Supplementary material: DELS-MVS: Deep Epipolar Line Search for Multi-View Stereo

Christian Sormann [1] christian.sormann@icg.tugraz.at
Emanuele Santellani [1] emanuele.santellani@icg.tugraz.at
Mattia Rossi [2] mattia.rossi@sony.com
Andreas Kuhn [2] andreas.kuhn@sony.com
Friedrich Fraundorfer [1] fraundorfer@icg.tugraz.at

[1] Graz University of Technology
Institute of Computer Graphics and Vision

[2] Sony Europe B.V.
R&D Center - Stuttgart Laboratory 1

1. Implementation details

In this section, we provide additional implementation details for DELS-MVS, which we evaluated using an RTX 3090 GPU.

The feature volumes extracted from the reference and source image, depicted in Figure 1 of the main manuscript, are computed using the multi-resolution feature network architecture proposed in [2, 4], extended with deformable convolutions as employed in [6]. Given an input resolution, the employed feature network yields output feature maps F_h for the resolution levels $h = \{0, 1, 2\}$ representing full, half and quarter resolution, respectively. In terms of the number of feature channels f_h used in each resolution level, we adopt the settings used in [2, 8] and utilize $f_0 = 8$, $f_1 = 16$ and $f_2 = 32$.

DELS-MVS operates on three resolution levels as well. In particular, at the resolution level h, DELS-MVS employs an ER-Net, introduced in Sec. 3.3 of the main manuscript, and feeds it with F_h. During training, the three ER-Nets at the three different resolutions are trained sequentially with no gradient flow between them, therefore we couple each ER-Net with its own feature network. Within this setup, the ER-Net at resolution h is fed with the feature volume F_h from its own feature network. Finally, the network parameter settings of C-Net are reported in Table 1.

	parameters			
op. name	kernel	activation	channels	input
CONV2D$_\mathbf{1}$	3×3	leaky ReLU	$f_C \cdot 2$	$\tilde{C}^n$
CONV2D$_\mathbf{2}$	3×3	leaky ReLU	$f_C \cdot 2$	CONV2D$_\mathbf{1}$
CONV2D$_\mathbf{3}$	3×3	leaky ReLU	f_C	CONV2D$_\mathbf{2}$
CONV2D$_\mathbf{4}$	3×3	leaky ReLU	$\frac{f_C}{2}$	CONV2D$_\mathbf{3}$
C^n	3×3	sigmoid	1	CONV2D$_\mathbf{4}$

Table 1. C-Net network parameter settings. We adopt $f_C = 32$.

Given a pixel location $p^{\mathcal{R}} = (x, y) \in \mathbb{R}^2$ in the reference image, the objective of our Deep Epipolar Line Search, described in Sec. 3.2 of the main manuscript, is the retrieval of its projection $p^n \in \mathbb{R}^2$ along the corresponding epipolar line of the source image $\mathcal{S}^n$. Once p^n has been computed, the desired depth $\mathcal{D}^n(x, y)$ can be obtained by leveraging the known transformation between the reference and source image pixel domains $T_{\mathcal{R}\to\mathcal{S}^n} = [\bar{R}|\bar{t}]$ where $\bar{R} \in \mathbb{R}^{3\times3}$ and $\bar{t} \in \mathbb{R}^3$ denote the rotational and translational components. For the sake of simplicity, we denote $\mathcal{D}^n(x, y)$ as $\mathcal{D}^n$ in the following derivations. In the following, the subscript H denotes homogeneous coordinates and $(\cdot)_{j=x,y,z}$ denotes the j component of a vector. The relation between the reference image pixel $p^{\mathcal{R}}$ and its projection p^n is captured by the following two equations:

$$(\bar{R}p_H^{\mathcal{R}})_x \cdot \mathcal{D}^n + \bar{t}_x = p_x^n \cdot d^n \tag{1}$$

$$(\bar{R}p_H^{\mathcal{R}})_y \cdot \mathcal{D}^n + \bar{t}_y = p_y^n \cdot d^n \tag{2}$$

Eq. (2) provides the following expression for d^n:

$$d^n = \frac{(\bar{R}p_H^{\mathcal{R}})_y \cdot \mathcal{D}^n + \bar{t}_y}{p_y^n} \tag{3}$$

Substituting Eq. (3) in Eq. (1) and solving for D^n yields the desired depth:

$$\mathcal{D}^n = \frac{\bar{t}_y p_x^n - \bar{t}_x p_y^n}{(\bar{R}p_H^{\mathcal{R}})_x p_y^n - (\bar{R}p_H^{\mathcal{R}})_y p_x^n} \tag{4}$$

