

Revisiting Depth Completion from a Stereo Matching Perspective for Cross-domain Generalization – supplementary material

Luca Bartolomei^{*,†} Matteo Poggi^{*,†} Andrea Conti[†] Fabio Tosi[†] Stefano Mattoccia^{*,†}

^{*}Advanced Research Center on Electronic System (ARCES)

[†]Department of Computer Science and Engineering (DISI)

University of Bologna, Italy

{luca.bartolomei5, m.poggi, andrea.conti35, fabio.tosi5 stefano.mattoccia}@unibo.it

<https://vppdc.github.io/>

This document provides additional details regarding the 3DV 2024 paper “Revisiting Depth Completion from a Stereo Matching Perspective for Cross-domain Generalization”.

1. Implementation details

In this section, we report additional details concerning our experiments.

1.1. Stereo matching networks

We now describe the training protocol adopted for stereo architectures involved in our experiments. In this phase, none of the models sees any virtual patterned image – i.e., they are trained only on the original, vanilla stereo pairs.

- **RAFT-Stereo [8]**. We trained this stereo network using SceneFlow [9] synthetic dataset, following the original training protocol.
- **PSMNet [1]**. We trained PSMNet using SceneFlow [9], following the original training protocol.
- **GMStereo [18], IGEV-Stereo [17]**. We use authors’ weights to test the networks with our framework.
- **SDC [2] – stereo module (CCVNorm [15])**. We trained CCVNorm on [9] using the PSMNet training protocol.
- **SDC [2] – view synthesis module**. Following the authors’ guidelines, we employed a U-Net [13] style network with a double encoder and a single decoder. The first encoder processes context RGB image and produces multi-scale features $\mathcal{F}_{\text{rgb}}^s$. The second encoder processes sparse warped RGB images and produces multi-scale features $\mathcal{F}_{\text{warp}}^s$. Then, $\mathcal{F}_{\text{rgb}}^s$ and $\mathcal{F}_{\text{warp}}^s$ are concatenated at each scale in the channel dimension and fed to the decoder that generates a dense synthetic right image. We trained the image synthesizer network for 15 epochs on synthetic data [9], using a batch size of 2 and a learning rate of 0.0001. We used AdamW as the optimizer and applied a OneCycle schedule. We generate sparse points randomly sampling 1% of groundtruth data. We also employ RGB augmentation (i.e., color jittering, channels shuffling, histogram equalization) and spatial augmentation (i.e., horizontal flip, random 512×384 crop). Following [2], we employed the \mathcal{L}_{sdc} loss:

$$\mathcal{L}_{\text{sdc}} = \mathcal{L}_{\text{photo}} + \mathcal{L}_{\text{vgg}} \quad (1)$$

\mathcal{L}_{sdc} loss is a sum of two terms: $\mathcal{L}_{\text{photo}}$ (Eq. 2) regularizes the synthetic image I_s with the groundtruth real image \hat{I} , while \mathcal{L}_{vgg} (Eq. 3) indirectly regularizes I_s by enforcing its features representation \mathcal{F}_s to be close to features representation $\hat{\mathcal{F}}$ of \hat{I} . Features \mathcal{F}_s and $\hat{\mathcal{F}}$ are computed using a pretrained VGG.

$$\mathcal{L}_{\text{photo}} = \mathbb{E} \left[\lambda \frac{1 - \text{SSIM}(I_s, \hat{I})}{2} + (1 - \lambda) |I_s - \hat{I}| \right] \quad (2)$$

Where $\lambda = 0.85$ is a hyperparameter that weight structure similarity loss and \mathcal{L}_1 loss.

$$\mathcal{L}_{\text{vgg}} = \mathbb{E} \left[\left(\mathcal{F}_s - \hat{\mathcal{F}} \right)^2 \right] \quad (3)$$

Differently from SDC’s protocol, we employed a \mathcal{L}_2 loss instead of KL-divergence loss for \mathcal{L}_{vgg} term since we encountered numerical stability issues using original loss.

- **OpenCV SGM [6].** To conclude, we report the parameters used for OpenCV SGM algorithm. We set P1, P2, BlockSize to 1176, 4704 and 16 respectively. We apply background interpolation [10] to fill invalid disparity values.

1.2. Depth completion networks

We now describe in detail the training protocol adopted for the conventional completion networks used in our evaluation.

- **CompletionFormer [19].** At first, using the authors’ training protocol, we train the network using synthetic data [9]. Specifically, we train in depth domain for 20 epochs, sampling 1% random points from the groundtruth to generate sparse measurements. The learning rate was set to 0.001 and batch size to 4. We set a depth cap to 90 meters for numerical stability. In addition to the authors’ proposed data augmentations, we used 512×384 random crop. Starting from a synthetic checkpoint, we also fine-tuned the network on real indoor [11] and outdoor [14] datasets. Specifically, on NYU [11] we trained CompletionFormer for 20 epochs with batch size 16, a learning rate set to 0.0001 and a depth clamp at 10 meters. Five hundred depth seeds were sampled randomly from groundtruth. We trained on full image size, discarding random crops. Instead, on KITTI DC [14] we trained for 10 epochs with batch size 4, a learning rate set to 0.0001 and a depth clamp at 90 meters. We used the raw LiDAR data as the depth seed source. We trained on full image size, discarding random crops. Finally, for training on KITTI DC and NYU from scratch, we used the original authors’ weights.
- **NLSPN [12] and SpAgNet [4].** Starting from the authors’ proposed training protocols, we train the network on synthetic data [9] using Adam for 20 epochs at batch size 6 and a learning rate set to 0.001. We generated deep seeds randomly sampling 1% of groundtruth data. As in [19], we set a depth cap at 90 meters, and used random 512×384 crop in addition to the original data augmentation protocol. From synthetic pre-training, we fine tuned in real indoor [11] and outdoor [14] scenarios. On NYU [11], we trained those networks for 20 epochs at batch size 20, a learning rate set to 0.0001 and a depth cap at 10 meters. We sampled 500 depth seed from groundtruth as sparse measurements. We trained on full image size, discarding random crops. Instead, on KITTI DC [14] we apply the same parameters described before for CompletionFormer. Finally, for training on KITTI DC and NYU from scratch, we used original authors’ weights. When evaluating with NLSPN [12] on DDAD [5], we downsampled at half resolution due to memory constraints, then after upsampling the predicted depth map, we evaluated at full resolution.
- **VPP4DC.** As mentioned in the main paper, we selected RAFT-Stereo [8] as the reference stereo network in our framework. Starting from the original training protocol, we trained on KITTI DC [14] from scratch for 10 epochs with batch size 4 and a learning rate of 0.0001. As done before, we applied a depth clamp at 90 meters. To save memory and time, we used 608×240 random crop in addition to spatial augmentation (i.e., horizontal flip) and RGB augmentation (i.e., color jittering, channels shuffling, histogram equalization). On the same dataset, we fine-tuned from Sceneflow [9] pre-train using the same configuration aforementioned, except for the learning rate set to 0.00001. Instead, on NYU [11], we trained from scratch for 20 epochs at batch size 12, a learning rate of 0.0001 and a depth clamp at 10 meters. Five hundred points were randomly sampled from groundtruth as depth seeds. We trained on full image size, discarding random crops. On the same dataset, we fine-tuned from Sceneflow [9] pre-train using the same configuration aforementioned, except for the learning rate set to 0.00001.

