DROID-SLAM

Witold Marek Pacholarz

Department of Informatics - Technische Universität München

Abstract

Zachary Teed and Jia Deng in their paper *DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras* [9], presented at NeurIPS 2021, introduce a fully differentiable end-to-end deep learning-based Visual SLAM system. The model based on optical flow estimation iteratively corrects pixelwise depth and camera pose estimations. Evaluation results on monocular, stereo and RGB-D sequences indicate significant performance improvement over both classical and deep learning-based approaches, although the system have been trained on just one monocular dataset.

1 Introduction

The goal of Visual SLAM is to estimate the pose of the agent and create a 3D map of the environment at the same time [9], relying only on image sequences. Over many years multiple classical approaches leveraging Bundle Adjustment have been proposed [3, 4]. Such techniques, however, suffered from frequent failures, making their robustness insufficient for real-world scenarios. More recently, there has been many trials of augmenting classical methods by learned components [2, 1]. Additionally, deep learning-based end-to-end architectures have been introduced [6, 7]. In spite of significant robustness improvement, they provided lower accuracy and poor generalization.

The Differentiable Recurrent Optimization-Inspired Design, proposed in the paper, combines advantages of both traditional and deep learning methods. It builds upon SOTA optical flow estimation RAFT [8] and uses a Dense Bundle Adjustment layer to iteratively update camera poses and pixelwise depths. Finally, global bundle adjustment is applied to refine the results.

One of the closest deep architectures, BA-Net, focuses on depth prediciton and delivers limited SLAM capabilities [6]. In contrary to BA-Net, which optimizes photometric error, DROID-SLAM optimizes geometric error, ensuring smoother energy function. Additionally, the DROID-SLAM bundle adjustment optimizes for pixelwise depth updates directly which eliminates the need for a learned depth basis used in BA-Net and hence, improves generalization across datasets. Another system trained end-to-end is DeepTAM [11]. While focusing on pose tracking and dense mapping, it lacks global bundle adjustment and loop closure. For this reason, it is not suitable for large-scale reconstructions, unlike DROID-SLAM.

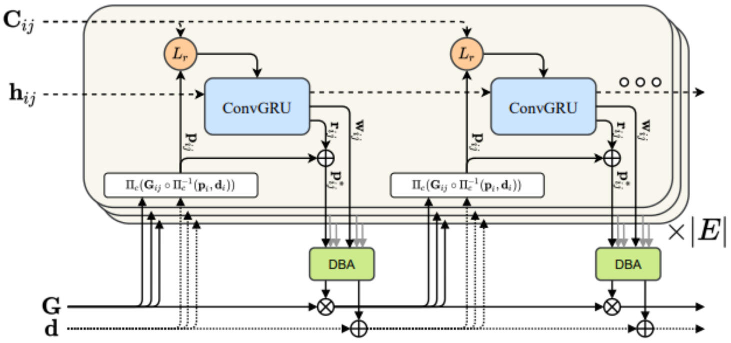


Figure 1: Architecture of an update operator

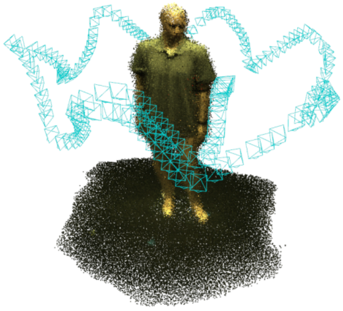


Figure 2: Reconstruction

2 Method description

Notation: The network is provided with a sequence of images $\{\mathbf{I}_t\}_{t=0}^{N}$. Its goal is to estimate a set of camera poses $\mathbf{G}_t \in SE(3)$ and inverse depths $\mathbf{d}_t \in \mathbb{R}_+^{H\times W}$ (for simplicity, in the report inverse depths are referenced as depths). The co-visibility between frames is encoded by a frame-graph (V, E) and updated on the fly during training. Edges $(i, j) \in E$ indicate that there exists sufficient flow overlap between images $\mathbf{I}_i$ and $\mathbf{I}_j$.

2.1 Update Operator

An update operator (Fig. 1) forms the main component of the DROID-SLAM. It is a convolutional GRU with hidden state $\mathbf{h}_{ij}$ and input formed by concatenating context, correlation and correspondence field $\mathbf{p}_{ij}$.
The outputted hidden state is processed by two neural networks in order to get confidence weights $\mathbf{w}_{ij}$ and revision flow field $\mathbf{r}_{ij}$ which is used to correct the inputted correspondence field: $\mathbf{p}^*_{ij} = \mathbf{r}_{ij} + \mathbf{p}_{ij}$. The process is repeated for all the (i, j) pairs in the frame graph. The corrected flow fields are then passed together with the estimated confidence weights to the Dense Bundle Adjustment Layer.

