**lgorithm (PCA) to enhance the density of depth input significantly, as shown in Fig. 1 (Row 3). This algorithm supplements the missing value in the original sparse depth map. Compared with the raw depth map, the network can learn more detailed features from semi-dense depth maps generated by PCA.

Finally, the simple fusion strategy used in the current methods [9], [12] is another major cause of bad performance and hard optimization. We realize that the semantic information provided from image features is strongly related to the depth affinity. For more effective guidance of depth completion, we propose an effective fusion strategy, named **D**eformable **G**uided **F**usion **L**ayer (DGFL), to generate accurate dense depth map. Rather than adopting fusion by summation or concatenation, we explore the rich semantic information in image features to guide the aggregation of depth features. This strategy utilizes the advantages of multi-modal features and helps to facilitate the better performance and the smoother optimization.

In summary, the major contributions of this paper are:

- We propose MDANet composed by MDA Blocks to aggregate multi-modal data on the semantic and spatial level deeply and effectively for depth completion.
- We design PCA, which greatly increases the density, thereby helping the network understand the depth structure across the whole scene while retaining the original and accurate depth information.
- DFGL is inventively introduced to predict dense depth output accurately under the geometric guidance from image features.
- We compare our MDANet with state-of-the-art methods on KITTI Depth Completion Benchmark [22] in IV-D and achieve superior results. Furthermore, we conduct detailed ablation study to demonstrate the effectiveness and generalization performance of PCA and DFGL in IV-E.

II. Related Work

A. Depth Completion

The current depth completion methods are mainly divided into two categories. The first category of methods focuses on learning the image-dependent affinity through a deep network to refine the blur depth map iteratively. Recently, Cheng et al. [20] learn the affinity more efficiently through CNN, and extend it to the field of depth estimation. They also developed CSPN++ [21] to learn stronger representation and more resource-friendly structure yielding significant performance boost compared to CSPN. Park et al. [4] introduce the non-local information that allow CSPN to be no longer limited to a fixed structure but adaptively find relevant neighbours.

Another category of methods investigate to integrates multi-modal data by exploiting the complementarity between them or discovering more instructive information in the image. Zhang et al. [23] propose to perform depth completion in indoor scenes under the guidance of surface normal the occlusion boundary of the object. Qiu et al. [13] extend to adapt the complex situation in outdoor scenes. Huang et al. [6] develop the sparse invariance operation to efficiently process sparse data. Eldesokey et al. [24] introduce the confidence propagation with image information for depth completion. Hui et al. [25] exploit multi-scale features to obtain the higher resolution of the depth map. Similarly, Li et al. [9] design a cascaded hourglass network to improve the quality of dense depth maps gradually. Following these works, we realize that the key of fusion is how to dig out meaningful features from sparse data.

B. Architecture for Feature Aggregation

In order to solve more complex computer vision tasks, many networks are designed with more parameters, more complex structures, and deeper levels to improve the capacity and fitting ability of the model. However, since improving the width and depth of the network structure for better performance has reached a certain bottleneck, more and more methods are turning to focus on more effective feature aggregation for richer information and deeper feature representation. He et al. [26] creatively introduce short-cut connection and bottlenecks structure in linear convolutional neural network. Lin et al. [27] demonstrate that channel mixing is a general and effective extension for feature aggregation. Huang et al. [28] propose to concatenate all the layers in stage by skip connections for better feature propagation. Lin et al. [29] design to enhance feature connection and learn multi-scale information by connect features across levels of the pyramid. Yu et al. [14] take advantage of non-linear feature aggregation and reuse strategy to strength semantic feature and spatial information. Motivated by these works, we are committed to exploring how and where to aggregate multi-modal features effectively for better performance.

C. Multi-modal Information Fusion

Multi-modal information fusion is essential to many computer vision tasks. Many previous works [30]–[32] apply attention mechanism to fuse cross-view spatial feature for 3D object detection and reading comprehension and question answering respectively. As for depth completion, current work rarely considers effective fusion strategies. Zhao et al. [10] propose Symmetric Gated Fusion strategy to adaptively select effective information by gated weight. Following guided image filtering, Tang et al. [7] investigate to apply the spatially variant convolution kernels to extract features from sparse depth. Inspired by the concept of adaptive geometry-aware method, the DFGL we proposed empowers depth features to actively search for relevant information and avoid harmful guidance. Specifically, we apply a deformable convolutional operation on guiding dense depth map with spatial variant offset according to image features.

