Supplementary Material

Abstract

In this supplementary document we provide additional implementation details, including the detailed network architecture, and data preparation/training details. We also provide additional results and discussion on predicted global trajectories.

1. Network Architecture

The implementation details for the architecture that was used in our experiments are reported in Table 1. We use the following abbreviations:
Conv: convolution, **FC**: fully-connected, **activ.**: activation function, $\oplus$: concatenation; $\odot$: element-wise multiplication; B: batch size.

The visual encoder follows the encoder structure of the FlowNetS architecture [3]. We initialize the visual encoder weights with the weights of a model that was pre-trained on the FlyingChairs dataset[1], since training from scratch would require larger amounts of data compared to our dataset sizes. For this reason, training the encoder from scratch experimentally gave worse results.

Bi-directional LSTMs were used for the inertial encoder (as in [2]) as well as in the recurrent part of the pose regressor. Bi-directional RNNs provide higher data efficiency when dealing with small datasets. The LSTM layers include dropout regularization on the recurrent connections.

2. Dataset Preprocessing Details

Few publicly available datasets provide camera images, high-frequency IMU and ground-truth for training visual inertial odometry (i.e., inertial data is missing in the Oxford Robotcar Dataset [5] and 7-Scenes Dataset [9]; ground-truth for the full trajectories is not available in TUM VIO [8]). Therefore, we use the KITTI [4], EuRoC [1], and PennCOSYVIO [7] datasets for training and evaluation.

KITTI High-frequency inertial data (100Hz) is only available in the raw unsynced data packages. Hence, we manually synchronize inertial data and images according to their timestamps. We used Sequences *00, 01, 02, 04, 06, 08, 09* for training and tested the network on Sequences *05*, *07*, and *10*, excluding sequence 03 as the corresponding raw file is unavailable. Correspondences between the KITTI Odometry Number, the raw data file names and the corresponding number of sequences are reported in Table 2. The images are resized to 512×256.

EuRoC For the EuRoC MAV dataset [1] we use grayscale video images from CAM1 at 20fps of the VI-Sensor and the IMU measurements from the same sensors at 200Hz. The data is tightly synchronized. Doubling the sensor rates compared to the ones in KITTI helps to deal with the faster, jerky MAV movements. As with KITTI, images are resized to 512×256. We train on all sequences, minus *MH_04_difficult*, which is used for testing.

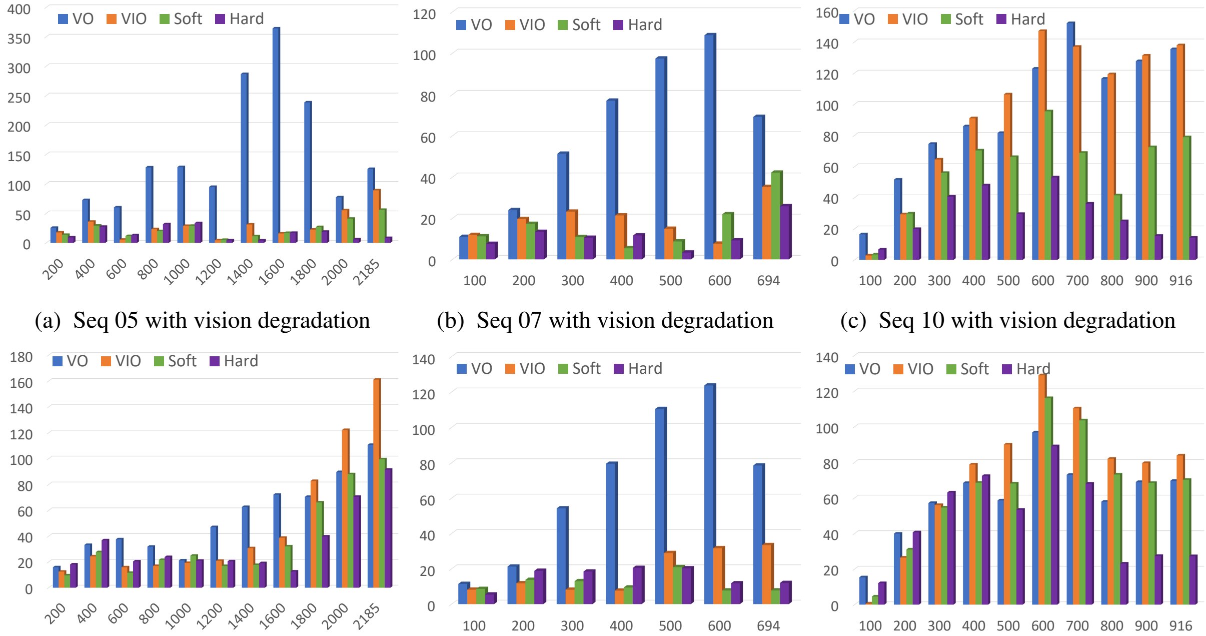
PennCOSYVIO A number of sensors are included in the PennCOSYVIO dataset [7]. For our experiment we select the video images from the Tango Bottom camera, running at 30fps, and inertial measurements from the VI-Sensor, running at 200Hz. Images are subsampled to 10Hz, and IMU measurements are subsampled to 100Hz, similarly to KITTI. Images are cropped and resized to 512×256. We train on sequences *as, bf, bs* and test on sequence *af*. Small time-synchronization errors between the Tango camera and the VI-Sensor are likely present in the data, leading to worse results in all scenarios.