Context: Global context is provided as the average of the hidden state over spatial image dimensions. This allows handling incorrect correspondences which might be a result of, e.g., large moving objects.
Correlation: To get correlation, first, feature encoder networks are used to extract features for pairs of images (i, j). Then, 4D correlation volume $\mathbf{C}_{ij}$ is computed as a dot product of all pairs of extracted feature vectors from two images. By average pooling over last two dimensions, 4-level correlation pyramid is created. Afterwards, correlation lookup is performed. Current correspondence field $\mathbf{p}_{ij}$ is used to map each pixel $\mathbf{x}$ in $\mathbf{I}_i$ to its estimated correspondence $\mathbf{x}'$ in $\mathbf{I}_j$. Correlation values are then retrieved from the correlation volume for each pixel in the local grid around $\mathbf{x}'$ and the process is repeated on each level of the correlation pyramid. Finally, values from each level are concatenated into a single feature map.

2.2 Dense Bundle Adjustment Layer

The Dense Bundle Adjustment Layer estimates pose and pixelwise depth updates. The optimization objective

$$\mathbf{E}(\mathbf{G}', \mathbf{d}') = \sum_{(i,j)\in\mathcal{E}} ||\mathbf{p}^*_{ij} - \Pi_c(\mathbf{G}_{ij}^{-1} \circ \Pi_c^{-1}(\mathbf{p}_i, \mathbf{d}_i))||^2_{\Sigma_{ij}} \qquad \Sigma_{ij} = diag\mathbf{w}_{ij} \tag{1}$$

is defined using Mahalanobis distance where residuals are weighted by estimated correspondence weights. The nonlinear problem is solved using the Gauss-Newton algorithm. To take advantage of the block diagonal structure of the Hessian matrix, the Schur complement is used. Finally, the camera poses and depths become updated

$$\mathbf{G}^{(k+1)} = \mathrm{Exp}(\Delta\xi^{(k)}) \circ \mathbf{G}^{(k)} \qquad \mathbf{d}^{(k+1)} = \Delta\mathbf{d}^{(k)} + \mathbf{d}^{(k)} \tag{2}$$

and backpropagation through the layer is performed.
In order to adjust the model for RGB-D sequences, additional term penalizing $L2$ distance between the predicted and measured depth is added to the cost function in Eq. 1. Similarly, in the case of the stereo video, the relative pose between the two cameras needs to be fixed.

2.3 Global Bundle Adjustment and Network Loss

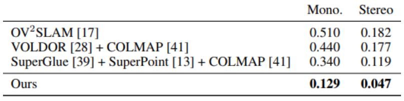
Global Bundle Adjustment runs on a separate GPU, over the whole history of keyframes. It is responsible for refining the results and performng loop closure.

The network loss is composed of flow loss and pose loss. The flow loss is calculated for the flow fields of adjacent frames as the average $L2$ distance, whereas the pose loss is the distance between the predicted and grоung truth poses:

$$L_{pose} = \sum_i ||Log_{SE3}(\mathbf{T}_i^{-1} * \mathbf{G}_i)||. \tag{3}$$

3 Experiments and results

The system was trained just on a synthetic monocular TartanAir dataset. It was then evaluated without retraining on a few banchmarks using three different sensor modalities: monocular, stereo and RGB-D. Absolute Trajectory Error (ATE) [5] was used for comparison with other methods.



	Mono.	Stereo
OV^{2}SLAM [17]	0.510	0.182
VOLDOR [28] + COLMAP [41]	0.440	0.177
SuperGlue [39] + SuperPoint [13] + COLMAP [41]	0.340	0.119
Ours	**0.129**	**0.047**

Figure 3: TartanAir

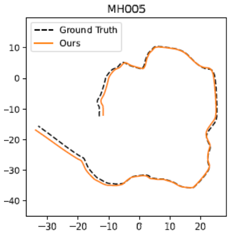
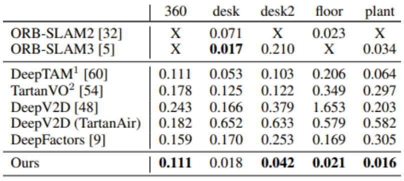


Figure 4: Drift



	360	desk	desk2	floor	plant
ORB-SLAM2 [32]	X	0.071	X	0.023	X
ORB-SLAM3 [5]	X	**0.017**	0.210	X	0.034
DeepTAM^1 [60]	0.111	0.053	0.103	0.206	0.064
TartanVO^2 [54]	0.178	0.125	0.122	0.349	0.297
DeepV2D [48]	0.243	0.166	0.379	1.653	0.203
DeepV2D (TartanAir)	0.182	0.652	0.633	0.579	0.582
DeepFactors [9]	0.159	0.170	0.253	0.169	0.305
Ours	**0.111**	0.018	**0.042**	**0.021**	**0.016**