In Table 2, we show a runtime and memory comparison with other methods on the Tanks and Temples [3] benchmark using $N = 6$ source images with an input image resolution of 1920×1056. We additionally provide the input downscale factor: this indicates whether the method is working, internally, on a downscaled version of the input,

as this results in a lower resolution output. This is also the case for [9], whose output is the result of an up-scaling from half resolution. It can be seen that we achieve competitive memory consumption among the considered methods, especially taking into account that we work at full resolution, like [2, 8, 10].

method	runtime (s)	mem. (GB)	input DS.
PatchMatchNet [9]	**0.30**	**3.9**	0.5
CVP-MVSNet [10]	1.64	9.4	1.0
IB-MVS [8]	7.17	7.4	1.0
CasMVSNet [2]	0.70	9.6	1.0
ours (DELS-MVS)	2.87	6.2	1.0

Table 2. Runtime and memory comparison on Tanks and Temples [3], with resolution 1920×1056 and $N = 6$ source views.

2. Additional visualizations

In this Section we provide additional visual results for DELS-MVS.

In Figure 3, we compare the point cloud reconstructions of the ETH3D [7] sequence `Meadow` provided by DELS-MVS and the state-of-the-art deep-learning-based methods EPP-MVSNet [5] and PatchMatchNet [9]. It can be observed that DELS-MVS reconstruction is more complete and exhibits less noise. In Figure 1, we consider a reference image in the ETH3D [7] sequence `Statue` and visualize both the depth and confidence maps estimated using two source images and the final fused depth and confidence maps. It can be observed that the confidence maps C^n of each source image exhibit a low confidence, denoted by dark areas, in those regions of the reference image that are occluded in the source. This is crucial, as the confidence maps are employed in our robust fusion step, described in Sec. 3.4 of the main manuscript, to fuse the reliable areas of each depth map into a single one. The result of the fusion is shown in the third row of the same figure, where it can be observed that the reliable areas of the depth maps estimated using the two source images, denoted by white areas in the confidence maps, complement each other. In Figure 4, we consider a reference image from one of the DTU [1] sequences and sketch the evolution of the epipolar residual maps $\mathcal{E}^n$ estimated by DELS-MVS over its iterations and scales, for three source images $\mathcal{S}^n$. Additionally, we visualize the final depth maps $\mathcal{D}^n$ obtained from the three source images as well as their fusion $\mathcal{D}$.

Finally, we provide additional DELS-MVS point cloud reconstructions for DTU [1] and Tanks and Temples [3] in Figure 2. It can be observed that the reconstructions exhibit high completeness and low noise.

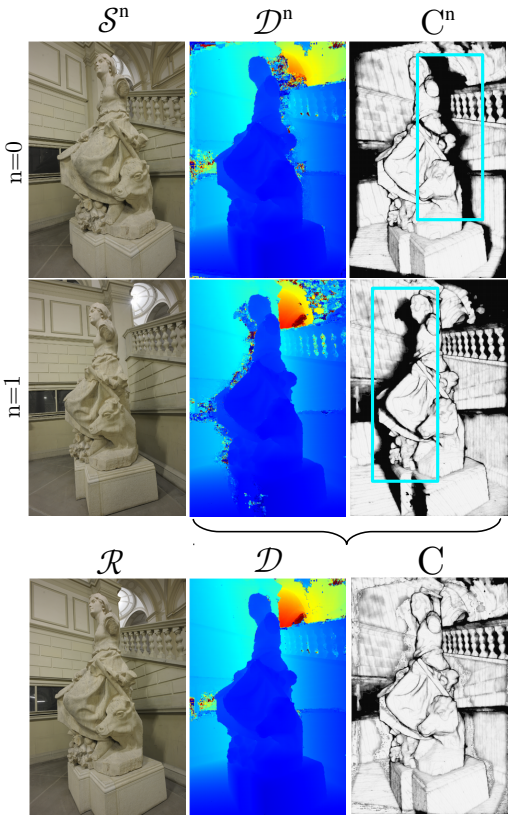

Figure 1. Depth maps estimated by DELS-MVS for the ETH3D high resolution [7] sequence `Statue` using $N = 2$ source images. Each column of the two top rows hosts, from left to right, one of the considered source images $\mathcal{S}_n$, its estimated depth map $\mathcal{D}^n$ and confidence map C^n. The third row hosts, from left to right, the considered reference image, its fused depth map $\mathcal{D}$ and confidence map C.

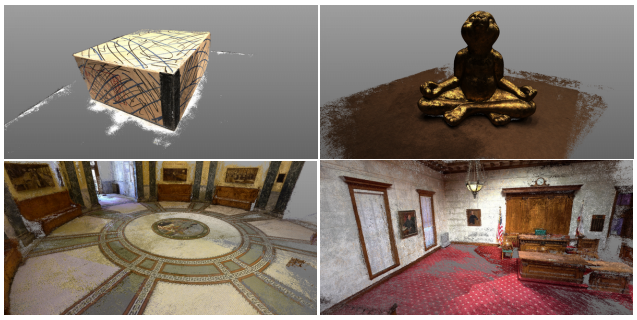

Figure 2. DELS-MVS point cloud reconstructions of sequences from DTU [1] (top) and Tanks and Temples [3] (bottom).