All depth completion training and fine tuning use a \mathcal{L}_{12} loss:

$$\mathcal{L}_{12} = \mathbb{E} [|z_r - \hat{z}_r|] + \mathbb{E} \left[(z_r - \hat{z}_r)^2 \right] \quad (4)$$

For traditional depth completion method SCPU [7], we set the hyper parameter of maximum depth equal to 10 meters for NYU [11] and VOID, keeping the original hyperparameters for KITTI DC [14]. Instead, for DDAD [5], we set maximum depth equal to 250 meters. Finally, for RAL [20] depth completion method, we disabled internal occlusion system as outdoors methods are already filtered using [3]. Also, we deploy small changes to execute the algorithm to other datasets rather than KITTI DC [14]: we removed depth clamps and hardcoded depth map shape.

Network	Test domain	RMSE (m)	MAE (m)
SCPU [7]	NYU [11]	0.544	0.240
RAL [20]	NYU [11]	0.495	0.237
VPP4DC [6] (Ours)	NYU [11]	0.321	0.129
SCPU [7]	VOID500 [16]	1.749	1.347
RAL [20]	VOID500 [16]	1.129	0.481
VPP4DC [6] (Ours)	VOID500 [16]	0.707	0.260
SCPU [7]	KITTI DC [14]	1.591	0.329
RAL [20]	KITTI DC [14]	1.571	0.345
VPP4DC [6] (Ours)	KITTI DC [14]	3.442	0.835
SCPU [7]	DDAD [5]	29.144	14.389
RAL [20]	DDAD [5]	7.383	1.579
VPP4DC* [6] (Ours)	DDAD [5]	8.763	2.254

Table I. **Capabilities of traditional algorithms in LiDAR-only depth completion.** The adaptive patch approach is turned off in VPP4DC to ignore the RGB context information. * With a baseline of 0.22 meters and maximum disparity of 256 pixels.

	Test domain	RAFT-Stereo[8]		IGEV-Stereo[17]	
		RMSE (m)	MAE (m)	RMSE (m)	MAE (m)
With context	NYU [11]	0.247	0.077	0.324	0.107
Without context	NYU [11]	0.343	0.151	0.277	0.132
With context	KITTI DC [14]	1.609	0.413	2.258	0.604
Without context	KITTI DC [14]	5.567	1.448	2.430	0.700

Table II. **Capabilities of context-aware stereo networks without context.** As expected, additional information from the RGB context image helps reduce errors.

2. Additional experiments

We investigate the impact of using the RGB contextual information on the performance of traditional stereo matchers such as SGM [6] and context-aware deep learning stereo matchers [8, 17].

LiDAR-only depth completion using traditional stereo matching. We now assess the performance of VPP4DC using a popular traditional stereo algorithm (i.e., OpenCV-SGM [6]) with respect to other handcrafted depth completion methods [7, 20]. Table I shows the accuracy achieved on four different indoor and outdoor datasets (i.e., NYU [11], VOID [16], KITTI DC [14], DDAD [5]). Not surprisingly, RAL and SCPU outperform VPP4DC with KITTI DC evaluation since these traditional algorithms are conceived ad-hoc for this dataset. However, moving to a similar outdoor dataset such as DDAD, SCPU accuracy drops compared to other methods: this evidence proves that SCPU is too specialized on the KITTI DC dataset. RAL is the absolute winner, reducing by 30% the MAE error compared to the second-best method VPP4DC. Instead, when facing challenging indoor datasets, VPP4DC outperforms all other traditional depth completion methods, showing a better capability to handle these different domains.

Effects of context on context-aware stereo networks. Given a context-aware stereo matching network such as RAFT-Stereo [8] or IGEV-Stereo [17], we ablate the impact of the RGB context feeding such networks with the real input image or a black image as context. Table II shows the result of this experiment. As expected, the RGB image context is highly beneficial for both networks – especially for RAFT-Stereo – and crucial for achieving the best results, as shown in Figure I.

3. Design choices: qualitative results

We show additional qualitative results about the proposed virtual projection method to demonstrate the effects of i) random projection; ii) raw depth occlusion filtering; iii) left padding; iv) adaptive patch.