Figure 5: TUM-RGBD

Fig. 3 presents qualitative comparison with other ECCV 2020 SLAM competition submissions, tested on the monocular and stereo TartanAir benchmark. The system reduces the error by around 60% in both cases. However, it is worth noting that, because of lack of loop closure in sequence MH005 (as seen in Fig. 4), the error for monocular video is much higher than in the case of sequences where loop closure is ensured.

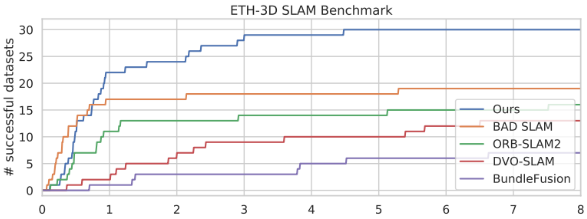


Figure 6: ETH3D-SLAM

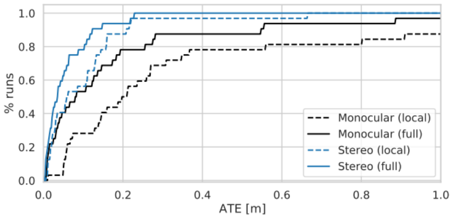


Figure 7: Ablation study

Results achieved on TUM-RGBD benchmark depicted in Tab. 5 indicate that DROID-SLAM achieves high robustness and outperforms not only other DL-based approaches, but also classical approaches on working sequences. This proves very good generalizability. Similarly, it manages to track 30/32 sequences of the RGB-D ETH-3D SLAM benchmark (Fig. 6), outperforming the second best result by 35%. Additionally, taking into account challenging nature of TUM-RGBD and ETH-3D SLAM, this shows that the system works well in the case of heavy rotation, motion blur, rolling shutter and in dark environments.

The ablation study in Fig 7 presents that the model benefits both from stereo data and global bundle adjustment.

4 Discussion / Conclusion

DROID-SLAM is currently the state-of-the-art Visual SLAM with high generalization capabilities across datasets and sensor modalities. It outperforms both classical and learned approaches by a large margin. In particular, it consequently achieves lower errors than DeepTAM described in introduction, even on small-scale tracks.

The biggest drawback of the system are large resource and memory requirements. To tackle it, the model is trained on low-resolution video which may result in low-quality reconstruction (additional artifacts visible in Fig. 2). In spite of the lowered computational effort, it is still not able to run in real time on some sequences (e.g. TartanAir). To this end, sparser frame associations in the frame graph could be tested in order to reduce computations.

Moreover, as presented in Fig. 4, its performance drops significantly for the cases in which loop closure is not performed. Interestingly, as depicted in Fig 7, using stereo data without global bundle adjustment reduces ATE to higher extent than applying global bundle adjustment on monocular data. I believe this observation could serve as a starting point for future work on drift reduction, which could be inspired by the virtual stereo approach presented in [10].

References

[1] Christopher B. Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Krishna Chandraker. Universal correspondence network. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2406–2414, 2016.

[2] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 224–236. Computer Vision Foundation / IEEE Computer Society, 2018.

[3] Raul Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robotics*, 31(5):1147–1163, 2015.

[4] Raul Mur-Artal and Juan D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robotics*, 33(5):1255–1262, 2017.

[5] Jürgen Sturm, Wolfram Burgard, and Daniel Cremers. Evaluating egomotion and structure-from-motion approaches using the tum rgb-d benchmark. 2012.

[6] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[7] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[8] Zachary Teed and Jia Deng. RAFT: recurrent all-pairs field transforms for optical flow. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, volume 12347 of *Lecture Notes in Computer Science*, pages 402–419. Springer, 2020.

[9] Zachary Teed and Jia Deng. DROID-SLAM: deep visual SLAM for monocular, stereo, and RGB-D cameras. *CoRR*, abs/2108.10869, 2021.

[10] Nan Yang, Rui Wang, Jörg Stückler, and Daniel Cremers. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, volume 11212 of *Lecture Notes in Computer Science*, pages 835–852. Springer, 2018.

[11] Huizhong Zhou, Benjamin Ummenhofer, and Thomas Brox. Deeptam: Deep tracking and mapping. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XVI*, volume 11220 of *Lecture Notes in Computer Science*, pages 851–868. Springer, 2018.