Deep Surface Reconstruction from Point Clouds With Visibility Information

Raphael Sulzer[1], Loic Landrieu[1], Alexandre Boulch[2], Renaud Marlet[2,3], Bruno Vallet[1]

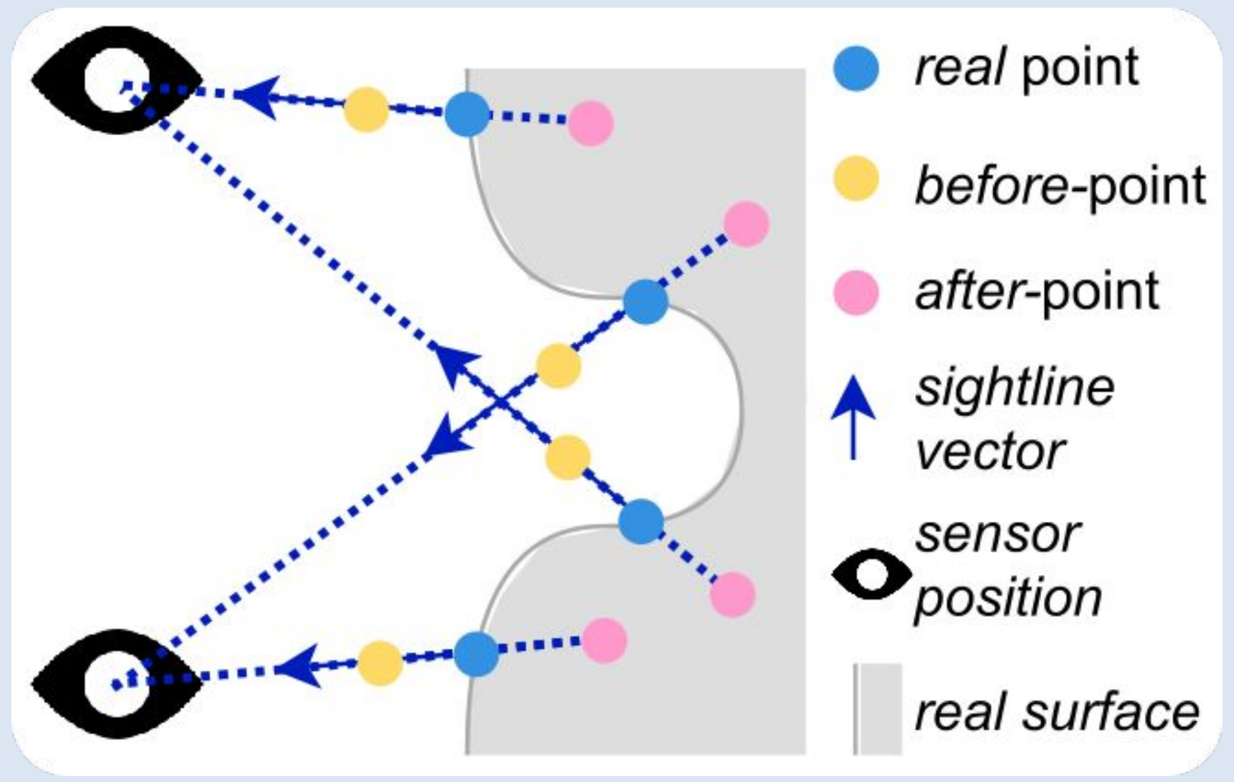
[1]LASTIG, Univ Gustave Eiffel, IGN-ENSG, F-94160 Saint-Mandé, France [2]Valeo.ai, Paris, France
[3]LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France

Introduction

- Surface reconstruction from point clouds has recently been addressed by the deep learning community
- Most deep surface reconstruction (DSR) networks only operate on point locations and ignore sensor poses
- In this work, we consider a 3D point cloud P where each point $p \in P$ has some coordinates $X_p \in R^3$ and coordinates of the sensor observing it $S_p \in R^3$
- We propose two simple ways to augment a point cloud with visibility information using the sensor poses and adapt DSR methods to use such a visibility-augmented point

Method

- *Sightline Vector (SV):* For each $p \in P$, we define a unit vector $v_p \in R^3$ pointing from the observation X_p to the sensor S_p: $v_p = (S_p - X_p)/ // S_p - X_p //$
- *Auxiliary Point (AP):* We augment each point p with a before-point p_b and an after-point p_a, located along the sightline on each side of p: $X_{pb} = X_p + dv_p$, $X_{pa} = X_p - dv_p$, where d is the average distance from a point to its nearest neighbor
- To identify the point type we use tags $t \in R^2$: t_p = [0 0] (observed point), t_{pb} = [1 0] (before-point), and t_{pa} = [0 1] (after-point)