- **(i) Uniform areas in RGB projection.** Figure II shows a qualitative view of a KITTI DC [14] fictitious scene projected using RGB projection (up) and random projection (bottom), respectively. RGB projection shows a large white uniform area due to the massive direct light on the camera sensor. This uniform area propagates in the fictitious stereo pair with RGB projection; consequently, the stereo pair is ambiguous. In contrast, Random projection does not rely on RGB information to generate a fictitious stereo pair, inherently making ambiguous uniform areas distinctive.
- **(ii) Artifacts on raw sparse depth map.** When facing depth completion setups where the RGB camera and depth sensors do not share the very same view, occlusions appear in the camera view’s sparse depth map. As shown in the top of Figure III, when we fed this raw depth map to our virtual projection framework, repetitive patterns appeared: occlusions lead the

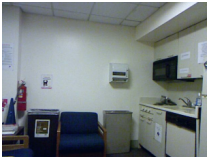

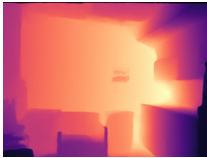

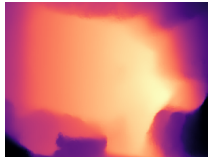





Context	Context aware		Context unaware	
 NYU [11]	 RAFT-Stereo [8]	 IGEV-Stereo [17]	 RAFT-Stereo [8]	 IGEV-Stereo [17]
 KITTI DC [14]	 RAFT-Stereo [8]	 IGEV-Stereo [17]	 RAFT-Stereo [8]	 IGEV-Stereo [17]

Figure I. **Visual effects of RGB context on stereo networks.** The RGB image allows the networks to recover better fine details not available in the sparse fictitious images generated by our projection method.

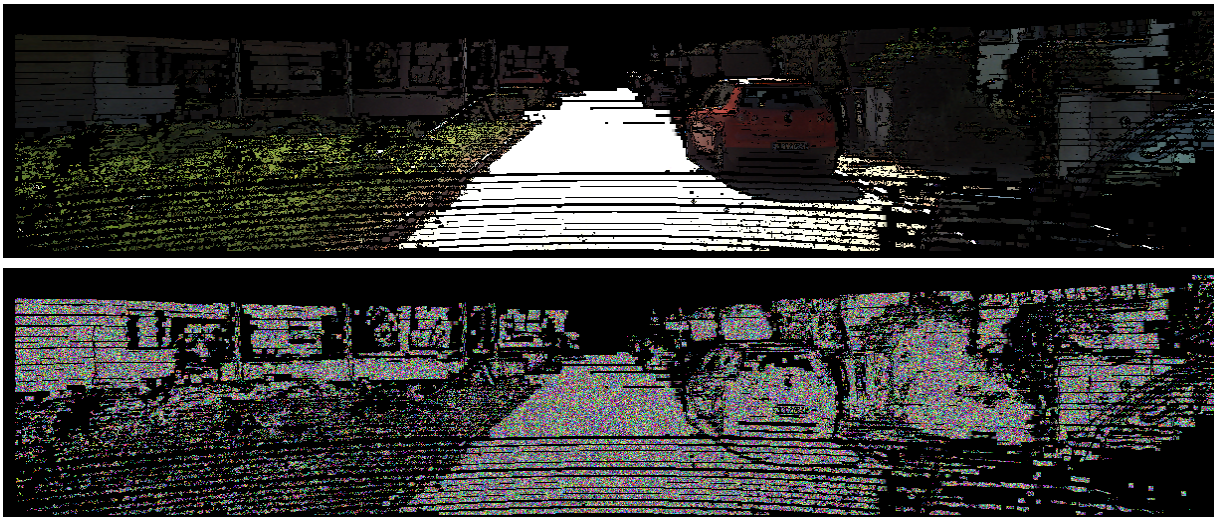


Figure II. **Qualitative view on uniform areas.** RGB projection keeps uniform regions as such, while random projection makes them distinctive.

stereo matching algorithm to match the same region (e.g., the right side of cyclist’s body) to multiple locations. However, using the filtering technique [3] (Fig. III, bottom), the artifacts above vanish, and only the real matching region remains.

- **(iii) Recovering out-of-image depth seeds using left padding.** We qualitatively investigate the behaviour of RAFT-Stereo [8] and OpenCV-SGM[6] stereo matchers when facing left padding. Figure IV shows the effects of left padding on the stereo matchers above. In this particular indoor scenario [11], differences are less visible since there are only a few out-of-border depth seeds. We argue that these scarce depth points add a marginal contribution to the dense depth reconstruction process. Instead, in the outdoor scenario [14], we can appreciate a stronger recovery of out-of-border objects, such as the road blue sign.
- **(iv) Adaptive patch.** Figure V shows the effects of the adaptive patch used for an outdoor scenario [14]. Given a context image (Fig. V, left), our methodology shapes the original block-based patch (Fig. V, center) to a more appropriate silhouette (Fig. V, right). Supported by the figure above and hyper-parameters ablation, we argue that this method helps to preserve fine details that otherwise would have been lost.

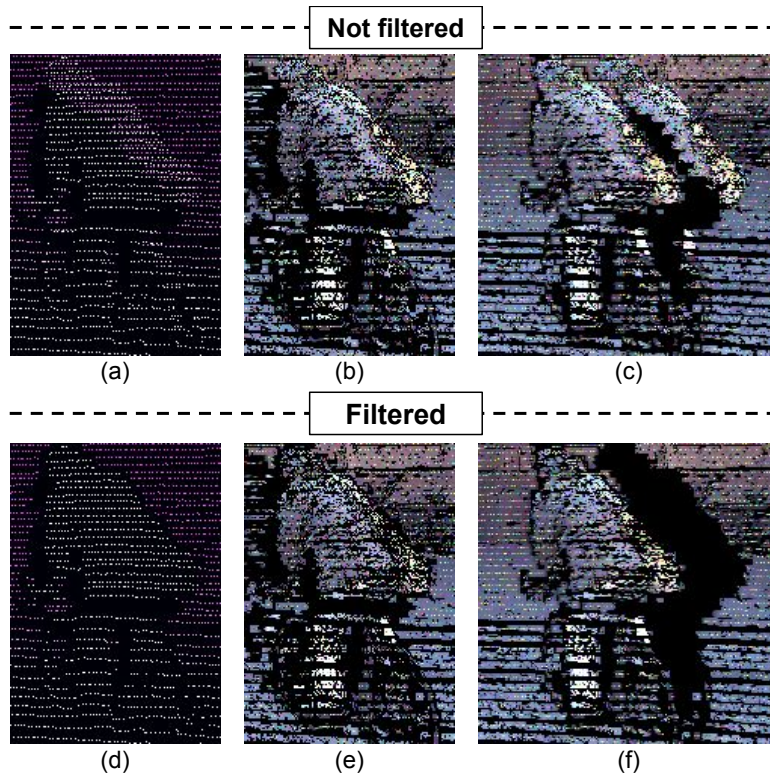


Figure III. **Qualitative effects of depth occlusion filtering.** The different position between the RGB camera and depth sensor (a) yields artifacts in virtual projection (b,c). Filtering out occluded points according to [3] (d) significantly softens the problem (e,f).

