2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Depth Completion using Galerkin Attention

Yinuo Xu* and Xuesong Zhang*

Abstract— **Current depth completion methods usually employ a pair of calibrated RGB and depth sensors to reconstruct a dense depth map. Although RGB (dense) and depth (sparse) measurements are collected from the same underlying scene, they reflect different physical characteristics and thus it remains rather intricate how the devised RGB guidance scheme can effectively leads to a faithful depth recovery. Different from existing 3D geometry representations, such as point cloud, voxels or meshes, we propose to define 3D scenes as vector-valued functions, $f : \Omega \ni (u, v) \mapsto (r, g, b, d) \in \mathbb{R}^4$, mapping from the image plane Ω to RGBD vectors. This scene function representation brings two benefits: 1) allowing for the adaptation of the Galerkin method to explore the nodal basis of the scene function space, and 2) transforming the irregularly scattered (X,Y,Z) points in the Euclidean space into the depth function defined over the regular grid in the image plane. We further leverage these two benefits within a deep neural network, characterized by an efficient Galerkin attention-based RGBD function embedding to effectively explore the interaction of color and depth information, and by the utilization of equivariant convolution operation on the RGBD feature map as efficient basic blocks. Experiments show that the proposed method achieves significant performance improvement over state-of-the-arts. Code at https://github.com/ZXS-Labs/DCGA.**

I. INTRODUCTION

The current methods for guided depth image completion with color images typically involve using a pair of calibrated RGB and depth sensors [1] to reconstruct a dense depth map[2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. Since RGB and depth measurements are collected from the same scene, a straightforward approach for RGB-guided depth completion is to transform them into a point cloud representation using camera intrinsic and extrinsic parameters[15]. But directly converting RGBD measurements into point cloud results in sparse and irregular data distribution, making it difficult to use the efficient CNN blocks for feature extraction. To regularize point cloud data distribution, there are several methods available, such as voxelization[4] by dividing the point cloud into equispaced voxel blocks, or meshing[16] by decomposing the point cloud into roughly regular facets, which are then processed by CNN modules. In terms of the joint RGB-D feature extraction, CNNs are confined to local feature aggregation while incapable to attend global features. Furthermore, lack of correlation aware interaction between the two modalities gives rise to somewhat blindness in exploiting RGB information to guide depth feature completion.

Attention mechanisms[17] have shown appealing performance in computer vision[18], [19] and have been introduced into depth completion[20], [21]. They can effectively capture global information and compensate for the limitations of CNNs. Nevertheless, attention mechanisms are computationally expensive when directly applied to images, usually demanding down-sampling operations prior to the attention operations. However, operating on low resolution feature maps inevitably leads to a loss of local detail information, which is a dilemma confronted by the computer vision researchers. Inspired by the recent works[22], [16] employing the Galerkin method for continuous signal approximation, we introduce the Galerkin attention mechanism into the RGBD fusion and embedding process, which achieves the same effect as the attention mechanism[23] while significantly reduces computational complexity. This allows us to perform attention computations at the original grid resolution, capturing both local and global information.

Both the RGB and the depth images of the underlying scene reflect the shapes of objects therein projected onto the image plane, but RGB measurements are rich in textures while depth maps not. Therefore, in the fusion and interaction process of the RGBD information, we should consider how to guide and aggregate them in a principled way. Previous works [4], [21] put the focus on decoding the low-resolution embeddings into the high-resolution depth map, usually progressively in scales. However, in the entire depth completion task, effectively embedding the RGBD information at high resolution plays an important role. Embedding the raw measurements into a high-dimensional latent space involves guiding and fusing the two branches of features in a way that benefits the subsequent decoding process. In viewing of this, we propose an RGBD function embedding scheme. We define the scene as a vector-valued function, $f : \Omega \ni (u, v) \mapsto (r, g, b, d) \in \mathbb{R}^4$, mapping from the image plane Ω to RGBD vectors. We can regard the depth map as a set of samples from this continuous vector-valued function in the depth dimension. Our goal is to obtain a dense set of samples, which can be achieved by learning the mapping from an approximation space determined by sparse samples to one determined by dense samples. Inspired by the recent works on neural operator [22], [24], we search for a set of basis functions that can be used to represent any continuous RGBD function. By learning these basis functions from the available RGB and depth information, the denser set of depth values are nothing but the evaluations of the RGBD function on the regular image grid.