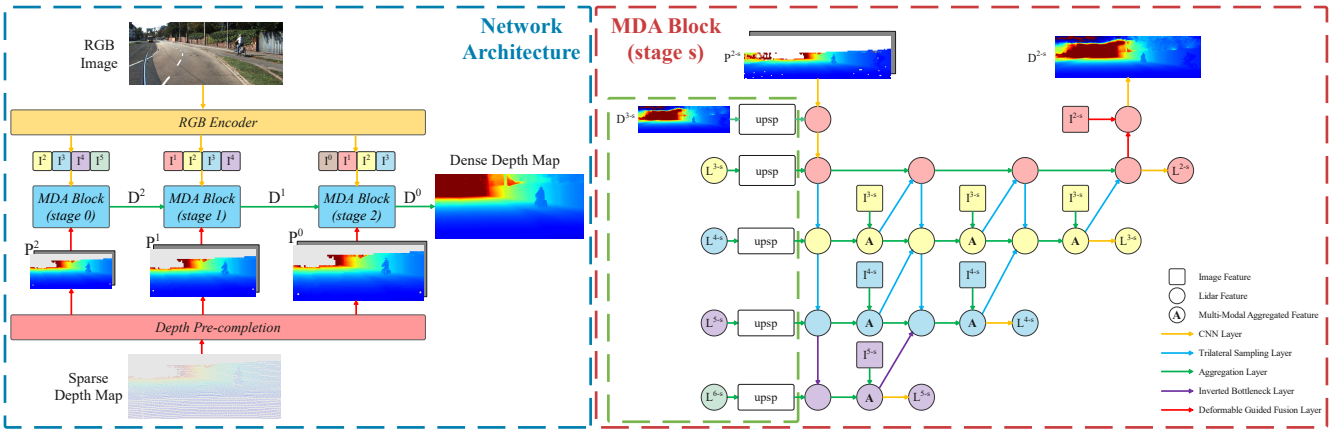

Fig. 2. **Illustration of the proposed network.** Left part: The overview of our network architecture; Right part: The detail of s-th stage MDA Block. Our network can be seen as consisting of three parts: RGB Encoder, Depth Pre-completion and Multi-Modal Deep Aggregation. In RGB Encoder, we employ a series of down-sample operations to extract image features, denoted as I^k. In Depth Pre-completion, we generate the semi-dense depth input, denoted as P^k, via pre-completion algorithm (III-B). In Multi-Modal Deep Aggregation, we stack three MDA Blocks to receive image features and semi-dense depth map with the corresponding resolution. Except for the stage 0, MDA Block also absorbs the information, denoted as L^k, from the previous stage, as shown in the green dotted box. Then each down-sampled depth feature will be aggregated with L^k, and each up-sampled depth feature will be aggregated with the image feature (III-C). Finally, dense depth output D^k is calculated via Deformable Guided Aggregated Layer (III-E). The superscript k means the tensor's scale is $1/2^k$ of the input RGB image, and features signed by same color have the same scale.

III. The Proposed Method

A. Overview

We can easily obtain a depth map projected from the lidar point cloud to the image plane. However, it is so sparse to distinguish the specific information of the observed object. Our goal is to reconstruct a dense depth map with 100% density under the guidance of the image. We present the MDANet to tackle this problem, as illustrated in Fig. 2. We feed the quarter-sized Pre-completion map P^2, half-sized P^1 and full-sized P^0 respectively into three MDA Blocks to obtain the depth features. The RGB image is encoded into image features of different scales and aggregated with depth features accordingly. At the end of each MDA Block, the DGFL will further merge image and depth features to generate a dense depth map output.

