Event-aided Direct Sparse Odometry
– Supplementary Material –

Javier Hidalgo-Carrió[1], Guillermo Gallego[2], Davide Scaramuzza[1]
[1]Dept. of Informatics, Univ. of Zurich and Dept. of Neuroinformatics, Univ. of Zurich and ETH Zurich.
[2]Technische Universitt Berlin, Einstein Center Digital Future and SCIoI Excellence Cluster, Germany.

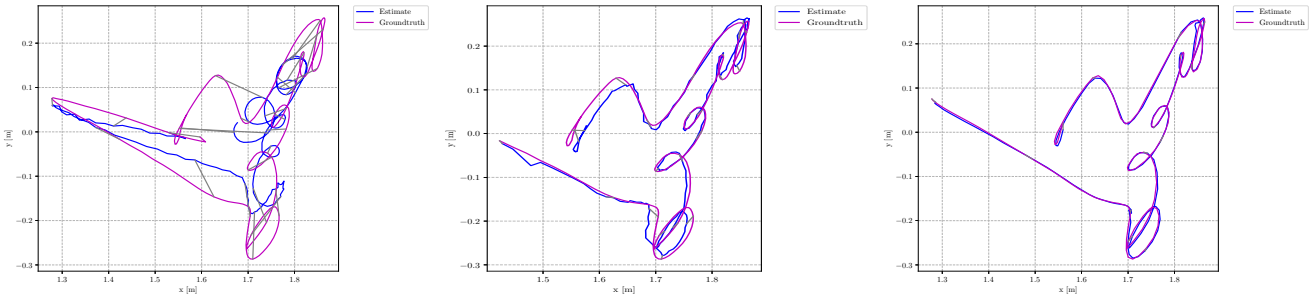

Figure 1. DSO (left), ORB-SLAM (center) and EDS (right) camera trajectory for sequence *desk* in the dataset from [1] at 20 FPS.

	20 FPS				10 FPS				7 FPS				5 FPS			
	DSO*	DSO	ORB_SLAM*	EDS (**ours**)	DSO*	DSO	ORB_SLAM*	EDS(**ours**)	DSO*	DSO	ORB_SLAM*	EDS(**ours**)	DSO*	DSO	ORB_SLAM*	EDS(**ours**)
Input	F	F	F	E+F	F	F	F	E+F	F	F	F	E+F	F	F	F	E+F
bin	1.1	1.2	2.4	**1.1**	1.8	1.8	2.4	**1.8**	3.5	2.5	**2.4**	2.5	16.9	4.8	**2.5**	2.6
boxes	**2.0**	2.0	3.9	2.1	13.5	**3.1**	3.9	3.8	14.8	14.6	**3.9**	5.0	14.8	14.6	7.0	**5.8**
desk	10.0	10.0	3.8	**1.5**	13.4	9.1	3.8	**3.4**	21.1	16.2	7.8	**4.7**	21.6	19.2	9.3	**5.0**
monitor	**0.9**	0.9	3.1	1.0	3.9	**1.5**	10.6	2.3	26.5	12.1	10.9	**2.5**	28.0	27.1	10.3	**8.0**

Table 1. Performance at different frame rates in terms of Absolute Trajectory Error (RMS) [t: cm]. Data from [1].

Overview

In this supplementary material we present:

- Additional details on the low frame rate experiments (Sec. 1).
- Qualitative results about the sensitivity study in depth inaccuracies (Sec. 2).
- Our beamsplitter device and two additional experiments recorded with it (Sec. 3).
- A novel dataset with high quality events & RGB frames recorded with the beamsplitter device to foster research on the topic (Sec. 4).
- A more thorough discussion of limitations (Sec. 5).

1. Low Frame Rate Experiments

Table 1 shows the numerical values in the low frame rate experiment, used for plotting Fig. 5 in the main paper. The values shows that EDS performs the best when the number of frames decreases. Direct methods are less accurate than indirect methods at a lower frame rate (FPS). However at a higher frame rate direct methods use more information from the scene; achieving a higher accuracy.