*Equal contributions. This work was supported by the National Natural Science Foundation of China under Grant 61871055. (Corresponding author: Xuesong Zhang).

Y. Xu and X. Zhang are with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing, 100876, China (e-mail: map0420@bupt.edu.cn; xuesong_zhang@bupt.edu.cn).

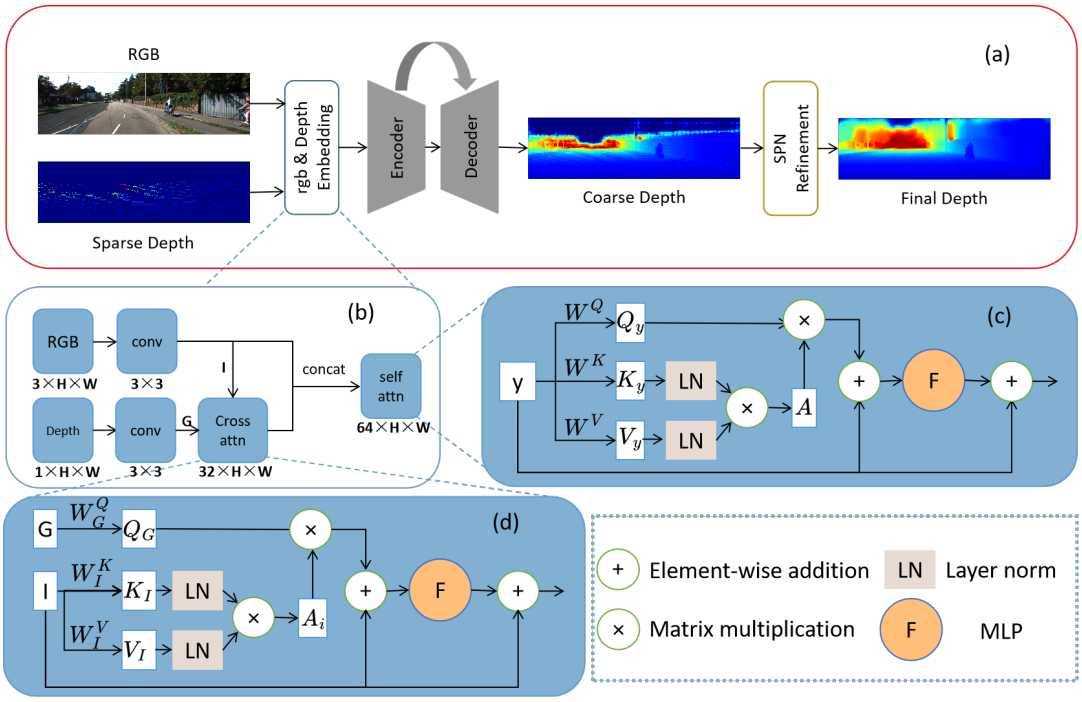

Fig. 1. Overview of our method that augments the backbone network[21] with a high-resolution RGBD function embedding module. (a) Pipeline of the proposed depth completion architecture. (b) RGBD function embedding module. (c) Self attention of the fused RGBD features. (d) Cross attention leveraging color semantic and contextual information to guide depth aggregation.

Our main contributions are summarized as follows:

- We propose an RGBD function embedding module, which effectively extracts, guides, and fuses the high-resolution color features and sparse depth features.
- To tackle high computational complexity in embedding the raw RGBD information, we adopt the Galerkin attention method to reduce the attention calculation complexity.
- Through experimental validation, we demonstrate that significant improvements can be achieved on the KITTI and NYUv2 datasets only by using an RGBD function embedding module at the initial stage of the depth completion network.