B. Pre-completion Algorithm (PCA)

Inspired by the down-sampling operation in MSG-CHN [9], we put forward PCA to increase the density of sparse input. The down-sampling operation proposed in MSG-CHN [9] can be described as:

$$C(x,y) = \begin{cases} 0, & \text{if } sD(x,y) = 0 \\ 1, & \text{otherwise} \end{cases},$$

$$sD^k(x,y) = \frac{\sum_{i,j}^{2^k-1} sD(2^k x + i, 2^k y + j)}{\sum_{i,j}^{2^k-1} C(2^k x + i, 2^k y + j) + \epsilon},$$

where sD is the sparse depth map and $sD^k(x,y)$ is the depth value at position (x,y) on the sD, in which k denotes that the down-sampling factor is 2^k, and $k = 0,1,2,3$; ϵ is a small number to avoid division by zero.

Our PCA extends it further. Specially, for the missing depth value at position (x,y) on sD^k, we fill it with the average of observed depth in the window of pooling. Mathematically, PCA can be expressed as:

$$C^k(x,y) = \begin{cases} 0, & \text{if } sD^k(x,y) = 0 \\ 1, & \text{otherwise} \end{cases} \quad k \in \{0,1,2,3\},$$

$$mD^k = \begin{cases} sD^3, & \text{if } k = 3 \\ sD^k + (\mathbf{1} - C^k) m D^{k+1}_{\times 2}, & \text{if } k \in \{0,1,2\} \end{cases},$$

where mD is the semi-dense depth map whose density is increased from the 5.9% to 60.7%; $\mathbf{1}$ is a matrix full of 1 with corresponding resolution and $_{\times 2}$ means interpolate operation by nearby value.

Although we have obtained a semi-dense depth map, error interference is introduced as the same time: the network will pay equal attention to the observed accurate depth and the complemented estimated depth. Therefore, we further introduce a confidence map that can adaptively encode the credibility of depth value. We sign the confidence of the observed depth points with 1, and the complemented points according to the number, distance and the confidence of reference points. Mathematically, it can be expressed as:

$$W^3(x,y) = \begin{cases} 0, & \text{if } sD^3(x,y) = 0 \\ 1, & \text{otherwise} \end{cases},$$

$$A^k(x,y) = \frac{1}{4} \sum_{i,j}^{3} C^{k-1}(2x + i, 2y + j) \quad k \in \{1,2,3\},$$

$$\begin{aligned} W^k = {} & C^k + C^{k+1}_{\times 2} (\mathbf{1} - C^k) A^{k+1}_{\times 2} \\ & + \lambda (\mathbf{1} - C^{k+1}_{\times 2}) W^{k+1}_{\times 2} \qquad k \in \{0,1,2\}, \end{aligned}$$

where A is the average value of C in the 2×2 windows size; W is the confidence map; λ is a hyperparameter to control the ratio of weight transmitted and assigned with 0.25 for the best performance. At last, we can achieve pre-completion output P^k by concatenating the mD^k with W^k. We will further demonstrate the effectiveness of this algorithm in the IV-E.

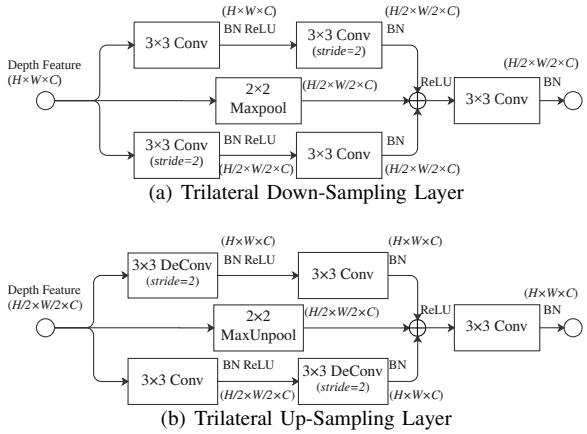

Fig. 3. **Illustration of TSL.** This Layer has three branches with different manners to down-sample (a) or up-sample (b) the feature representation. Then the feature response of three branches is element-wise added as the output. Notations: *Conv* is convolutional operation; *DeConv* is transposed convolutional operation; *BN* is the batch normalization; *ReLU* is the ReLU activation function; *Maxpool* is the max pooling; *MaxUnpool* is the max unpooling; + means element-wise addition; 2×2, 3×3 denote the kernel size and *stride* denotes the stride in convolutional operation; H×W×C means the tensor shape (height, width, depth).