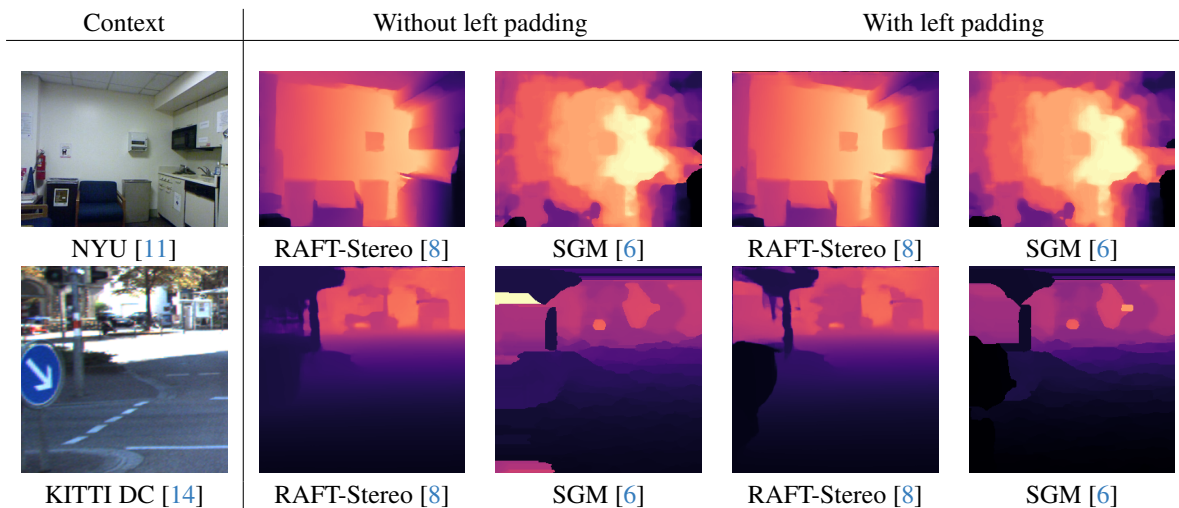


Figure IV. **Qualitative effects of left padding.** Effects are less visible in indoor dataset [11], while they are more apparent in outdoor scenarios [14]. Without left padding, stereo matchers struggle on left occluded objects (e.g., the blue road sign).

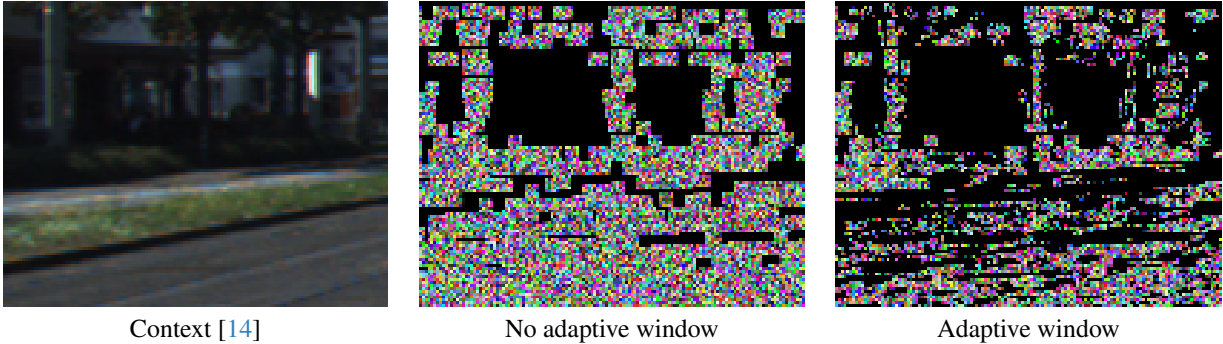


Figure V. **Qualitative effects of adaptive patch.** Given a context image, our proposal adapts the patch to objects in the scene (e.g., the tree at the center).

4. Virtual baseline study

In this section, we complement Fig. 7 in the main paper by adding a comparison between different virtual baselines in outdoor datasets [5, 14]. In particular, we inspect the discrepancy between the virtual baseline used in DDAD [5] and the one used in KITTI DC [14], – both acquired in a similar environment – in Fig. VI and VII. Firstly, we compare the depth distribution of the two datasets in Fig. VI (a) for the near depth range – *i.e.*, 0-10m. Supported by the distribution shown in the figure and the fact that both datasets are acquired in similar conditions, we agree that both outdoor datasets share similar close-range depth distributions. Then, we investigate the inverse close-depth function for both datasets in Fig. VI (b), using the focal length given by the real camera and the optimal virtual baseline given by Fig. 7. Additionally, we plot the inverse close-depth function for DDAD [5] with a larger virtual baseline, equal to 0.54m – *i.e.*, the same virtual baseline of KITTI DC [14]. As expected, the larger virtual baseline in [5] leads to a larger disparity range, even ~ 3 times the maximum disparity registered for KITTI DC [14] and ~ 2 times the maximum disparity registered using the optimal virtual baseline. This extended range might hinder a deep stereo model not used to deal with it, especially in the close-depth regions, such as the road surface just under the camera.