II. METHODOLOGY

A. Problem Formulation

By the calibration of the camera and depth sensor, we can define the scene as a vector valued function $f : \Omega \ni (u, v) \mapsto (r, g, b, d) \in \mathbb{R}^4$, where $\Omega \in \mathbb{R}^2$ is the image plane of the camera. Assuming f belongs to a Hilbert space $\mathcal{H}$, we further define two discretized versions of f to reflect the discretization discrepancy between the RGB and depth channels,

$$\begin{aligned} A_{h_f} \ni f_{h_f} &: \Omega_{h_f} \times \Omega_{h_f} \to \mathbb{R}^3 \times \mathbb{R}, \\ A_{h_c} \ni f_{h_c} &: \Omega_{h_f} \times \Omega_{h_c} \to \mathbb{R}^3 \times \mathbb{R}, \Omega_{h_c} \subset \Omega_{h_f} \end{aligned} \tag{1}$$

where Ω_{h_f} and Ω_{h_c} are discretized by the nodes $\{x_i\}_{i=1}^{n_{h_f}}$ and $\{\widetilde{x_i}\}_{i=1}^{n_{h_c}}$ with the mesh size h_f and h_c, respectively, and $A_{h_f}, A_{h_c} \subset \mathcal{H}$ are the approximation spaces associated with the corresponding nodes.

To solve the depth completion problem, we learn an operator parameterized by θ, $C_\theta{:}A_{h_c}{\to}D_{h_f}$, $f_{h_c}{\mapsto}d_{h_f}{:}\Omega_{h_f}{\to}R$, which maps the incomplete RGBD measurements to a dense depth map. Given N training pairs $\{f_{h_f}, f_{h_c}\}_{k=1}^{N}$, the network parameter θ can be solved through the empirical risk minimization:

$$\min_\theta \frac{1}{N} \sum_{k=1}^{N} \|\Pi_d f_{h_f}^{(k)} - \mathcal{C}_\theta(f_{h_c}^{(k)})\|_{\mathcal{H}} \tag{2}$$

where $\Pi_d(\cdot)$ represents the extraction of the depth channel. Note that the discretization intervals vary among the training samples, so we hope the performance of C_θ is robust to discretization, which is exactly pursued by the mesh-free PDE solutions [25], [22] as well as arbitrary scale super-resolution [23]. In Fig.1, the network architecture C_θ is composed of three parts: i) the RGBD embedding module that will be discussed in Sec.II B and C, ii) an encoder-decoder backbone [21] that layer-wisely updates the latent nodal bases supported on $\{x_i\}_{i=1}^{n_{h_f}}$, and iii) an SPN refinement component [6].

B. Galerkin Attention

The attention mechanism [17] has been manifested in various problems as an effective approach to both global aggregation (via self attention) and to multimodality fusion (via cross attention). However, the quadratic computational complexity of the attention map hinders its application to long sequences or high-resolution images. Now we introduce the Galerkin type attention [23] to efficiently update the basis functions of the latent representation space, through the lens

of Galerkin method in Finite Element Analysis [26]. Given the input latent representation $y \in \mathbb{R}^{n\times c}$, with n the node numbers and c the feature length, the Galerkin attention is softmax free with linear complexity (see Tab. I for a comparison),