C. Multi-Modal Deep Aggregation Block (MDA Block)

Recent works [7], [10] show that direct aggregation brings about misleading guidance and disturbing noise due to the inherent heterogeneity between image and depth information. To address this problem, we propose the MDA Block to aggregate image and depth feature deeply. Rather than aggregating depth and image representation at the same time, we analyze their respective advantages and aggregate with different information under different circumstances. On the one hand, considering the depth representation's spatial information can guide the network to weigh the value of each element, we aggregate with the depth features before the down-sampling operation. On the other hand, taking advantage of the image representation's semantic guidance that promotes the network to learn detailed information, we aggregate with image features before up-sampling operation. In this way, shallow features are refined as they are propagated through different aggregation strategies.

D. Trilateral Sampling Layer (TSL)

We put forward the TSL served in MDA Block to fully extract multi-scale and multi-level information. This layer has two forms: trilateral down-sampling layer and trilateral up-sampling layer, as illustrated in Fig. 3, which are responsible for contracting and expanding feature representation respectively in each stage. Inspired from ERFNet [33] and BiSeNet [34], we design the trilateral down-sampling layer with three down-sampling branches and a fusion block. With different down-sampling manners, we can obtain three half-resolution feature maps, which encodes the multi-scale information. Then we employ a summation operation followed by a 3×3 convolution to fuse these feature maps. In contrast to the downsampler block in ERFNet [33], we replace the concatenation operation in the last fusion with addition. Considering the homogeneity of feature response from three sampling branches, addition operation does not lead to worse accuracy but achieves less computation cost and memory access cost.

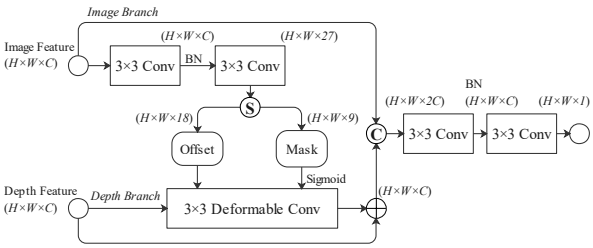

Fig. 4. **Illustration of DFGL.** This Layer consists of Image Branch and Depth Branch. In Image Branch, we take the image feature as the input to generate the offset and mask by convolutional operation, and then feed them with depth feature into the deformable convolution in Depth Branch. A skip connection is designed to forward depth feature directly. Finally both feature response of two branches is concatenated as the output. Notations: *Conv* is convolutional operation; *Deformable Conv* is deformable convolutional operation; *BN* is the batch normalization; *Sigmoid* is the Sigmoid activation function; *ReLU* is the ReLU activation function; + means element-wise addition; **C** means channel-wise concatenation; **S** means splitting along channel axis; 3×3 denote the kernel size; H×W×C means the tensor shape (height, width, depth).

The trilateral up-sampling layer is designed in a symmetrical way. Following SegNet [35], we introduce the Max-Unpooling operation to our up-sampling Module, which is an effective up-sampling manner to recover spatial information. Specially, we adopt the max pooling indices to upsample the feature map and fill the remaining points with 0.

E. Deformable Guided Aggregated Layer (DGFL)

Taking advantage of deformable convolution network [36], [37], we design the DGFL to generate the accurate dense depth map under the geometric guidance of image feature, as illustrated in Fig. 4. This layer consists of image branch and depth branch. In image branch, we adopt the 3×3 convolution to shrink the channels of feature to 27. The data in the first 18 channels and next 9 channels is adopted as offset and weight respectively. Then we feed them with depth features into deformable convolution in depth branch. We also design a skip connection to allow the depth feature to forward directly. Finally, both feature response of two branches is concatenated as the output.

F. Loss Function

During training, we adopt a weighted sum of mean squared error (MSE) and mean absolute error (MAE) to compute the loss between ground truth and predicted depth.

$$\mathcal{L}^i = \sum_{p \in \mathbf{V}} \left(\left\| \mathbf{D}_p^i - \widehat{\mathbf{D}}_p^i \right\|_2^2 + \gamma \left\| \mathbf{D}_p^i - \widehat{\mathbf{D}}_p^i \right\|_1 \right),$$

$$L = \delta \mathcal{L}^2 + \delta \mathcal{L}^1 + \mathcal{L}^0,$$

where $\mathbf{V}$ represents the set of valid pixels. $\mathbf{D}_p^i$ and $\widehat{\mathbf{D}}_p^i$ denote the predicted depth and ground truth at the pixel p in $1/2^i$ resolution respectively. We set $\gamma = 1$ at the first 40 epochs and 0 at the rest. For δ, we set it as 1 at the first 6 epochs, then reduce it to 0.1. At last, we set it as 0 after 11 epochs.