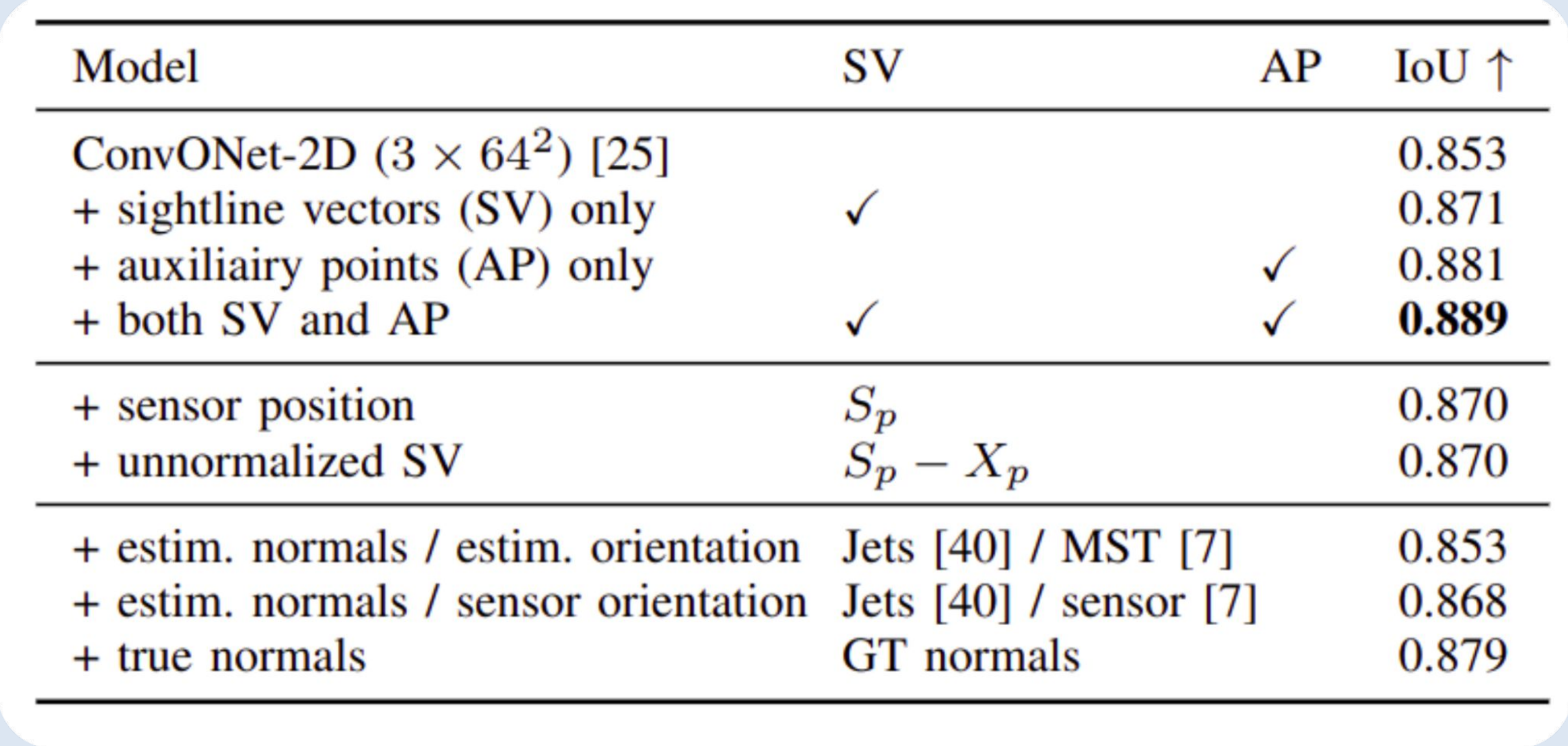
- When combining both kinds of visibility information (SV+AP), before-point p_b and after-point p_a are given the same sightline vector as their reference point, i.e., $v_{pa} = v_{pb} = v_p$, and we take as network input $I_p = (X_p \oplus v_p \oplus t_p) \in R^8$
- We can adapt most DSR networks to handle visibility-augmented point clouds with only few modifications:

i. We change the input size (number of channels) of the first layer of the network (generally an encoder), increasing it by 5