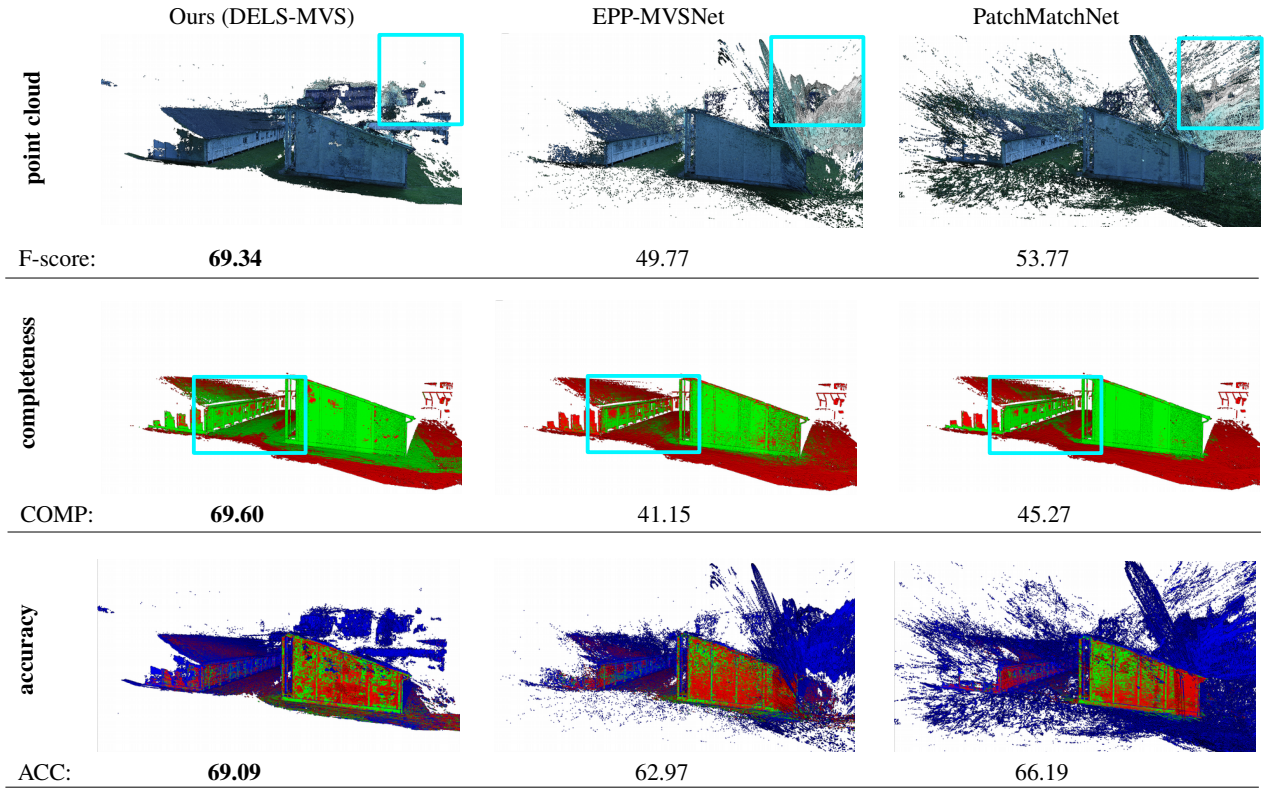

Figure 3. Point cloud reconstructions of the ETH3D [7] sequence `Meadow` provided by DELS-MVS (left column), EPP-MVSNet [5] (central column) and PatchMatchNet [9] (right column). The top row depicts the colored point clouds. The central row depicts the point cloud completeness: green points are complete, red are incomplete. The bottom row depicts the point cloud accuracy: green points are accurate, red are inaccurate and blue not observed. The depicted completeness and accuracy consider a $2cm$ error threshold.