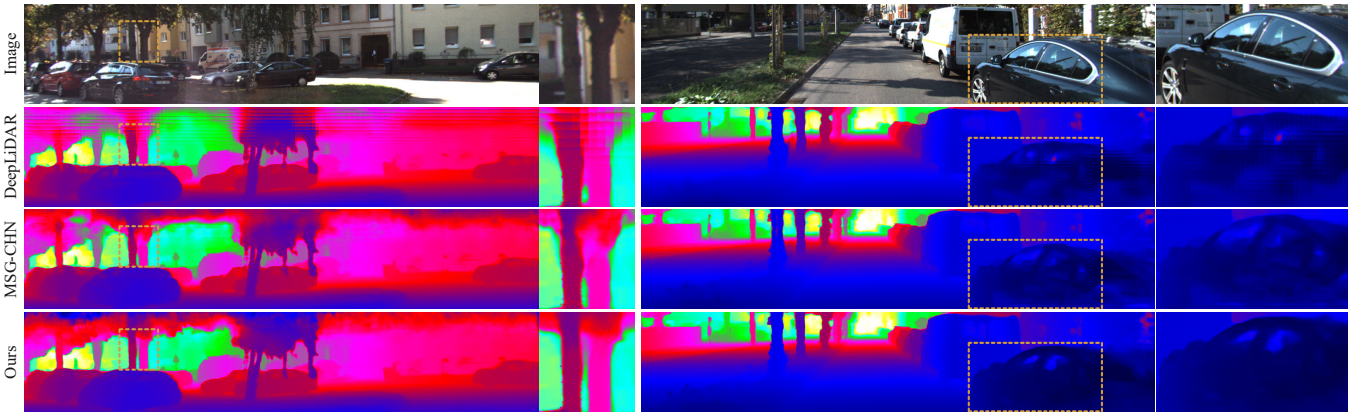

Fig. 5. **Qualitative comparison results on KITTI Depth Completion test set.** The depth images are colorized according to the depth value. Our results are shown in the bottom row and compared with state-of-the-art methods "DeepLiDAR" [13] and "MSG-CHN" [9]. In the yellow dotted regions, our method recovers better 3D details.

IV. Experiments

A. Dataset

KITTI Depth Completion Benchmark [22] provides sparse depth maps captured by projecting the lidar point clouds into the camera field of view, the camera images obtained at the same time and corresponding ground truth that obtained by the depth accumulation of continuous frames and stereo matching. It also provides a leader-board to compare the pros and cons of different methods. The KITTI dataset contains 85898 frames for training, 1000 frames for evaluation and 1000 frames for testing without ground truth.

B. Evaluation Metrics

Following the KITTI Depth Completion Benchmark and current depth completion methods [5]–[13], we report four common metrics: the mean absolute error (MAE, mm), root mean squared error (RMSE, mm), mean absolute error of the inverse depth (iMAE, 1/km) and root mean squared error of the inverse depth (iRMSE, 1/km).

C. Implementation Details

Our models are implemented in PyTorch [38] 1.5 and trained on eight NVIDIA Tesla P100 with the CUDA 10.1 and CUDNN 7.0. Adam algorithm [39] ($\beta 1 = 0.9$, $\beta 2 = 0.999$ and weight_decay $= 0.0002$) is employed to optimize the parameters of our models. We train the network for 50 epochs from scratch with the "kaiming normal" [40] initialization and batch size 2 per GPU card. The learning rate is initialized to 0.001 and multiplied by 0.5 every 5 epochs. For data augmentation, we flip the image and depth map left and right with a probability of 0.5. Random cropping and center cropping are applied during training and inference respectively to adjust the resolution of data to 1216×352.