$$\mathbf{z} = Attn_g(\mathbf{y}) := Q(\widetilde{K}^T\widetilde{V})/n \tag{3}$$

$$\widetilde{\mathbf{z}} = \mathbf{y} + F(\mathbf{z} + \mathbf{y}) \tag{4}$$

where $\widetilde{K}$=$Ln(K)$, $\widetilde{V}$=$Ln(V)$, with $Ln(\cdot)$ the layer normalization, and the columns of $Q/K/V = \mathbf{y}W^{Q/K/V}, W^{Q/K/V} \in \mathbb{R}^{c\times c}$, are the latent basis functions evaluated on $\{x_i\}_{x_i=1}^{n}$, spanning three subspaces of the latent space. The columns of $\widetilde{K}^T\widetilde{V}$ contains the coefficients for the linear combination of the basis vectors in Q to form the output $\mathbf{z}$. Eq.(3) is actually the discrete version of the Petrov-Galerkin projection [23]. Eq.(4) involves a feed-forward network $F : \mathbb{R}^{n\times c}\rightarrow\mathbb{R}^{n\times c}$, which augments the output representation $\widetilde{\mathbf{z}}$ with extra nonlinear components. When multiple Galerkin attention layers are concatenated, we can progressively update the latent bases over which the latent representation get optimized.

TABLE I

COMPARISON OF THE COMPUTATIONAL COMPLEXITY BETWEEN TRADITIONAL ATTENTION AND GALERKIN ATTENTION, N=100, C=32.

Method	GPU Memory (MB)	GPU Time (ms)
Traditional Attention	3059.1	16
Galerkin-type Attention	**21.1**	**0.7**

C. RGBD Function Embedding

Neural operators are characterized by the employment of kernel integrals in each network layer, which contributes to maintain the continuum between adjacent layers and thus plays an important role in the robustness to discretization sizes [25]. Both the traditional attention [17] and the Galerkin attention [23] fulfill such an kernel integral operation, but CNNs not. In this work, we devise a RGBD embedding module that complies with the neural operator formulation while employ a traditional attention based backbone[21].

In order to employ the efficient Galerkin attention at the original high-resolution scale to explore the interaction of the RGB and depth modalities, we need two preprocessing operations. First, transform the discrete representations in Eq.(1) over direct product spaces into ones over a common support $\{x_i\}_{i=1}^{n_{h_f}}$. Second, map the raw measurements of two modalities into respective latent feature with same dimensions. In Fig.1, these can be easily implemented via convolutional layers. Then we use the Galerkin attention mechanism to further fuse and optimize the latent representations. Specifically, the respective feature maps of the RGB and the depth channel, $G, I \in R^{n_{h_f}\times 32}$, are fused via the cross attention form,

$$\mathbf{z} = Attn_{CA}(G, I) := Q_G(\widetilde{K_I}^T\widetilde{V_I})/n \tag{5}$$

where Q_G=GW_G^Q, K_I=IW_I^K, V_I=IW_I^V, with the projection matrices $W_G^Q, W_I^K, W_I^V \in \mathbb{R}^{32\times 32}$. Through the cross-attention calculation in Eq.(5), the RGBD latent representations are fused together. To further optimize the RGBD latent expression, the feature obtained through cross-attention is concatenated with the RGB feature in the channel dimension, resulting in a latent representation $\mathbf{y}$ of 64 channels.

$$\mathbf{z} = Attn_g(\mathbf{y}) := Q_y(\widetilde{K_y}^T\widetilde{V_y})/n \tag{6}$$

As shown in Eq.(6), through another set of learnable matrices $W^Q, W^K, W^V \in \mathbb{R}^{64\times 64}$, the fused representation $\mathbf{y}$ undergoes a self attention processing, obtaining three instance specific latent bases for the input representation $\mathbf{y}$. Note that although linear transformations can not improve the expressiveness capacity, the nonlinear FFN in Eq.(4) provides opportunity of basis function augmentation, Q_y=$\mathbf{y}W^Q$, K_y=$\mathbf{y}W^K$, V_y=$\mathbf{y}W^V$. By this self-correlation attention calculation, further fusion and optimization are achieved, thus obtaining a latent expression with high-resolution RGBD information.

III. EXPERIMENTS

A. Datasets

The KITTI depth completion dataset[27]: which contains 86898 training data, 1000 selected validation data, and 1000 test data. Sparse depth maps have only 5.9% of the pixels in each image containing depth information, and the dense ground truth contains 30% real depth measurements.

