LEARNING BY INERTIA: SELF-SUPERVISED MONOCULAR VISUAL ODOMETRY FOR ROAD VEHICLES

Chengze Wang, Yuan Yuan, Qi Wang*

School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China

ABSTRACT

In this paper, we present iDVO (inertia-embedded deep visual odometry), a self-supervised learning based monocular visual odometry (VO) for road vehicles. When modelling the geometric consistency within adjacent frames, most deep VO methods ignore the temporal continuity of the camera pose, which results in a very severe jagged fluctuation in the velocity curves. With the observation that road vehicles tend to perform smooth dynamic characteristics in most of the time, we design the inertia loss function to describe the abnormal motion variation, which assists the model to learn the consecutiveness from long-term camera ego-motion. Based on the recurrent convolutional neural network (RCNN) architecture, our method implicitly models the dynamics of road vehicles and the temporal consecutiveness by the extended Long Short-Term Memory (LSTM) block. Furthermore, we develop the dynamic hard-edge mask to handle the non-consistency in fast camera motion by blocking the boundary part and which generates more efficiency in the whole non-consistency mask. The proposed method is evaluated on the KITTI dataset, and the results demonstrate state-of-the-art performance with respect to other monocular deep VO and SLAM approaches.

***Index Terms*—** Inertia, Self-supervised Learning, Visual Odometry, RCNN

1. INTRODUCTION

Simultaneous localization and mapping (SLAM) is one of the critical capabilities for robots and self-driving vehicles to navigate in no GPS or RTK (Real-Time Kinematic) available environment. VO is the front-end module of a typical visual SLAM, which uses visual information to preliminarily estimate each pixel's depth and camera motion in the adjacent frames.

In the past decades, multiple selections of sensors have been explored in the VO area, such as monocular camera, stereo camera, RGB-Depth, LiDAR, and visual-IMU (inertia measure unit). The monocular VO is under most investigations for its ubiquity and low cost.

For the monocular VO, model-based (or geometric) VO have been widely researched [1, 2, 3]. However, some disadvantages of these methods are insurmountable, e.g., it is hard to handle non-consistency (non-rigid) structure and not robust in the high dynamic situation.

To overcome the limitations as mentioned above, recent deep learning based VO has been implemented and achieved a considerable performance against traditional methods. One key shortcoming of deep VO is that the collection of the ground truth is expensive and laborious [4], so the self-supervised methods which mainly based on the view synthesis [5] emerge.

However, most existing deep VO are imposing constraints on the feature/pixel-level error between adjacent frames [6, 7, 8], or by the view synthesis [9, 10]. These VO estimate

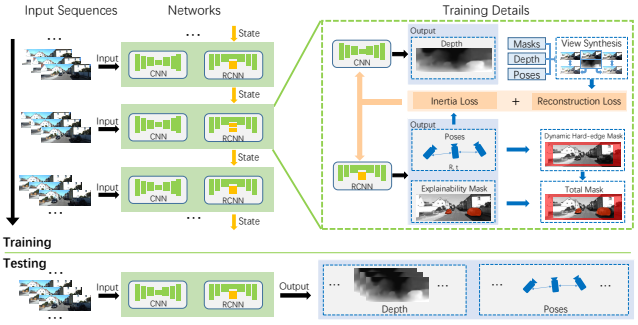

Fig. 1. The overview of iDVO. The CNN-RCNN dual networks structure takes the sequences as input, and outputs the per-pixel depth with 6DoF poses sequentially. The total mask used in view synthesis is combined by the computed dynamic hard-edge mask and the estimated explain-ability mask. Both networks can be tested individually.