D. Result Analysis

We compare MDANet with other top-ranking published methods on the KITTI leaderboard. As shown in Table I, our method achieves the best performance under the primary RMSE metric. Meanwhile, note that our method surpasses all depth completion algorithms with only a small amount of parameters. Taking CSPN++ [21] and NLSPN [4] as examples, our method only uses about one-eighth of their parameters but shows more advanced results.

Fig. 5 shows some visual comparison results with several state-of-the-art methods on the KITTI's test set. Our results are shown in the last row. It shows that our estimated depth maps reveal more accurate depth around close as well as far away objects. For example, as highlighted by the yellow dotted box, our method can better recover the depth of background between the trees and the most accurate contour in the black car region. Obviously, the semantic information of RGB image provides the effective geometric guidance at the edge of objects and helps to generate accurate depth completion in our model.

TABLE I

Quantitative Results on KITTI Test Set. The comparison result is provided by KITTI Testing server and ranked by RMSE.

Methods	Params (M)	**RMSE (mm)**	MAE (mm)	iRMSE (1/km)	iMAE (1/km)
DDP [41]	18.8	832.94	203.96	2.10	0.85
NConv-CNN [42]	0.36	829.98	233.26	2.60	1.03
Revisiting [43]	-	792.80	225.81	2.42	0.99
PwP [5]	28.99	777.05	235.17	2.42	1.13
FusionNet [8]	2.55	772.87	215.02	2.19	0.93
MSG-CHN [9]	1.2	762.19	220.41	2.30	0.98
DeepLiDAR [13]	53.44	758.38	226.50	2.56	1.15
FuseNet [12]	1.89	752.88	221.19	2.34	1.14
CSPN++ [21]	∼26	743.69	209.28	2.07	0.90
NLSPN [4]	25.84	741.68	**199.59**	**1.99**	**0.84**
MDANet (Ours)	3.07	**738.23**	214.99	2.12	0.99

TABLE II

Ablations of Pre-completion Strategy on KITTI's Validation Set. We validate the effectiveness of PCA.

	λ	**RMSE**	MAE	iRMSE	iMAE
w/o PCA	-	766.58	214.55	2.19	0.98
with PCA	0.5	771.84	215.72	2.21	1.02
	0.25	**764.91**	**213.96**	**2.17**	**0.97**
	0.125	766.20	214.21	2.18	1.00
	0	774.47	216.37	2.23	1.08

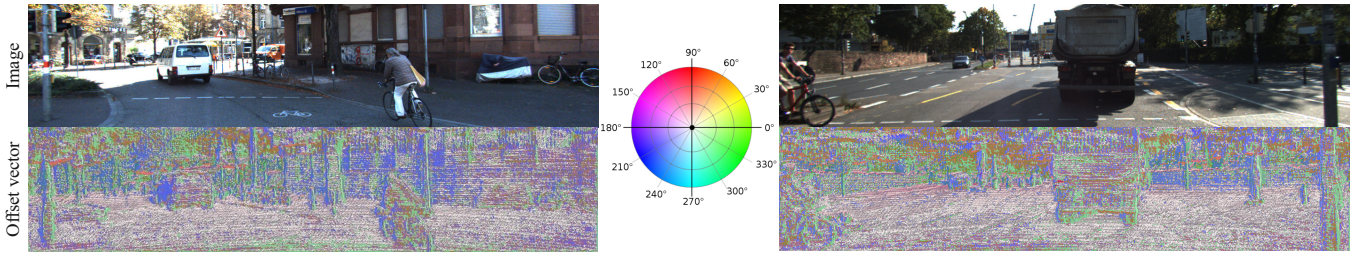

Fig. 6. **Visualization of the offset vector in DFGL**. For each pixel in the image, we only visualize the most weighted offset vector by the colored polar coordinate system.

TABLE III

ABLATIONS OF FUSION STRATEGIES ON KITTI'S VALIDATION SET. WE VALIDATE THE EFFECTIVENESS OF DFGL.

	RMSE	MAE	iRMSE	iMAE
Add.	779.47	224.72	2.35	1.05
Concat.	773.78	221.25	2.21	1.03
Deform.	**764.91**	**213.96**	**2.17**	**0.97**
Gated.	775.70	217.61	2.31	1.08
Gated. + Deform.	776.46	219.83	2.39	1.13

TABLE IV

VERIFICATIONS OF GENERALIZATION PERFORMANCE ON KITTI'S VALIDATION SET. WE SET $\lambda = 0.25$ IN PCA.