Among all sequences in dataset [1], *desk* has the most challenging camera motions. In this scenario events excel and help standard frame-based methods. This is also shown in the results (see Tab. 1) where EDS outperforms previous frame-based methods at any given frame rate. Figure 1 shows the trajectories of DSO, ORB-SLAM and EDS for the *desk* sequence. It is qualitatively visible how events enhance standard direct methods guiding the camera pose and producing a more accurate trajectory.

2. Sensitivity with respect to Depth Noise

Depth estimation is the main limitation of our EDS method. This is because events are only used for camera tracking (front-end) and the EGM highly depends on optical

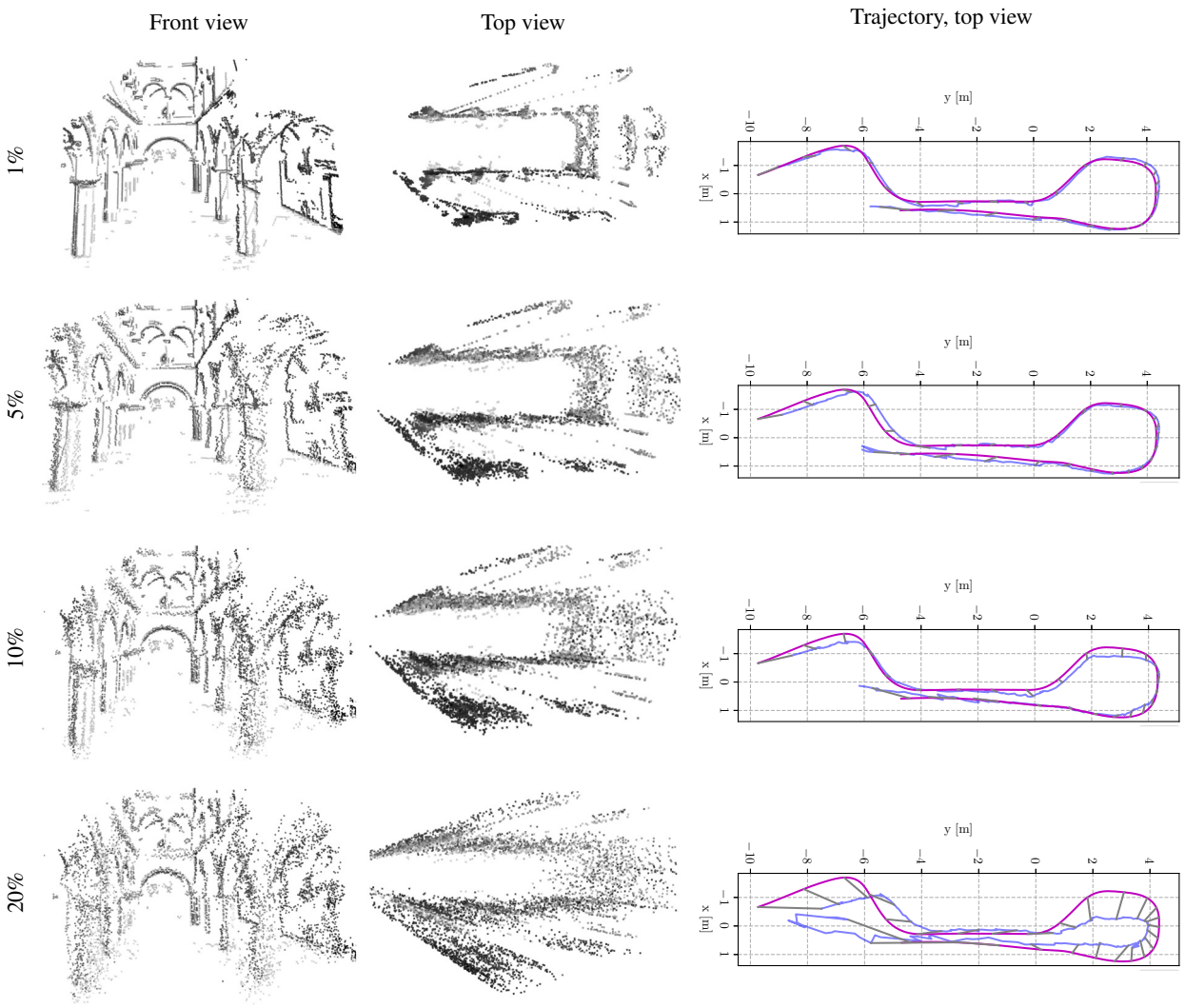

Figure 2. Sensitivity with respect to increasing noise in the depth of the 3D points, from 1% to 20% of the median scene depth. Atrium scene. The ground truth trajectory is in purple, while the estimated trajectory is in blue.

flow, which is a function of depth. We previously presented how depth noise affects the camera tracker and entails camera pose inaccuracies. Figure 2 shows the Atrium sequences with the perturbed map. We might conclude that up to 10 % standard deviation of the median depth in the scene, events are still useful to track the camera motion. However, when the mapper wrongfully estimates depth points, events do not help the tracker, generating more inaccuracies than if those events were not used. Therefore, it is important to correctly select which points to incorporate in the EGM calculation.