*Corresponding author. This work was supported by the National Natural Science Foundation of China under Grant U1864204 and 61773316, State Key Program of National Natural Science Foundation of China under Grant 61632018, Natural Science Foundation of Shaanxi Province under Grant 2018KJXX-024, Projects of Special Zone for National Defense Science and Technology Innovation, Fundamental Research Funds for the Central Universities under Grant 3102017AX010, and Open Research Fund of Key Laboratory of Spectral Imaging Technology, Chinese Academy of Sciences.

ego-motion only by the spatial information existing in several frames, which means temporal information within the frames is not fully utilized. As a result, the output of deep VO is inaccurate and discontinuous. One of the most obvious results is that the estimated speed curves (calculate by poses) present a severe saw-tooth undulation.

Actually, with the effect of inertia, the motion variation of road vehicles is usually smooth, which means acceleration value is limited in a certain range. Under the normal driving situation, the vehicle's acceleration value is usually under 0.3g (gravitational acceleration) [11], and the deceleration caused by nonurgent braking is mostly less than 0.2g [12, 13]. In the following part of this paper, we will use the word "inertia" to describe these attributes of the motion status of vehicles.

Focusing on the dynamic characteristic caused by inertia, we proposed the iDVO, a self-learning based inertia-embedded deep visual odometry, as illustrated in Fig. 1.

The main contributions of this work are the following:

1) We propose the inertia loss to introduce the abnormal variation of motion status. This new constraint enables the model to learn from temporal and spatial consecutiveness within the vehicle motion simultaneously.

2) We extend the full convolutional architecture to RCNN while maintaining the system self-supervised. This RCNN is capable to automatically learn sequences' internal coupling.

3) Dynamic hard-edge mask is proposed to handle the massive viewpoint displacement in high-speed motion.

To the best of our knowledge, this is the first approach that imposes the dynamic characteristic for the deep VO. It is possible to train with any forward-looking driving video. The experimental results on KITTI [14] Odometry datasets have verified the effectiveness of iDVO.

2. OUR METHOD

Our method uses deep neural networks to learn the sequential ego-motion and the depth of frames from unlabelled dynamic driving video. On the network architecture, we introduce RNN block to our models to enhance the ability in dealing with temporal information. To treat with the non-consistency or dynamic structure in the frames, dynamic hard-edge mask is proposed to block unnecessary part of the image in projection. Lastly, the inertial attribution function is introduced, and the relevant spatial constraints are also discussed.

2.1. Network Architecture

The encoder-decoder architecture based on DispNet [15] has been proven to work efficiently in deep VO. This full-convolutional architecture automatically learns the features for ego-motion and depth estimation. However, as we mentioned in Sec. 1, CNN focuses on the limited input frames, and ignores the temporal continuity in the whole shot. Therefore, in our approach, we adopt RCNN architecture to implement an end-to-end deep VO equipped with the ability to keep the continuity in ego-motion.

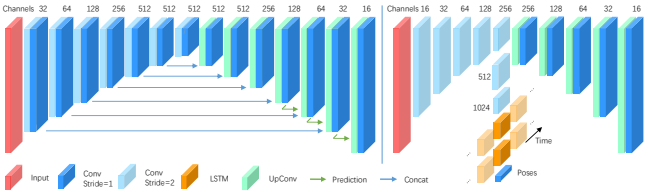

Fig. 2. The CNN-RCNN network structure. The correspondence between color and layer/operation is shown in the legend at the bottom. *The left one*: the DispNet [15] architecture is adopted for the depth estimator CNN. *The right one*: The pose-prediction RCNN. The decoder part is for multi-scale explain-ability mask prediction.

Inspired by SfMLearner [9], our networks have two independent networks in form of encoder-decoder as shown in Fig. 2, one for the depth inferring, and the other one for the pose estimating and explain-ability mask generation. The depth inferring network is fully convolutional, estimating each pixel's depth of the input single image. For the first 4 convolutional layers, the kernel size is set to 7, 7, 5, 5, while all the other convolutional layers use the size of 3.