Method	PCA	DGFL	**RMSE**	MAE	Runtime (s)
FusionNet [8]			778.41	215.07	0.023
	✓		**775.68**	**214.86**	0.024
MSG-CHN [9]			825.14	226.91	0.010
	✓		821.17	226.40	0.011
		✓	812.32	217.14	0.014
	✓	✓	**809.79**	**216.48**	0.015

E. Ablation Study

1) Effectiveness of PCA: In order to assist the network in learning representation deeply, we fill in the missing values in the depth map by PCA. Here, we compare with a model trained end-to-end with or without PCA, and further compare the performance of selecting different values for λ , as shown in Table II.

From the table, we can see that extracting features from semi-dense depth maps generated by PCA helps improve the performance of the network. This result demonstrates that richer depth information indeed lead to better aggregation between depth and image features. We can also find that the network achieves the best performance when $\lambda = 0.25$, which is adopted by our best model. This is reasonable because the credibility of the depth value estimated by nearby depth must be greater than the depth value estimated by distant. Moreover, 0.25 is a suitable parameter for λ that can properly characterize the confidence across the whole semi-dense depth map.

2) Effectiveness of DFGL: To verify that the effectiveness of the proposed DFGL, we replace this module in our network by feature addition or concatenation but keep the other network components and settings unchanged. The results are indicated as "Add." and "Concat." respectively. Our deformable guided fusion strategy is denoted as "Deform.". We have further designed a gated fusion strategy, termed as "Gated.". Specifically, this strategy refers to multiplying the image features by a gated weight map, which is generated by the depth feature and normalized from 0 to 1. "Gated. + Deform." means to adopt gated fusion strategy and deformable guided fusion strategy at the same time.

As the results shown in Table III, we find that addition and concatenation operations lead to poor performance, because the network difficultly handle the heterogeneity between multi-modal information following these fusion strategy. Interestingly, the adoption of deformable guided fusion strategy alone achieves remarkable improvement. However, the introduction of gated fusion strategy brings about lower training loss but bad performance on testing. We have reason to believe that gated fusion strategy exacerbates the "overfitting" which leads to worse generalization performance.

Furthermore, we visualize the offset vector with the largest weight of each pixel by the coloured polar coordinate system, as shown in Fig. 6. Slight guidance is provided in the wide area such as the road or the wall. In contrast, at the edge of objects, obvious geometric orientation is generated by image features to guide depth features to absorb the effective information and avoid harmful interference.

3) Generalization Performance of Proposed Components: We apply the PCA and DFGL on the other methods to verify the generalization performance of our components. The comparison results are listed in Table IV. Specially, for convincing qualitative comparison, we replaced the sparse depth input of FusionNet [8] with semi-dense depth input generated by PCA with $\lambda = 0.25$. We replace the input data of MSG-CHN [9] by the same way, and replace its output module with DGFL. The results demonstrate that PCA and DFGL can be successfully applied to other methods to improve performance only with little additional time cost.

V. CONCLUSIONS

In this paper, we proposed MDANet to recover a dense depth map from sparse depth and RGB image. The proposed MDA Block in our novel network architecture helps to achieve promising performance by aggregation between multi-modal representations. Each block receives the semi-dense depth map generated by the PCA at different resolutions, and provides an accurate dense depth map via the DFGL. Extensive experiments demonstrate the superior performance of our MDANet, and ablation study verifies the effectiveness of the PCA as well as the DFGL. Although this paper mainly focuses on depth completion task, we believe that our method is universally applicable to other computer vision tasks with multi-modal signal input.

REFERENCES

[1] C. Weitkamp, *Lidar: range-resolved optical remote sensing of the atmosphere*. Springer Science & Business, 2006, vol. 102.

[2] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2002–2011.

[3] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, "Ga-net: Guided aggregation net for end-to-end stereo matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 185–194.

[4] J. Park, K. Joo, Z. Hu, C.-K. Liu, and I. S. Kweon, "Non-local spatial propagation network for depth completion," *arXiv preprint arXiv:2007.10042*, 2020.