3. Beamsplitter and Additional Experiments

Publicly available event-based Visual Odometry (VO) datasets are limited and/or contain low quality sensory data [1,2], either because of noisy events recorded with pioneer neuromorphic devices (such as the DAVIS240B [3] or DAVIS346B) or due to low quality frames (i.e., no gamma correction, small fill factor). Recent publicly available datasets, such as [4, 5], are equipped with newer sensors, however they do not share the same optical axis, having frames and events in two image planes, standard and event-based camera separately. There is a clear need for up-to-date datasets in the event-based VO community with state-of-the-art sensors. We now evaluate our monocular VO method in an extended dataset with our custom co-capture device. First, we present our custom-made beamsplitter device and the calibration procedure in brief (Sec. 3.1). Second, we evaluate the performance of the method (Sec. 3.2).

3.1. Beamsplitter: Sensors and Calibration

We build a custom-made sensing device consisting of a Prophesee Gen3 event camera [6] (640×480 px resolution)

Figure 3. Calibration. Frames from the standard FLIR camera (top) and grayscale frames reconstructed from events (bottom), both with checkerboard corners used during calibration.

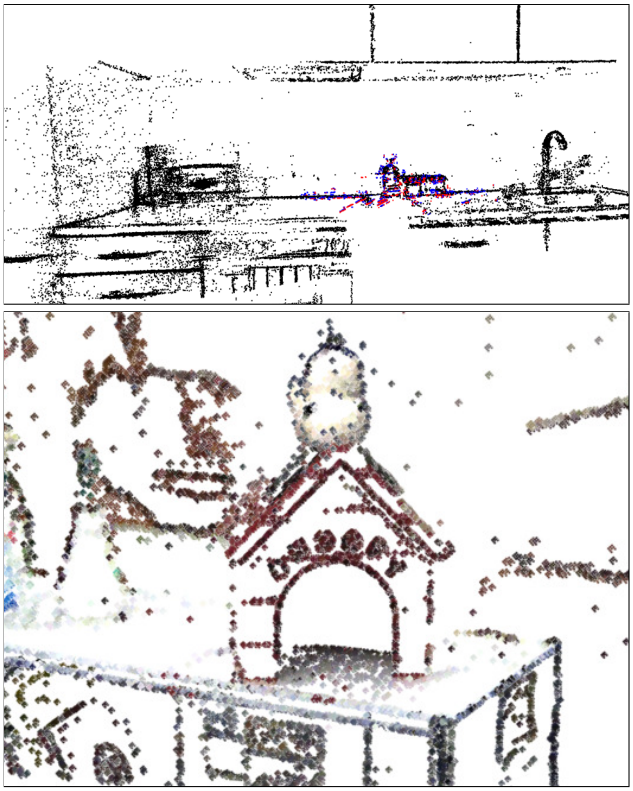

Figure 4. Point cloud maps of the kitchen (top) and snoopy (bottom) sequences.

and a color FLIR camera viewing the same scene through a beamsplitter. The Snoopy house scene in Fig. 1 in the main paper is recorded with such a device (see Fig. 6 and Tab. 3 for details).