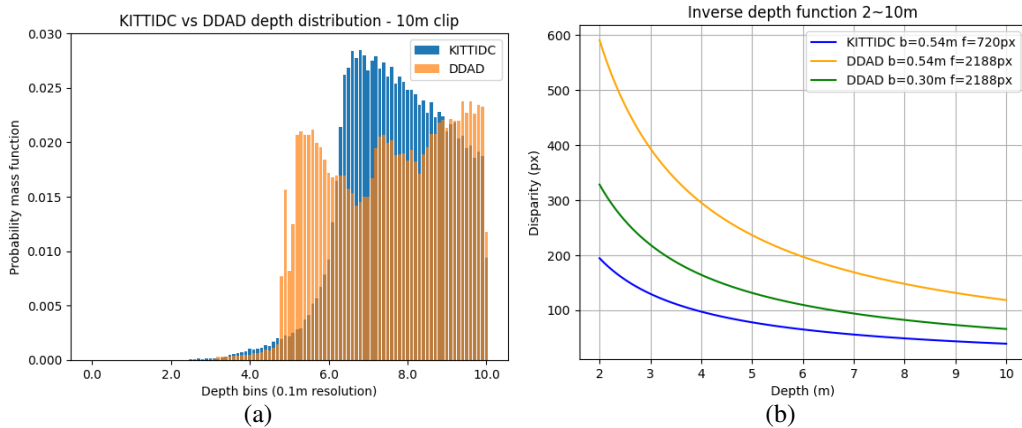


Figure VI. **DDAD [5] virtual baseline study – near depth range.** DDAD [5] and KITTI DC [14] share similar distribution of near range depth values (a). A shorter baseline aligns the disparity range of DDAD [5] to the disparity range seen during training [9] (b).

However, a reduction in baseline leads intrinsically to higher errors, that are proportional to the square of the distance between the camera and the objects. In Fig. VII (a) and Fig. VII (b) we analyze respectively the behavior of the inverse depth function in far depth bound of KITTI DC [14] (*i.e.*, 80-100m) and the one in far depth bound of DDAD [5] (*i.e.*, 200-220m). The former plot qualitatively shows that DDAD [5] resolution is higher than the one of KITTI DC [14], even with a shorter baseline: this is due to the longer focal length used to acquire the former. The latter plot shows that an error of 1px in disparity estimation equals more than 20 meters of depth error, even with a larger baseline. In conclusion, the virtual baseline is an important hyperparameter that needs to be tuned precisely to achieve optimal performance on both near and far depth ranges.

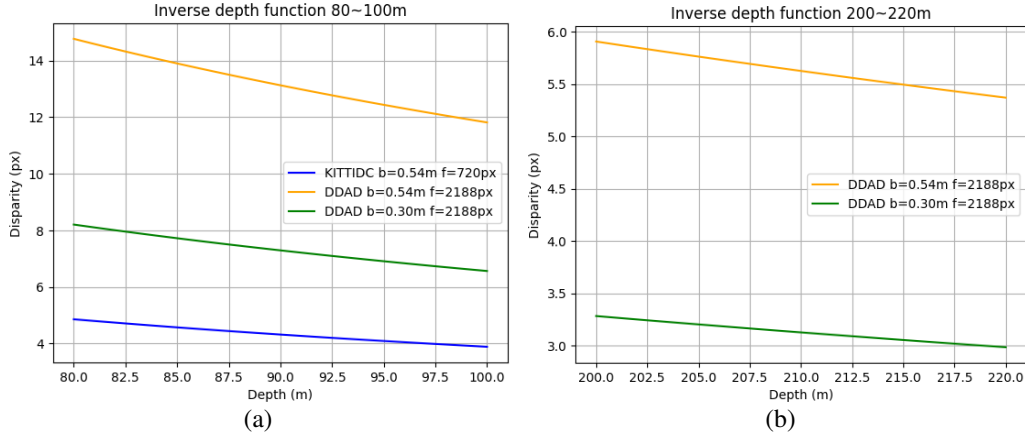


Figure VII. DDAD [5] virtual baseline study – far depth range. Despite the shortened baseline, DDAD [5] maintains comparable disparity resolution to KITTIDC [14], which employs a larger baseline (a). Notably, extended distances exhibit decreased accuracy, even with a larger baseline (b).

5. Cross domain generalization: qualitative results

We conclude by showing some qualitative examples to confirm the benefits of our method in terms of cross domain generalization. In particular, we test the performance of networks v) trained on synthetic data [9] and tested on real scenarios [5, 16]; vi) trained or finetuned indoor [11] and tested outdoor [5]; vii) trained or finetuned outdoor [14] and tested indoor [16]. We apply a 7×7 dilation to sparse depth hints and sparse ground-truth for visualization purpose.

- **v) From synthetic to real.** Fig. VIII and Fig. IX show the performance of three depth completion networks [4, 12, 19] compared to our proposal on both indoor [16] and outdoor [5] scenarios. Even if all the networks are trained on SceneFlow [9] only, in Fig. VIII they manage to reconstruct the road plane. Nonetheless, our proposal outperforms others when dealing with long distance details. Furthermore, in Fig. IX we show a challenging frame from [16] where depth hints are missing at the top of the image. In this case only our method performs adequately for the scenario, yet being not capable of producing meaningful depth values in the portion of the scene lacking any depth hints.