[5] Y. Xu, X. Zhu, J. Shi, G. Zhang, H. Bao, and H. Li, "Depth completion from sparse lidar data with depth-normal constraints," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2811–2820.

[6] Z. Huang, J. Fan, S. Cheng, S. Yi, X. Wang, and H. Li, "Hms-net: Hierarchical multi-scale sparsity-invariant network for sparse depth completion," *IEEE Transactions on Image Processing*, vol. 29, pp. 3429–3441, 2019.

[7] J. Tang, F.-P. Tian, W. Feng, J. Li, and P. Tan, "Learning guided convolutional network for depth completion," *arXiv preprint arXiv:1908.01238*, 2019.

[8] W. Van Gansbeke, D. Neven, B. De Brabandere, and L. Van Gool, "Sparse and noisy lidar completion with rgb guidance and uncertainty," in *2019 16th International Conference on Machine Vision Applications (MVA)*. IEEE, 2019, pp. 1–6.

[9] A. Li, Z. Yuan, Y. Ling, W. Chi, C. Zhang *et al.*, "A multi-scale guided cascade hourglass network for depth completion," in *The IEEE Winter Conference on Applications of Computer Vision*, 2020, pp. 32–40.

[10] S. Zhao, M. Gong, H. Fu, and D. Tao, "Adaptive context-aware multi-modal network for depth completion," *arXiv preprint arXiv:2008.10833*, 2020.

[11] S. Wang, S. Suo, W.-C. Ma, A. Pokrovsky, and R. Urtasun, "Deep parametric continuous convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2589–2597.

[12] Y. Chen, B. Yang, M. Liang, and R. Urtasun, "Learning joint 2d-3d representations for depth completion," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 10 023–10 032.

[13] J. Qiu, Z. Cui, Y. Zhang, X. Zhang, S. Liu, B. Zeng, and M. Pollefeys, "Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3313–3322.

[14] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2403–2412.

[15] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[16] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "Densenet: Implementing efficient convnet descriptor pyramids," *arXiv preprint arXiv:1404.1869*, 2014.

[17] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.

[18] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 472–480.

[19] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European conference on computer vision*. Springer, 2016, pp. 483–499.

[20] X. Cheng, P. Wang, and R. Yang, "Learning depth with convolutional spatial propagation network," *arXiv preprint arXiv:1810.02695*, 2018.

[21] X. Cheng, P. Wang, C. Guan, and R. Yang, "Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion." in *AAAI*, 2020, pp. 10 615–10 622.

[22] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

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

[24] A. Eldesokey, M. Felsberg, and F. S. Khan, "Propagating confidences through cnns for sparse data regression," *arXiv preprint arXiv:1805.11913*, 2018.

[25] T.-W. Hui, C. C. Loy, and X. Tang, "Depth map super-resolution by deep multi-scale guidance," in *European conference on computer vision*. Springer, 2016, pp. 353–369.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[27] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

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

[30] J. H. Yoo, Y. Kim, J. S. Kim, and J. W. Choi, "3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection," *arXiv preprint arXiv:2004.12636*, 2020.

[31] C. Hori, T. Hori, G. Wichern, J. Wang, T.-y. Lee, A. Cherian, and T. K. Marks, "Multimodal attention for fusion of audio and spatiotemporal features for video description." in *CVPR Workshops*, 2018, pp. 2528–2531.

[32] W. Wang, M. Yan, and C. Wu, "Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering," *arXiv preprint arXiv:1811.11934*, 2018.

[33] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 1, pp. 1–10, 2017.

[34] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation," 2020.

[35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[36] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 764–773.

[37] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9308–9316.

[38] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in neural information processing systems*, 2019, pp. 8026–8037.

[39] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[41] Y. Yang, A. Wong, and S. Soatto, "Dense depth posterior (ddp) from single image and sparse range," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3353–3362.

[42] A. Eldesokey, M. Felsberg, and F. S. Khan, "Confidence propagation through cnns for guided sparse depth regression," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[43] L. Yan, K. Liu, and E. Belyaev, "Revisiting sparsity invariant convolution: A network for image guided depth completion," *IEEE Access*, vol. 8, pp. 126 323–126 332, 2020.