Both cameras are calibrated and their outputs are aligned with sub-pixel accuracy, giving the equivalent of a DAVIS camera with higher quality color frames at VGA (640×480 px) resolution. To calibrate the cameras, we reconstruct grayscale frames from events using e2calib [7] and input them into Kalibr [8], which computes the intrinsic and extrinsic camera parameters. Figure 3 depicts individual camera outputs during the calibration process, and Fig. 5 shows the rectified and undistorted image output with events.

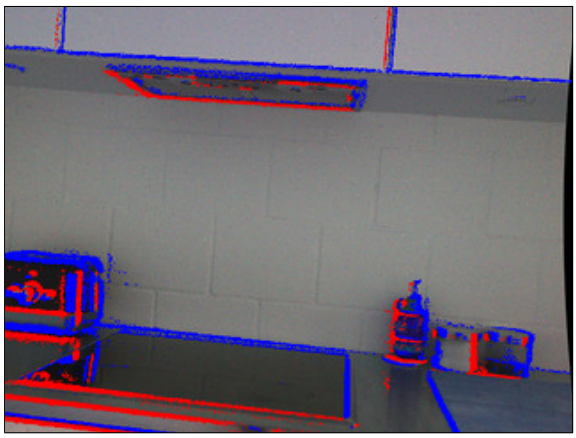

Figure 5. Beamsplitter output with aligned events and frames for the *kitchen* sequence. This figure contains animations that can be viewed in *Acrobat Reader*.

3.2. Ego-motion Estimation Results

We recorded two sequences with our beamsplitter device to prove generalization of our method to other (and newer) visual sensory data. The sequences were recorded in natural indoor scenes and we used COLMAP [9] to provide a ground truth trajectory, since no external motion-capture system (i.e., Vicon or Optitrack) was not available. Figure 4 shows the reconstructed 3D map for EDS in the *kitchen* and *snoopy* sequences using events and frames. The colored points in the central part of the kitchen correspond to the 3D locations which are active (i.e., generating events using the event generation model - EGM) in the current keyframe. *snoopy* points cloud used RGB color for a more appealing visualization of the Snoopy reconstructed house.

Quantitatively (Tab. 2), our method outperforms all other monocular pure VO baseline methods in the *kitchen* sequence and achieves state-of-the-art accuracy in the *snoopy* sequence. We did not include comparison with USLAM [13] in Tab. 2 since it requires an IMU, which is not available in these two sequences. The beamsplitter device (see Sec. 3.2) has a better sensor quality than the DAVIS240C in [1]. It produces events with less noise,

	ORB-SLAM [10]	EVO [11]	DSO [12]	EDS (**Ours**)
Input	F	E	F	E+F
kitchen	13.0±9.7	-	12.5±6.8	**9.6±5.5**
snoopy	35.1±28	-	**30.7±14**	30.9±13

Table 2. Comparison with state-of-the-art 6-DOF VO methods in terms of Absolute Trajectory Error (RMS) [t: cm] and its standard deviation on two beamsplitter sequences. Ground truth poses are computed using COLMAP. Input data may be: events (E) and/or grayscale frames (F). EVO entries marked with hyphen indicate the code did not manage to recover poses on these sequences.