3. Evaluation of global trajectories on KITTI

Figure 1 shows the global RMSE position errors on the three test KITTI trajectories (Seq 05, Seq 07, Seq 10) as a function of travelled distance, for both the normal dataset and the fully degraded dataset (visual degradation + sensor degradation). We compare the two VO and vanilla VIO baselines with the proposed soft and hard fusion strategies. It can be noticed how, while at start VIO performs as well as soft and hard fusion, on average, over time the proposed selective fusion strategies outperform the vanilla fusion, since the increased robustness reduces error accumulation. This is particularly visible in the most challenging Seq 05. As expected, a VO approach heavily underperforms in presence of large amounts of angular rotations (Seq 05, Seq 07).

Another interesting result is how VO performs slightly better in presence of IMU degradation and camera-IMU

[1] https://lmb.informatik.uni-freiburg.de/resources/datasets/FlyingChairs.en.html



(a) Seq 05 with vision degradation
(b) Seq 07 with vision degradation
(c) Seq 10 with vision degradation

(d) Seq 05 with full degradation
(e) Seq 07 with full degradation
(f) Seq 10 with full degradation

Figure 1: Global position errors (in meters, Y axis) on the KITTI dataset, over travelled distance (in meters, X axis).

synchronization (Figure 1, bottom row). That shows how a vanilla VIO fusion is unable to deal with these issues, to the point of underperforming compared to vision-only approaches. This result further corroborates the fact that in deep learning-based approaches explicitly learning the belief on the different components makes the estimation more robust, while stacking sensors without a sensible fusion strategy can lead to *catastrophic fusion*, similarly to traditional approaches. Catastrophic fusion happens when the single components of the system before fusion significantly outperform the overall system after fusion [6].

4. Supplementary Video

The supplementary video shows an evaluation of soft fusion and hard fusion on Seq 05 of the KITTI dataset, with vision degradation (10% occlusion, 10% blur+noise, 10% missing data), compared with two deep VO and VIO baselines. The degraded images and corresponding soft/hard masks are shown in the top-left and bottom-left respectively. The trajectories from the four methods are shown on the right side.