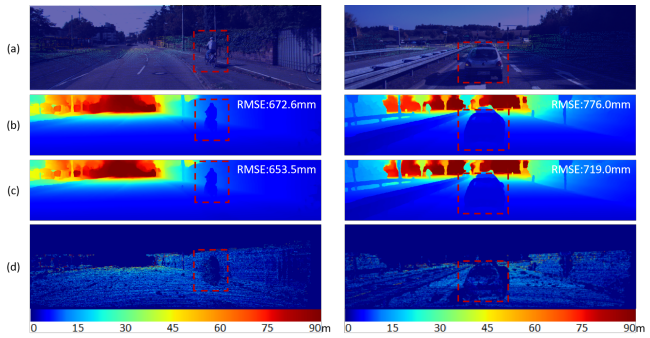

Fig. 2. Completion results on the KITTI dataset (zoom in for a better display) (a) Raw color image overlapped with sparse depth measurements; (b) completed depth map by the network[21]; (c) completed depth map by our model; (d) the residual maps of (c) with respect to the ground truth.

The NYUv2 dataset[28]: which contains 464 indoor scenes. We use the same method as previous work[21] to train 50,000 images and test the results on 654 images. The size of each image during training and testing is 640×480, and the sparse depth map comes from random sampling of the collected dense depth map with the sampling rate 1/500.

TABLE II

COMPARISON WITH OTHER MODELS ON DIFFERENT SCAN LINES IN THE KITTI DATASET

Scanning Lines	Method	RMSE (mm)	MAE (mm)	iRMSE $(1/km)$	iMAE $(1/km)$
4	NLSPN(ECCV2020)	2293.1	831.3	7.0	3.4
	DySPN(AAAI2022)	2285.8	834.3	6.3	3.2
	Completionformer(CVPR2023)	2241.2	795.9	5.8	2.9
	Ours	**2167.6**	**677.8**	**4.4**	**1.9**
16	NLSPN(ECCV2020)	1288.9	377.2	3.4	1.4
	DySPN(AAAI2022)	1274.8	366.4	3.2	1.3
	Completionformer(CVPR2023)	1268.9	360.7	3.3	1.3
	Ours	**1225.2**	**295.8**	**2.7**	**1.0**
64	NLSPN(ECCV2020)	889.4	238.8	2.6	1.0
	DySPN(AAAI2022)	878.5	238.6	2.5	1.0
	Completionformer(CVPR2023)	872.0	226.2	2.5	1.0
	Ours	**853.2**	**189.7**	**2.2**	**0.8**

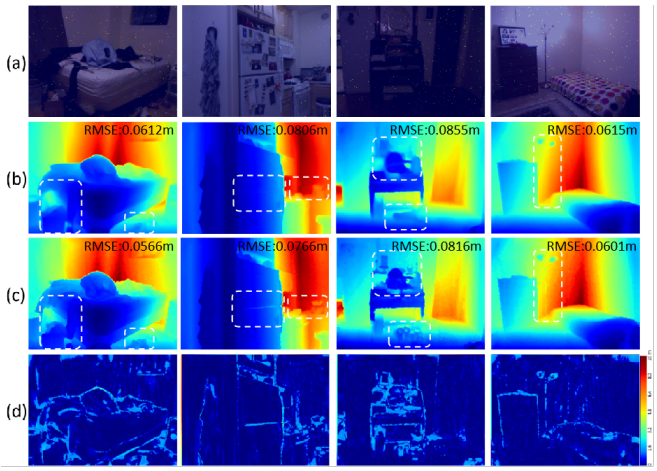

Fig. 3. Completion results on the NYUv2 dataset (zoom in for a better display).(a) Initial color image fused randomly sampled sparse depth map; (b) completed depth map by the network[21]; (c) completed depth map by our model; (d) the residual maps of (c) with respect to the ground truth.