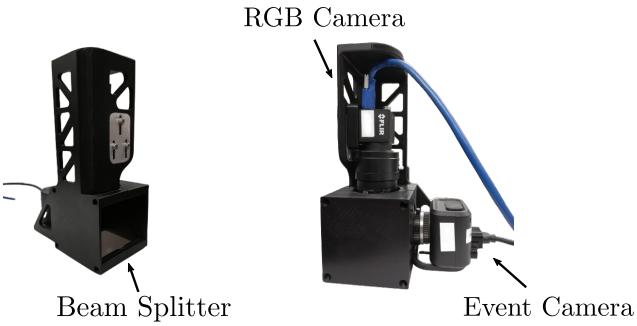

Figure 6. Beam-splitter with an event and a standard camera.

which makes the EGM more accurate (see multimedia material for details).

4. The EDS Dataset

We realized the existing gap in good quality monocular visual-inertial odometry (VIO) dataset with events. Therefore we built a beamsplitter device during the last part of our investigation. The aim of the beamsplitter was to record and release a new dataset to promote and facilitate research on the topic. The dataset targets VIO but it may be used to demonstrate other tasks, such as optical flow estimation, depth estimation, view synthesis (e.g., NeRFs), etc. Figure 7 shows snapshots of the recorded sequences with our beamsplitter device. When available, we also record ground truth poses from a motion-capture system.

In a nutshell, the EDS dataset provides synchronized events with RGB frames, IMU and ground truth data. The data is given in three different formats, pocolog[1], rosbag[2] and archive files with compressed events in HDF5 format. The images are timestamped, with exposure time and gain values, and the ground truth poses are at the camera frame (i.e., $T_{\text{marker\_cam}}$ already applied), to facilitate its use. The motion capture system is Vicon or the Optitrack depending on the scene. The long sequences such as *04_floor_loop* and *12_floor_eight_loop* provide ground truth at the start and end locations. *15_apartment_day* gives start and finish positions using an Apriltag[3] marker. In addition, the calibration results to align frames and events as well as the camera to IMU transformation are given (see the multimedia material for further details).

5. Limitations

5.1. Grayscale Frames are not HDR

Our method works under the assumption of the availability of collocated grayscale frames and events. Current devices, such as the DAVIS camera [3], produce low-quality grayscale frames, with a dynamic range of $\approx 55\,\text{dB}$, which is small compared to the high dynamic range (HDR) properties of event cameras ($>120\,\text{dB}$). Hence, the frames from the DAVIS are not HDR, and one could point this as a limitation of the approach.

We tackled this problem in two ways: (i) by building our own sensing device with higher quality frames than the DAVIS (Fig. 6), and (ii) by testing alternatives, such as using grayscale frames reconstructed from the events (e.g., using state of the art [14]). The latter was tested on DSO and our approach. The results from DSO are reported in the main paper, indicated with DSO† in Tab. 3, and we observe that it did not produce as good results as DSO on regular frames (it even failed in bin and boxes sequences). We observed the same failure effect when using reconstructed grayscale frames on EDS. Nevertheless, we expect that if using a higher-end device (e.g., (i)) is not an option, better grayscale frames for VO with our method would be obtained in the near future by means of improvements in image reconstruction methods and/or event cameras (i.e., lower event noise). Event cameras are evolving fast, and new prototypes, in combination with standard sensors may occur in the near future, for example to advance computational photography (e.g., [15]).

Sensor Type	**Description**
Prophesee Gen 3.1	Prophesee PPS3MVCD event camera 640 × 480 pixels, 3/4 CMOS Monochrome ≥120 dB dynamic range
FLIR Blackfly S USB3	FLIR BFS-U3-16S2C-CS 640 480 pixels, 1/2.9 CMOS Color 71.4 dB dynamic range uoto 75 Hz frame rate (depending on sequebce)
Inversense MPU-9250	Inertial Measurement Unit MEMS, 16bits resolution 3x Gyroscopes 3x Accelerometers 3x Magnetometers (not utilized) 1000Hz sampling rate
Motion-capture system	Optitrack: RPG Flying room, 11 cameras Vicon: RPG Drone Arena, 36 cameras Camera pose: position + rotation 150 Hz sampling rate