References

[1] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart. The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research*, 2016. 1

[2] C. Chen, C. X. Lu, A. Markham, and N. Trigoni. Ionet: Learning to cure the curse of drift in inertial odometry. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 1

[3] P. Fischer, E. Ilg, H. Philip, C. Hazrbas, P. V. D. Smagt, D. Cremers, and T. Brox. FlowNet: Learning Optical Flow with Convolutional Networks. In *International Conference on Computer Vision, ICCV*, 2015. 1

[4] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 1

[5] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2016. 1

[6] J. R. Movellan and P. Mineiro. Robust sensor fusion: Analysis and application to audio visual speech recognition. *Machine Learning*, 32(2):85–100, 1998. 2

[7] B. Pfrommer, N. Sanket, K. Daniilidis, and J. Cleveland. Pencosyvio: A challenging visual inertial odometry benchmark. In *2017 IEEE International Conference on Robotics and Automation, ICRA 2017, Singapore, Singapore, May 29 - June 3, 2017*, pages 3847–3854, 2017. 1

[8] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers. The TUM VI Benchmark for Evaluating Visual-Inertial Odometry. In *ICRA*, 2018. 1

[9] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *CVPR*, pages 2930–2937, 2013. 1

Visual Encoder
[input] Two stacked images: $B \times 512 \times 256 \times 6$
[layer 1] Conv. 7^2, Stride 2^2, Padding 3, LeakyReLU activ.
[layer 2] Conv. 5^2, Stride 2^2, Padding 2, LeakyReLU activ.
[layer 3] Conv. 5^2, Stride 2^2, Padding 2, LeakyReLU activ.
[layer 3_1] Conv. 3^2, Stride 1^2, Padding 1
[layer 4] Conv. 3^2, Stride 2^2, Padding 2, LeakyReLU activ.
[layer 4_1] Conv. 3^2, Stride 1^2, Padding 1
[layer 5] Conv. 3^2, Stride 2^2, Padding 2, LeakyReLU activ.
[layer 5_1] Conv. 3^2, Stride 1^2, Padding 1
[layer 6] Conv. 3^2, Stride 2^2, Padding 1
[layer 6 output] $B \times 4 \times 8 \times 1024$
[layer 7] FC 256
[output] $B \times 256$

Inertial Encoder
[input] IMU sequence: $B \times 10 \times 6$
[layer 1] FC 128
[layer 2] B-LSTM, 2-layers, hidden size 128, Dropout 0.2
[output] $B \times 256$

Direct Fusion Module
[input] $B \times (256 \oplus 256)$
[output] $B \times 512$

Soft Fusion Module
[input] $B \times (256 \oplus 256)$
[layer 1] FC 512 (input)
[layer 2] [(input) $\odot$ (layer 1)
[output] $B \times 512$

Hard Fusion Module
[input] $B \times (256 \oplus 256)$
[layer 1] FC 1024, Sigmoid activ. (input)
[layer 2] Gumbel-Softmax sampling, 512×2
[layer 2] [(input) $\odot$ (layer 2)
[output] $B \times 512$

Pose Regressor
[input] $B \times 512$
[layer 1] B-LSTM, 2-layers, hidden size 512, Dropout 0.2
[layer 2] Dropout 0.2
[layer 3_1] FC 3 (layer 2)
[layer 3_2] FC 3 (layer 2)
[layer 4] (layer 3_1) $\oplus$ (layer 3_2)
[output] $B \times 6$

Table 1: Implementation details for the proposed architecture. $\oplus$ denotes a concatenation operation; $\odot$ indicates an element-wise product between two tensors.

Table 2: Correspondences between the KITTI Odometry Number, the raw data file names and the corresponding number of sequences

Odometry Nr.	Raw Data Filename	Number of Sequences
00	2011_10_03_drive_0027	4390
01	2011_10_03_drive_0042	1173
02	2011_10_03_drive_0034	4485
03	2011_09_26_drive_0067	Not available
04	2011_09_30_drive_0016	284
05	2011_09_30_drive_0018	2750
06	2011_09_30_drive_0020	1091
07	2011_09_30_drive_0027	1111
08	2011_09_30_drive_0028	5149
09	2011_09_30_drive_0033	1599
10	2011_09_30_drive_0034	1215