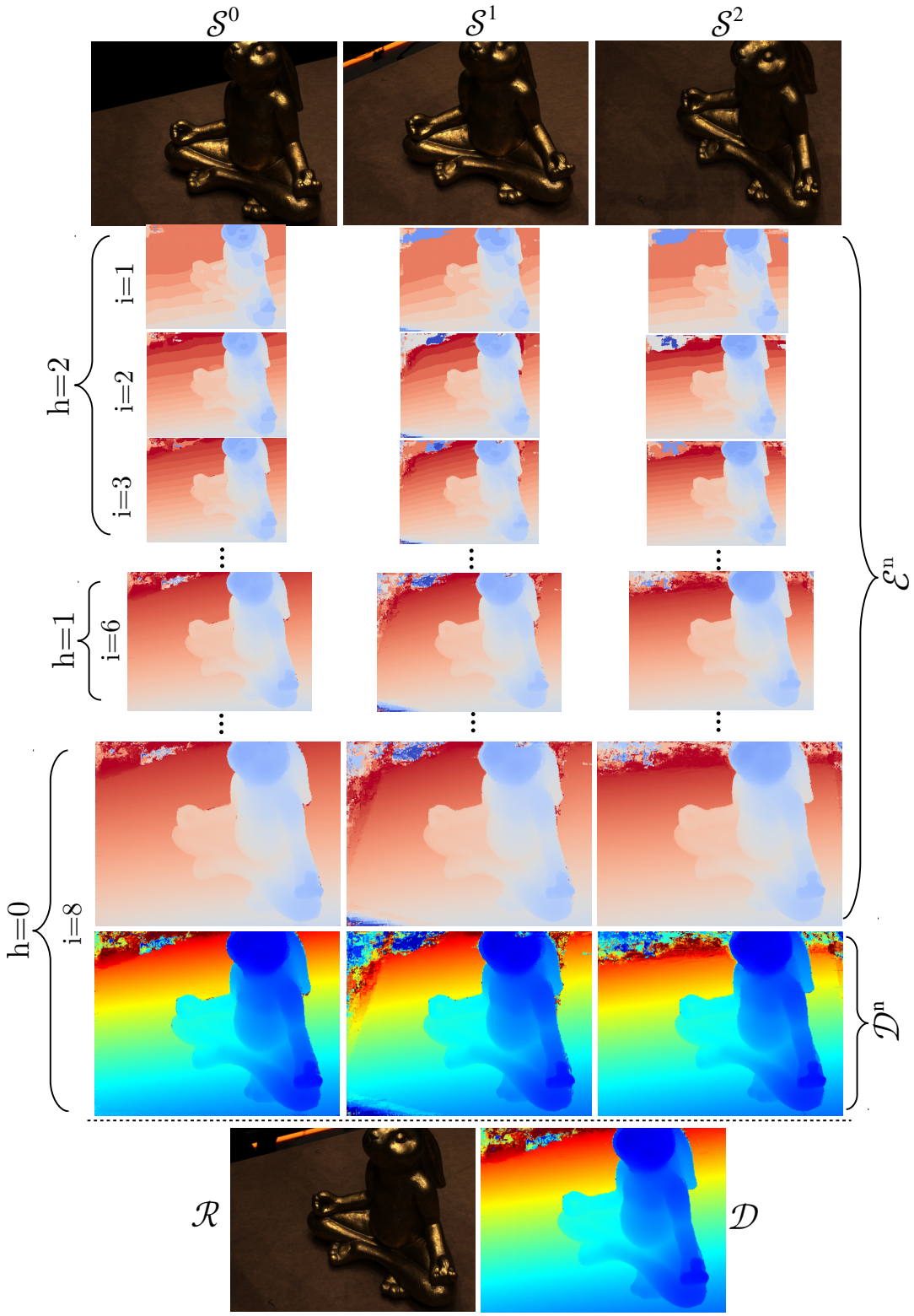

Figure 4. DELS-MVS results for multiple iterations on a DTU [1] sequence, setting the number of utilized source images to $N = 3$. The top row depicts the three considered source images $\mathcal{S}^n$. The rows one to five depict the predicted epipolar residual maps $\mathcal{E}^n$ at different iterations i of the three resolution levels h. Positive residual values are in blue, negative ones are in red. The three final epipolar residual maps at row five are converted into the three depth maps $\mathcal{D}^n$, depicted at row six. Finally, the bottom row depicts the considered reference image $\mathcal{R}$ and its final depth map $\mathcal{D}$ obtained by fusing the three depth maps $\mathcal{D}^n$.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, pages 1–16, 2016.

[2] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[3] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017.

[4] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] Xinjun Ma, Yue Gong, Qirui Wang, Jingwei Huang, Lei Chen, and Fan Yu. Epp-mvsnet: Epipolar-assembling based depth prediction for multi-view stereo. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[6] Z. Mi, C. Di, and D. Xu. Generalized binary search network for highly-efficient multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[7] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[8] Christian Sormann, Mattia Rossi, Andreas Kuhn, and Friedrich Fraundorfer. Ib-mvs: An iterative algorithm for deep multi-view stereo based on binary decisions. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2021.

[9] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[10] Jiayu Yang, Wei Mao, Jose M. Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.