Table 3. Details of the hardware utlized for the dataset collection

5.2. Computational Performance

The current implementation of our method is unoptimized and it is about $5\times$ slower than real time. However, we think that there is large room for code improvement and engineering to make it real-time capable. Specifically, the back-end is un-optimized since automatic differentiation in Ceres [16] is not real time when having a large number of parameters (i.e., 4000 points in a 7-keyframe sliding window). It could be sped up by feeding fewer points to the back-end and better selecting a sparse set of points. This is the reason why we combined our front-end with DSO's [12]

[1] https://www.rock-robotics.org
[2] http://wiki.ros.org/rosbag
[3] https://april.eecs.umich.edu/software/apriltag

Figure 7. *Sequences from our EDS Dataset*: the dataset contains 16 sequences with events, color frames, IMU data and ground truth poses on a diverse set of environments.

back-end, which is optimized and real-time capable with up to 2000 points in a similar sliding-window size. The main purpose of our work is to understand the limitations of previous event-based methods (e.g., why do many of the prior works lose track or have large errors?) and overcome them with a new design that has not been previously explored. Finally, if the method is used offline, e.g., for recovering a scene map and/or accurate camera trajectory, runtime is not an issue.

5.3. Setting an Arbitrary Scale for the World

Absolute scale is not observable in monocular VO without additional information or an IMU. Hence, in our method we need to provide values to set the depth range of the initial map, which will guide the rest of the visual odometry process. We did not consider using an IMU (accelerometer and gyroscope) to focus on the visual aspects of odometry. Sensor fusion with an IMU is left as future work.

References

[1] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza, "Semi-dense 3D reconstruction with a stereo event camera," in *Eur. Conf. Comput. Vis. (ECCV)*, pp. 242–258, 2018. 1, 2, 3

[2] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza, "The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM," *Int. J. Robot. Research*, vol. 36, no. 2, pp. 142–149, 2017. 2

[3] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck, "A 240x180 130dB 3μs latency global shutter spatiotemporal vision sensor," *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014. 2, 4

[4] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza, "DSEC: A stereo event camera dataset for driving scenarios," *IEEE Robot. Autom. Lett.*, 2021. 2

[5] Simon Klenk, Jason Chui, Nikolaus Demmel, and Daniel Cremers, "TUM-VIE: The TUM stereo visual-inertial event

dataset," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2021. 2

[6] Prophesee Evaluation Kits. https://www.prophesee.ai/event-based-evk/, 2020. 2

[7] Manasi Muglikar, Mathias Gehrig, Daniel Gehrig, and Davide Scaramuzza, "How to calibrate your event camera," in *IEEE Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW)*, 2021. 3

[8] Joern Rehder, Janosch Nikolic, Thomas Schneider, Timo Hinzmann, and Roland Siegwart, "Extending kalibr: Calibrating the extrinsics of multiple imus and of individual axes," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 4304–4311, 2016. 3

[9] Johannes Lutz Schönberger and Jan-Michael Frahm, "Structure-from-motion revisited," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2016. 3

[10] Raúl Mur-Artal and Juan D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, pp. 1255–1262, Oct. 2017. 3

[11] Henri Rebecq, Timo Horstschäfer, Guillermo Gallego, and Davide Scaramuzza, "EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real-time," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 593–600, 2017. 3

[12] Jakob Engel, Vladlen Koltun, and Daniel Cremers, "Direct Sparse Odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, pp. 611–625, Mar. 2018. 3, 4

[13] Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, and Davide Scaramuzza, "Ultimate SLAM? combining events, images, and IMU for robust visual SLAM in HDR and high speed scenarios," *IEEE Robot. Autom. Lett.*, vol. 3, pp. 994–1001, Apr. 2018. 3

[14] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019. 4

[15] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza, "TimeLens: Event-based video frame interpolation," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2021. 4

[16] A. Agarwal, K. Mierle, and Others, "Ceres solver." http://ceres-solver.org. 4