TABLE III

COMPARISON WITH OTHER MODELS ON THE NYUV2 DATASET

Method	NYUv2	
	RMSE (m)	REL
CSPN(ECCV2018)	0.117	0.016
DeepLiDAR(CVPR2019)	0.115	0.022
GuideNet(TIP2020)	0.101	0.015
NLSPN(ECCV2020)	0.092	0.012
ACMNet(2021)	0.105	0.015
Twise(CVPR2021)	0.097	0.013
RigNet(ECCV2022)	**0.090**	0.012
Completionformer(CVPR2023)	0.091	0.012
Ours	0.091	**0.011**

TABLE IV

ABLATION STUDY OF OUR MODULE ON DIFFERENT DATASETS

Method	KITTI DC				NYUv2	
	RMSE (mm)	MAE (mm)	iRMSE $(1/km)$	iMAE $(1/km)$	RMSE (m)	REL
Without our module	878.85	192.47	2.26	0.83	0.09187	0.0119
Ours	**864.72**	**190.61**	**2.21**	**0.82**	**0.09178**	**0.0117**

B. Implementation Details

In this paper, we use a UNet-like structure similar to the Completionformer[21] as our backbone network, and use NLSPN[6] as the subsequent spatial propagation network module to refine the depth map.

Due to the GPU constraint, we randomly selected multiple groups of 10,000 images from the KITTI data set, and downsampled them to the respective 4, 16 and 64 Lidar scanning lines[9] as training sets.

We build our model on Pytorch[29] and use 2 NVIDIA 3090 GPUs for training, validation and testing. We use AdamW[30] as the optimizer. For the KITTI dataset and the NYUv2 dataset, train for 100 and 72 epochs, respectively.

C. Evaluation criteria

We use RMSE (Root Mean Squard Error), MAE (Mean Squared Error), iRMSE (root mean squared error of the inverse depth), iMAE (mean absolute error of the inverse depth) and REL (mean absolute relative error) as the criteria[31].

D. Results and Comparisons With Prior Work

In Table II, we compare with other models using KITTI scenes with 4, 16, and 64 LiDAR scanning lines (the same training set with [21]). Our model achieves significant improvements on all metrics. We also conducted experiments on the indoor NYUv2 dataset and Table III shows that we achieved the best position on the REL metric and a competitive performance on the RMSE metric. Table IV

presents the ablation study results on our high-resolution RGBD embedding module, where we use another random training set and obtain consistent performance advantages as in Table II and III.

In Figure 2 and Figure 3, we show several results from the test datasets. Note that with our high-resolution embedding module, the object boundaries marked with white rectangles are successfully recovered with fine structures, even in large regions without any depth measurement, which manifests the significance of high-resolution RGB guidance and aggregation.

IV. CONCLUSIONS

We have shown that only by using an RGBD function embedding module at the very beginning stage of the depth completion network, significant improvements can be achieved, which is attributed to the efficient and expressive Galerkin attention blocks. Since depth completion tasks usually recover depth values at the regular pixel coordinates, the presented work does not contain a positional embedding component because CNNs are capable of implicit positional encoding [32]. However, we conjecture a suitable positional embedding scheme will support depth inference on arbitrary positions, towards a depth completion neural operator [22], which is an interesting problem in future research.

References are important to the reader; therefore, each citation must be complete and correct. If at all possible, references should be commonly available publications.

REFERENCES

[1] L. Keselman, J. Iselin Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel realsense stereoscopic depth cameras," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 1–10.

[2] Y. Lin, T. Cheng, Q. Zhong, W. Zhou, and H. Yang, "Dynamic spatial propagation network for depth completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 1638–1646.

[3] X. Cheng, P. Wang, C. Guan, and R. Yang, "Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 10 615–10 622.