To learn the connections in the sequence of video, we adopt the RCNN structure to the pose estimator network. Due to the regular RNN is hard to train on long time sequence, we choose to align the LSTM units with the original full convolutional network. Specifically, The ego-motion estimating and explain-ability mask generating network is extended with 2 layers of LSTM cells at the end of ego-motion estimating procedure. The LSTM cells input with the feature map generated by the fifth convolutional layer, and output pairs of relative poses in 6-DoF, and each of the LSTM layers has 1000 hidden states. The multi-scale explain-ability mask is generated by the last 4 convolutional layers in the decoder part. During the training, to maintain the order of sequence frames in every single shot, we abandon the file-level shuffle and replace it with a shot-level random selection.

2.2. Dynamic Hard-edge Mask

Because our networks are trained by the video captured in dynamic vehicles (we remove the static frames by optical flow from the sequences), the view point will differ time by time, which means some obstacles at the edges in one frame might not be captured in the next frame (when forward moving). As a result, in the reconstruction procedure, some pixels at the edge part of the resource frame will not be mapped in the boundary of the target frame.

To generate the non-consistency mask efficiently and pre-

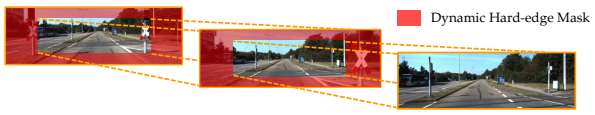

Fig. 3. The sketch of dynamic hard-edge mask (the red part). For better display, the frame-to-frame time between these 3 frames is ∼0.5s, so the mask is much larger than actual mask we used in experiments.

cisely, we propose the dynamic hard-edge mask (DHEM). The DHEM M_t^{h} is a hard mask in the edge of resource frame (as shown in Fig. 3), which together with the soft estimated explain-ability mask M_t^{e} will form a complete mask M_t at moment t. The hard mask blocks the edge pixels by ignoring the pixels in the mask during reconstruction. The edge mask's width of the mask is a positive correlation to the velocity at the frame-taken moment, which is calculated by the estimated pose from the ego-motion RCNN. The width of the left and right borders will be moderately adjusted according to the instant steering amplitude.

This efficient DHEM completes non-consistency mask with the explain-ability mask for the accurate view synthesis, which helps the training converge, also enhances the performance of depth and ego-motion estimation.

2.3. Loss Functions

2.3.1. Inertia Loss Construction

The inertia loss is constructed by the estimated poses in a period of time, more precisely it is calculated by the acceleration and jerk (the deviation of acceleration). We assume the vehicle accelerate and decelerate smoothly in the video capture procedure, and jerk of them also varies gently in a reasonable range. The input of our VO is sequential video captured on riding road vehicle. At the moment t , the captured frame is F_t, and the depth network estimates its corresponding depth map D_t. Then the ego-motion estimation network estimates the relative 6-DoF poses T_t and T_{t+1} between adjacent frames F_{t-1}, F_t and F_{t+1} (if using three-frame snippet). Every relative pose T is constructed by position P and orientation φ.

The inertia loss is calculated by the estimated poses $T_{(t_0,\ldots,t,t+1,t+2,\ldots)}$ in one single shot. Here we represent the displacement as velocity v_t for approximate fixed frame-to-frame time (∼0.1s in KITTI). In loss calculating, φ is in the form of Euler angles to avoid the possible optimization problem in learning. The straight-line velocity and angular velocity are calculated individually by:

$$v_t = \|P_t - P_{t-1}\|, \varphi_t = \|\varphi_t - \varphi_{t-1}\|. \tag{1}$$

Then, the sum of both velocity difference form the "acceleration" value a_t as follows:

$$a_t = (v_t - v_{t-1}) + \chi(\varphi_t - \varphi_{t-1}), \tag{2}$$

where χ is a scale factor to balance the angular acceleration. The "acceleration" value here is a scalar to describe the variation of the vehicle motion, rather than the exact acceleration vector. And the jerk j_t is simply calculated by: $j_t = a_t - a_{t-1}$. Because both a_t and j_t have a reasonable range, we ignore the value in loss if their values less than their typical value a_{typ} and j_{typ}, respectively:

$$\begin{aligned} L_{acce}(T_t) &= \max(0, \tfrac{|a_t| - a_{typ}}{a_t}) \\ L_{jerk}(T_t) &= \max(0, \tfrac{|j_t| - j_{typ}}{j_t}). \end{aligned} \tag{3}$$