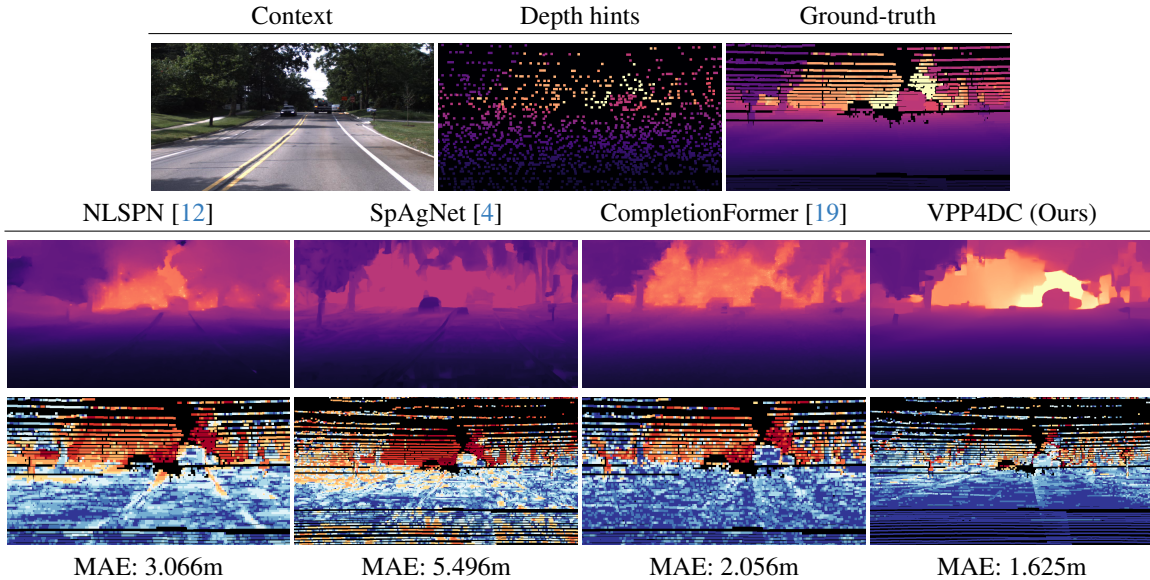


Figure VIII. Qualitative results – DDAD [5]. All networks are trained on SceneFlow [9].

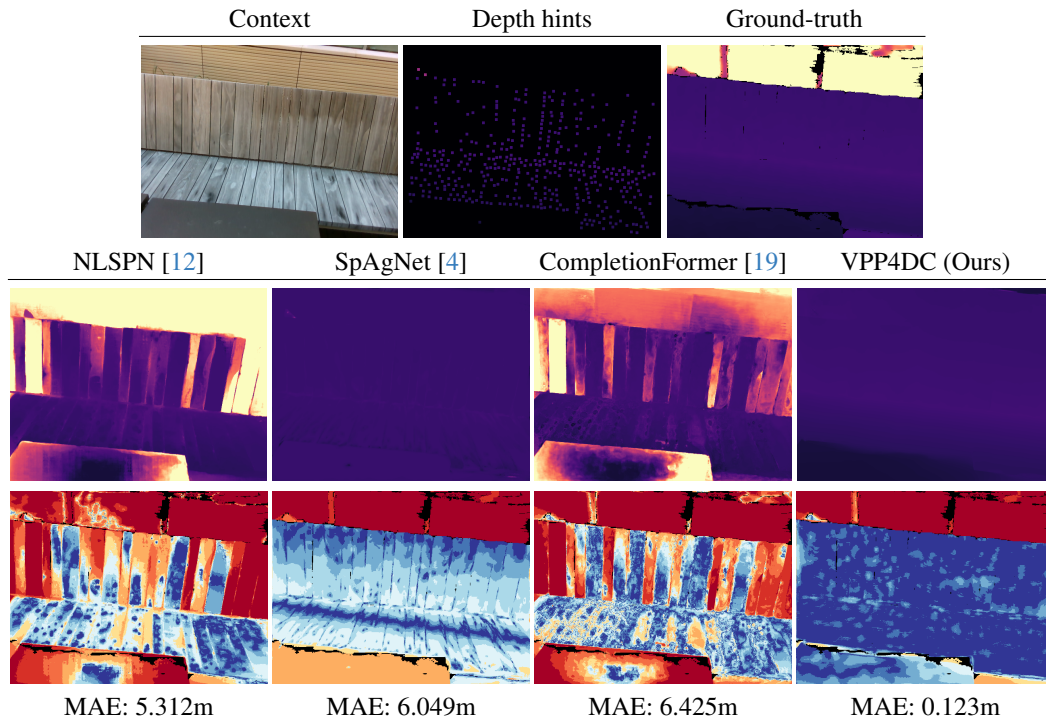


Figure IX. **Qualitative results – VOID500 [16]**. All networks are trained on SceneFlow [9].

- **vi) From indoor to outdoor.** When it comes to train on indoor data and run the model outdoor, significant domain shift occurs. In Fig. X and Fig. XI we show examples on the DDAD dataset, using models trained on NYU only, or pre-trained on SceneFlow and then fine-tuned on NYU, respectively. In the former case, NLSPN and CompletionFormer produce decent results on the road surface, whereas SpAgNet completely fails at delivering any meaningful prediction. Our approach instead predicts a quite structured depth map, with minor failures. When pre-training on SceneFlow, we observe the opposite behavior: NLPSN and CompletionFormer seem unable to generalize to outdoor data, while SpAgNet can produce some meaningful depth maps, yet far from being accurate. Finally, VPP4DC can improve the results even further thanks to the pre-training process.