[4] W. Zhou, X. Yan, Y. Liao, Y. Lin, J. Huang, G. Zhao, S. Cui, and Z. Li, "Bev@ dc: Bird's-eye view assisted training for depth completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9233–9242.

[5] Y. Wang, B. Li, G. Zhang, Q. Liu, T. Gao, and Y. Dai, "Lrru: Long-short range recurrent updating networks for depth completion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9422–9432.

[6] J. Park, K. Joo, Z. Hu, C.-K. Liu, and I. So Kweon, "Non-local spatial propagation network for depth completion," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*. Springer, 2020, pp. 120–136.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[8] Z. Yan, K. Wang, X. Li, Z. Zhang, J. Li, and J. Yang, "Rignet: Repetitive image guided network for depth completion," in *European Conference on Computer Vision*, 2022, pp. 214–230.

[9] S. Imran, X. Liu, and D. Morris, "Depth completion with twin surface extrapolation at occlusion boundaries," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2021, pp. 2583–2592.

[10] S. Liu, S. De Mello, J. Gu, G. Zhong, M.-H. Yang, and J. Kautz, "Learning affinity via spatial propagation networks," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[11] X. Cheng, P. Wang, and R. Yang, "Depth estimation via affinity learned with convolutional spatial propagation network," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 103–119.

[12] Z. Yan, X. Li, Z. Zhang, J. Li, and J. Yang, "Rignet++: Efficient repetitive image guided network for depth completion," *arXiv preprint arXiv:2309.00655*, 2023.

[13] Y. Wang, Y. Mao, Q. Liu, and Y. Dai, "Decomposed guided dynamic filters for efficient rgb-guided depth completion," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[14] L. Liu, X. Song, J. Sun, X. Lyu, L. Li, Y. Liu, and L. Zhang, "Mff-net: Towards efficient monocular depth completion with multi-modal feature fusion," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 920–927, 2023.

[15] M. Hu, S. Wang, B. Li, S. Ning, L. Fan, and X. Gong, "Penet: Towards precise and efficient image guided depth completion," *arXiv preprint arXiv:2103.00783*, 2021.

[16] J. Huang, H.-X. Chen, and S.-M. Hu, "A neural galerkin solver for accurate surface reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, pp. 1–16, 2022.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[18] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[19] D. Neimark, O. Bar, M. Zohar, and D. Asselmann, "Video transformer network," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 3163–3172.

[20] K. Rho, J. Ha, and Y. Kim, "Guideformer: Transformers for image guided depth completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6250–6259.

[21] Y. Zhang, X. Guo, M. Poggi, Z. Zhu, G. Huang, and S. Mattoccia, "Completionformer: Depth completion with convolutions and vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 527–18 536.

[22] M. Wei and X. Zhang, "Super-resolution neural operator," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 247–18 256.

[23] S. Cao, "Choose a transformer: Fourier or galerkin," in *Advances in Neural Information Processing Systems*, 2021.

[24] Y. Chen, S. Liu, and X. Wang, "Learning continuous image representation with local implicit image function," *arXiv preprint arXiv:2012.09161*, 2020.

[25] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces," *arXiv preprint arXiv:2108.08481*, 2021.

[26] A. Ern and J.-L. Guermond, *Theory and practice of finite elements*. Springer, vol. 159.

[27] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant cnns," *arXiv e-prints*, pp. arXiv–1708, 2017.

[28] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *12th European Conference on Computer Vision, ECCV 2012*, 2012, pp. 746–760.

[29] A. P. et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf

[30] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv e-prints*, pp. arXiv–1711, 2017.

[31] Y. Zhu, W. Dong, L. Li, J. Wu, X. Li, and G. Shi, "Robust depth completion with uncertainty-driven loss functions," *arXiv preprint arXiv:2112.07895*, 2021.

[32] M. A. Islam, S. Jia, and N. D. Bruce, "How much position information do convolutional neural networks encode?" in *International Conference on Learning Representations*, 2019.