Finally, the entire inertia loss becomes:

$$L_i = \sum_t |L_{acce}(T_t) + L_{jerk}(T_t)|. \tag{4}$$

2.3.2. Spatial Losses Construction

The main supervision in our method is still the view synthesis: given a frame shot in a scene, synthesize another frame in a different pose. In our learning procedure, given a pair of images F_t and F_{t-1}, we use the networks to estimate the ego-motion T_t and the depth D_t and D_{t-1}. The depth D_t can be projected as a point cloud C_t in the view position of F_t, so does the D_{t-1}. By using the spatial transformer [16], we can synthesize the reconstructed frames F_t' and F_{t-1}'. Comparing the reconstructed frames with the target frames, we can propose the differentiable image reconstruction loss L_{rec}:

$$L_{rec} = \sum_{u,v} \left\| (F_t^{uv} - F_t^{'uv}) M_t^{uv} \right\| \tag{5}$$

where $p_t(u, v)$ is one pixel in F_t ,and the M_t is the non-consistency mask as we mentioned in Sec. 2.2. To prevent the non-consistency mask M_t from minimizing to 0, the mask loss L_{mask} is implemented in our work, which is the binary cross-entropy loss between the mask and a same-size frame with constant 1 value at every pixel.

In order to comprehensively compare the reconstructed frame F_t' with the target frame F_t, structured similarity (SSIM) loss L_{SSIM} [17] and 3D point cloud alignment loss L_{3D} [18] are also introduced as additional loss.

Final Loss All loss functions in this section are computed at 4 different scales s, which are $\frac{1}{8}$, $\frac{1}{4}$, $\frac{1}{2}$, and original input resolution. The total loss is:

$$L_{total} = \omega_1 L_i + \sum_s (\omega_2 L_{rec}^s + \omega_3 L_{SSIM}^s + \omega_4 L_{3D}^s + \omega_5 L_{mask}^s). \tag{6}$$

The total loss leads the model to learn the VO task in 3 perspectives. The inertia loss demonstrates the temporal constraint, while the SSIM loss and 3D loss demonstrate the spatial constraint. The view synthesis loss and the mask loss incarnate temporal and spatial constraint simultaneously.

3. EXPERIMENTS

In this section, we evaluate the proposed iDVO system with several state-of-the-art VO and SLAM systems in two perspective, depth estimation and pose estimation.

3.1. Datasets and Implementation Details

The KITTI dataset [14] is the most common benchmark for VO algorithm in the transportation scene. During training the frames are cropped and resized to 416 $\times$ 128, and multiple images are stacked to one snippet as ego-motion RCNN's input. Data augmentation including rescaling, cropping and luminance correction is applied to enlarge the KITTI dataset. In training, we remove static frames automatically by limiting the optical flow between 2 frames.

We use Pytorch framework to implement our networks, and the optimizer is set to Adam with β_1 = 0.9, β_2 = 0.999, learning rate α_2 = 0.0002. Typical values a_{typ} and j_{typ} equal to 2.0, 0.5, and batch-size is set to 16. And hyper-parameters χ, ω_1, ω_2, ω_3, ω_4 and ω_5 are determinded empirically, which are respectively set to 100, 1, 1, 0.2, 0.1, 0.15. Due to using typical values a_{typ} and j_{typ}, the depth estimator starts training with pre-trained model (10k iteration, SfMLearner) to make it easier to converge.