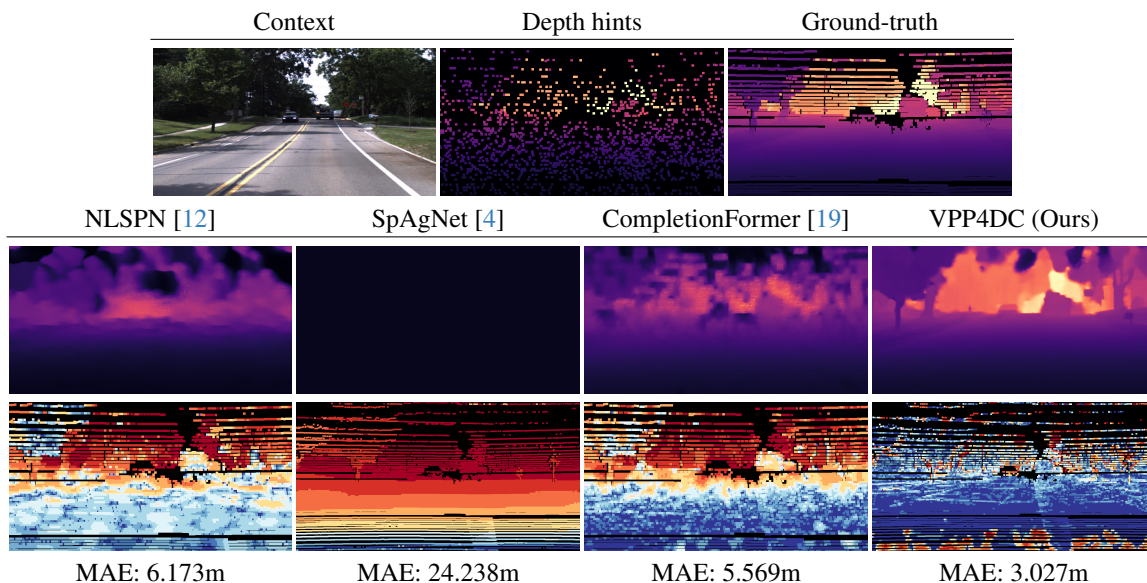


Figure X. **Qualitative results – DDAD [5]**. All networks are trained on NYU [11].

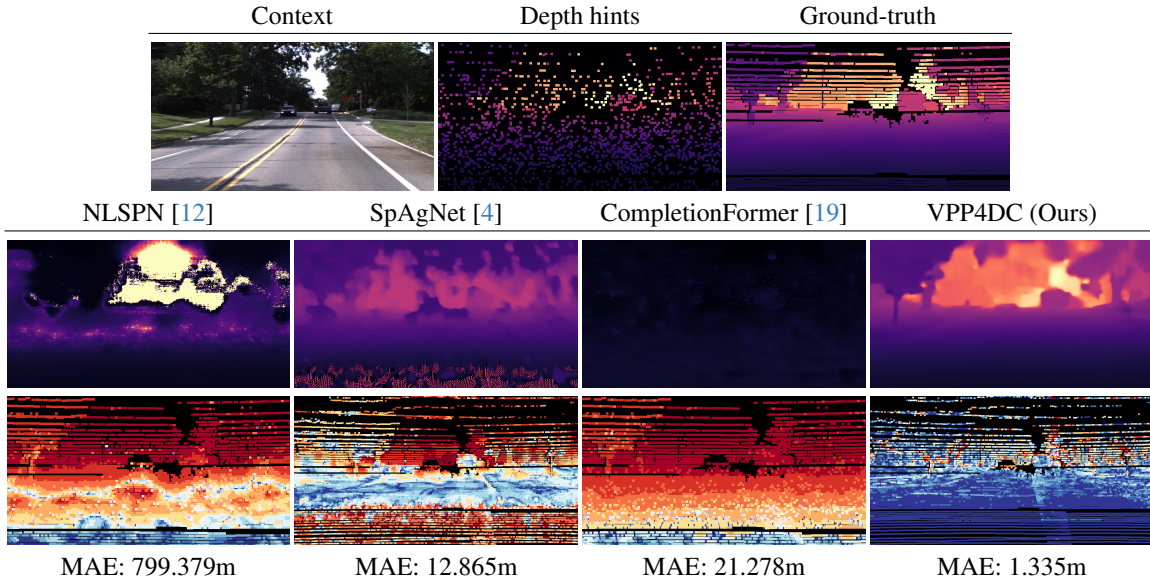


Figure XI. **Qualitative results – DDAD [5]**. All networks are pretrained on SceneFlow [9] and fine-tuned on NYU [11].

- **vii) From outdoor to indoor.** Finally, we consider the case complementary to the previous one – *i.e.*, with models trained outdoor and tested indoor. In Fig. XII and Fig. XIII we show examples on the VOID dataset, using models trained on KITTI only, or pre-trained on SceneFlow and then fine-tuned on KITTI, respectively. In the former case, NLSPN and CompletionFormer completely fail, whereas SpAgNet can predict a depth map that is reasonable to some extent. Our approach instead predicts very accurate results on regions covered by depth hints, yet failing where these are absent. When pre-training on SceneFlow, NLSPN and CompletionFormer can improve their results, yet being far from the results yielded by VPP4DC without any pre-training. Eventually, if SceneFlow pre-training takes place, our method can further improve its results.

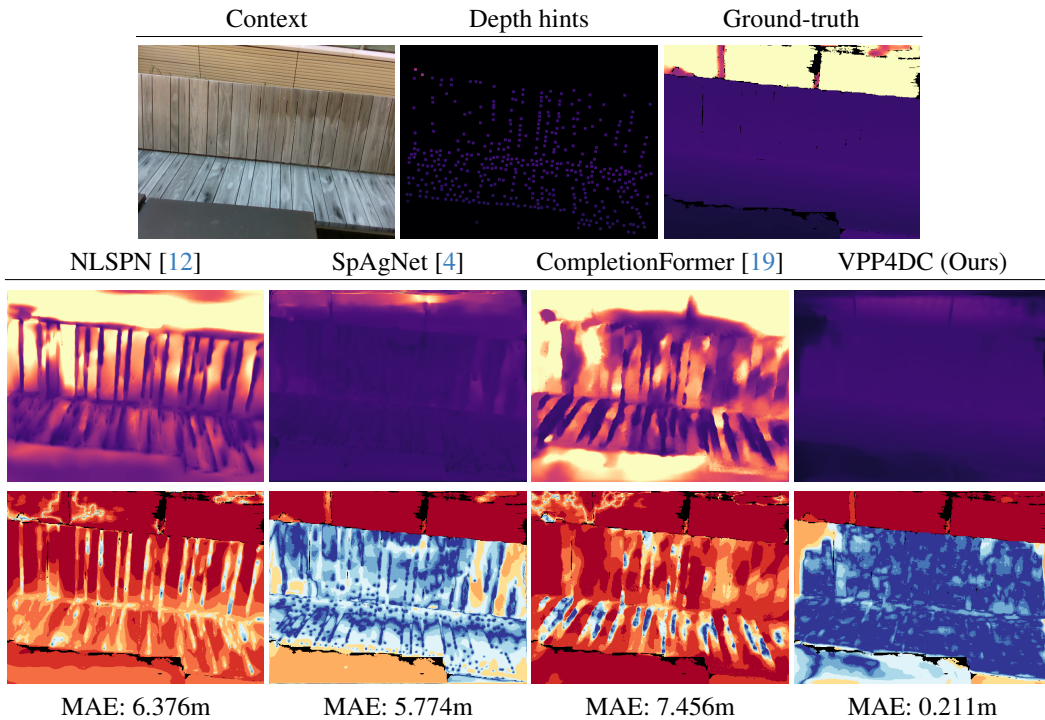


Figure XII. **Qualitative results – VOID500 [16]**. All networks are trained on KITTI [14].