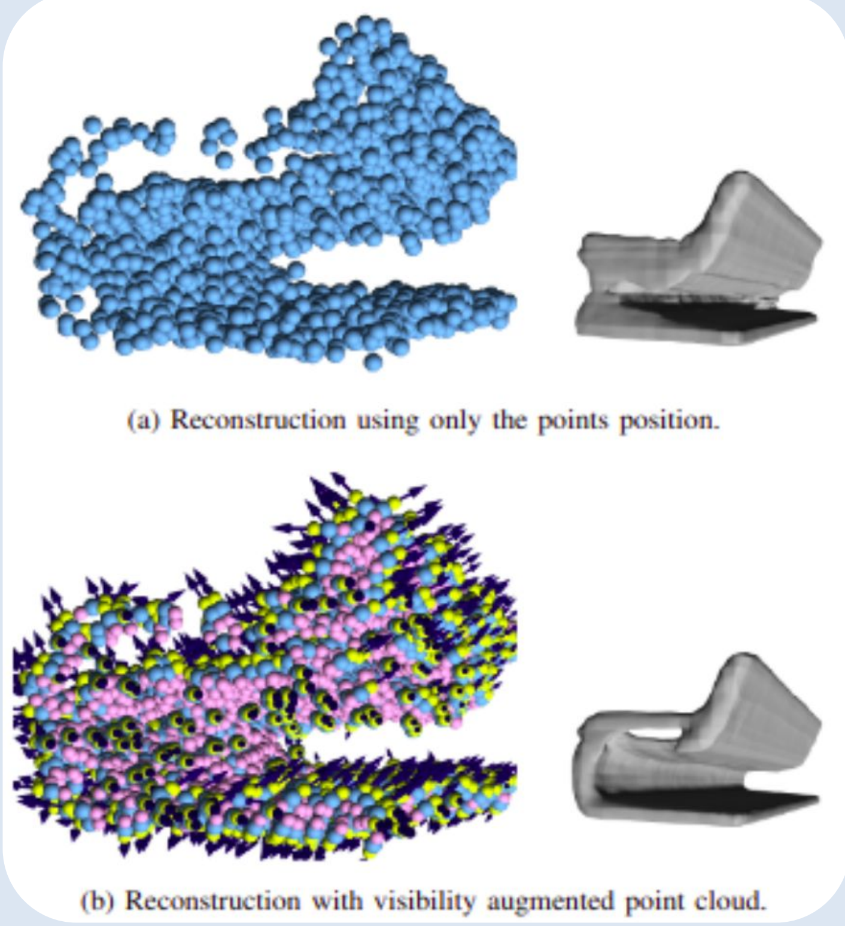
ii. We directly add auxiliary points to the point cloud, thus tripling the number of input points. The batch size may need to be adjusted to fit a larger point cloud in memory. The rest of the network stays unchanged.

Results

- To validate our design we compare various ways to add visibility information to ConvONet [1]. We train different models on the shape dataset ModelNet10 and compare the mean volumetric IoU of reconstructed test shapes:



Model	SV	AP	IoU ↑
ConvONet-2D (3×64^2) [25]			0.853
+ sightline vectors (SV) only	✓		0.871
+ auxiliairy points (AP) only		✓	0.881
+ both SV and AP	✓	✓	**0.889**
+ sensor position	S_p		0.870
+ unnormalized SV	$S_p - X_p$		0.870
+ estim. normals / estim. orientation	Jets [40] / MST [7]		0.853
+ estim. normals / sensor orientation	Jets [40] / sensor [7]		0.868
+ true normals	GT normals		0.879



(a) Reconstruction using only the points position.

(b) Reconstruction with visibility augmented point cloud.

- We also train Points2Surf [2], Shape As Points [3] and POCO [4] with augmented point clouds (SV+AP). The mean volumetric IoU of the ModelNet10 test set increases by **+1 IoU pt** compared to reconstructions from models trained on bare point clouds.
- We also test the same models on unseen shape categories from ShapeNet. Here the mean volumetric IoU increases by **+30 IoU pts** (ConvONet), **+3 IoU pts** (Points2Surf), **+25 IoU pts** (Shape As Points) and **+40 IoU pts** (POCO).

Conclusion

- Visibility information improves reconstruction accuracy as well as generalization capability of DSR networks to unseen domains

Code & Data

References

[1] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, “Occupancy networks: Learning 3D reconstruction in function space,” in CVPR, 2019.

[2] P. Erler, S. Ohrhallinger, N. Mitra, and M. Wimmer, “Points2Surf: Learning implicit surfaces from point clouds,” in ECCV, 2020.

[3] S. Peng, C. M. Jiang, Y. Liao, M. Niemeyer, M. Pollefeys, and A. Geiger, “Shape as points: A differentiable poisson solver,” in NeurIPS, 2021.

[4] A. Boulch and R. Marlet, “Poco: Point convolution for surface reconstruction,” in CVPR, 2022.