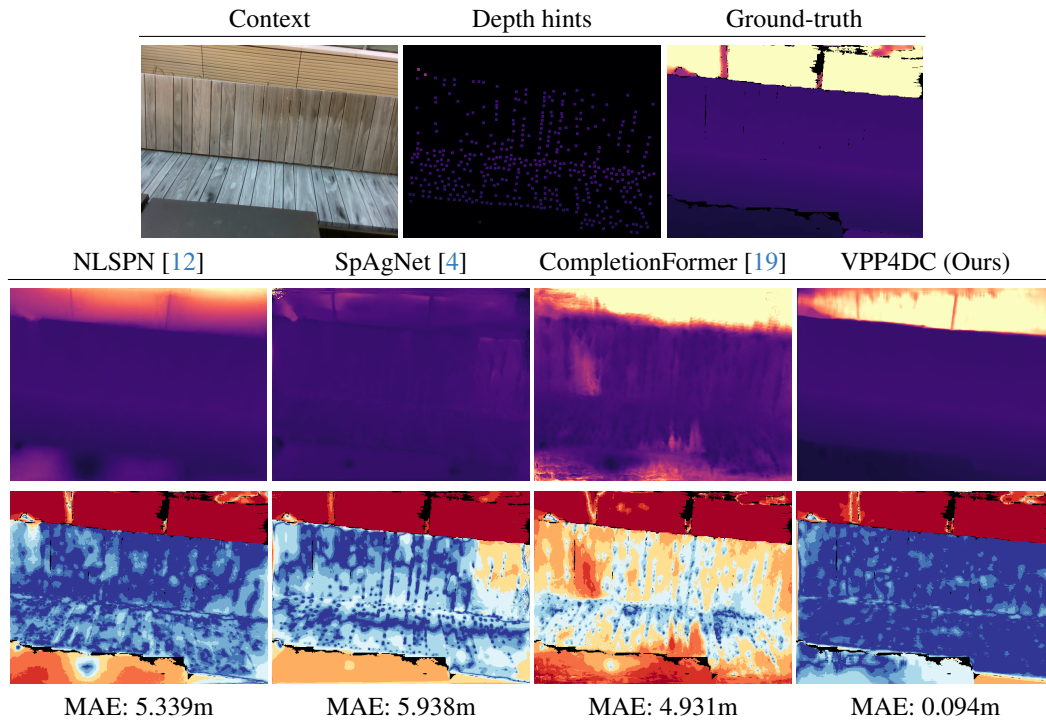


Figure XIII. **Qualitative results – VOID500 [16]**. All networks are pretrained on SceneFlow [9] and finetuned on KITTI [14].

References

- [1] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5410–5418, 2018. 12
- [2] Keunhoon Choi, Somi Jeong, Youngjung Kim, and Kwanghoon Sohn. Stereo-augmented depth completion from a single rgb-lidar image. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13641–13647, 2021. 12
- [3] Andrea Conti, Matteo Poggi, Filippo Aleotti, and Stefano Mattoccia. Unsupervised confidence for lidar depth maps and applications. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022. IROS. 13, 15, 16
- [4] Andrea Conti, Matteo Poggi, and Stefano Mattoccia. Sparsity agnostic depth completion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5871–5880, 2023. 13, 18, 19, 20, 21
- [5] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 13, 14, 17, 18, 19, 20
- [6] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007. 13, 14, 15, 16
- [7] Jason Ku, Ali Harakeh, and Steven L. Waslander. In defense of classical image processing: Fast depth completion on the cpu. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 16–22, 2018. 13, 14
- [8] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *International Conference on 3D Vision (3DV)*, 2021. 12, 13, 14, 15, 16
- [9] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. arXiv:1512.02134. 12, 13, 17, 18, 19, 20, 21
- [10] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 13
- [11] Pushmeet Kohli, Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 13, 14, 15, 16, 18, 19, 20
- [12] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *Proc. of European Conference on Computer Vision (ECCV)*, 2020. 13, 18, 19, 20, 21
- [13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 12

- [14] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *International Conference on 3D Vision (3DV)*, 2017. [13](#), [14](#), [15](#), [16](#), [17](#), [18](#), [20](#), [21](#)
- [15] Tsun-Hsuan Wang, Hou-Ning Hu, Chieh Hubert Lin, Yi-Hsuan Tsai, Wei-Chen Chiu, and Min Sun. 3d lidar and stereo fusion using stereo matching network with conditional cost volume normalization. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5895–5902. IEEE, 2019. [12](#)
- [16] Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 5(2):1899–1906, 2020. [14](#), [18](#), [19](#), [20](#), [21](#)
- [17] Gangwei Xu, Xianqi Wang, Xiaohuan Ding, and Xin Yang. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21919–21928, 2023. [12](#), [14](#), [15](#)
- [18] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezaatofghi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *arXiv preprint arXiv:2211.05783*, 2022. [12](#)
- [19] Youmin Zhang, Xianda Guo, Matteo Poggi, Zheng Zhu, Guan Huang, and Stefano Mattoccia. Completionformer: Depth completion with convolutions and vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18527–18536, 2023. [13](#), [18](#), [19](#), [20](#), [21](#)
- [20] Yiming Zhao, Lin Bai, Ziming Zhang, and Xinming Huang. A surface geometry model for lidar depth completion. *IEEE Robotics and Automation Letters*, 6(3):4457–4464, 2021. [13](#), [14](#)