3.2. Pose Estimation Evaluation

In order to facilitate evaluation results, we choose KITTI's monocular sequences 09 and 10 for pose estimation evaluation. ORB-SLAM [1] is a well-known SLAM system with loop closure and re-localization function, which indicates that the full version ORB-SLAM use the whole video sequences as supervision. ORB-SLAM (short) is supervised by 5 frame snippets, as same as our method's input. The baseline is the SfMLearner [9] trained with Cityscapes [19] and KITTI dual datasets. Futher more, we also compare our iDVO to the enhanced-SfMLearner proposed by Mahjourian et al. [18] , which represents the state-of-the-art performance. Similar to [9], we need to solve the scale factor by post-processing. Absolute Trajectory Error (ATE) for all methods is computed on 5-snippets and then averages over the entire sequence. The results of motion estimation are presented in Table 1. The results show that our method outperforms both the state-of-the-art deep VO approach and full SLAM system on monocular camera. More importantly, the smaller variance proves that our method has better stability on long-term ego-motion, which is critical in autonomous driving application.

To analyze the effectiveness of different components, we test three variants of our iDVO. 1) On non-consistency mask, we only use explain-ability mask. 2) The inertia loss is dropped. 3) The LSTM cells to predict poses in RCNN are replaced by the CNN blocks with kernel size of 1. The results of ablation study in Table 2. show all three modification are essential to enhance the performance.

Table 1. Absolute Trajectory Error (ATE) results on KITTI VO Sequence 09 and 10 (lower is better).

Method	Seq. 09	Seq. 10
ORB-SLAM (full)	0.014±0.008	0.012±0.011
ORB-SLAM (short)	0.064±0.141	0.061±0.130
Zhou et al. [9]	0.021±0.017	0.020±0.015
Mahjourian et al. [18]	0.013±0.010	0.012±0.011
Our iDVO	**0.012±0.011**	**0.012±0.010**

Table 2. Absolute Trajectory Error (ATE) results of ablation experiments.

Method	Seq. 09	Seq. 10
Full iDVO	**0.012±0.011**	**0.012±0.010**
w/o DHEM	0.013±0.012	0.014±0.010
w/o Inertia loss	0.016±0.014	0.015±0.015
w/o RCNN	0.014±0.016	0.013±0.014

3.3. Depth Estimation Evaluation

The depth estimation evaluation which using the Eigen [6] test set. Table 3 quantitatively compares the depth estimation between iDVO and several state-of-the-art deep VO systems [20, 9, 18, 21]. Table 3 shows that our method achieves the best performance except for Sq Rel section, and performs a big lead in RMSE section. On the whole, our method has a great improvement than state-of-the-art algorithms in depth estimation, and also proves that constraints we have added to the ego-motion estimation can be effectively reflected in the depth estimation.

Table 3. Single-view depth evaluation results on the KITTI Eigen et al. [6] set.

Method	Abs Rel	Sq Rel	RMSE	RMSE log
Eigen et al. [6]	0.203	1.548	6.307	0.282
Liu et al. [20]	0.202	1.614	6.523	0.275
Zhou et al. [9]	0.208	1.768	6.856	0.283
Mahjourian et al. [18]	0.163	**1.240**	6.220	0.250
Li et al. [21]	0.183	1.73	6.570	0.268
Our iDVO	**0.162**	1.277	**5.114**	**0.248**

4. CONCLUSION

Most existing unsupervised deep VO optimize errors by using the image-based constraint such as the inaccuracy and coarseness of the reconstructed frame. These methods lack considerations of the temporal continuity of the camera pose, resulting in a severe saw-tooth undulation in the velocity curves. Based on the prior of road vehicles' dynamic characteristics, we present a loss function to describe the abnormal estimated motion, and the results proved that it is effective to build constraints on long-term sequential ego-motion to supervise the training. Also, the RCNN and the DHEM improve the performance of the algorithm from temporal and spatial perspective respectively. Experimental results show that our iDVO achieves state-of-the-art performance.

5. REFERENCES

[1] Raul Mur-Artal and Juan D Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[2] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison, "Dtam: Dense tracking and mapping in real-time," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2320–2327.

[3] Jakob Engel, Vladlen Koltun, and Daniel Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2018.

[4] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[5] Richard Szeliski, "Prediction error as a quality metric for motion and stereo," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*. IEEE, 1999, vol. 2, pp. 781–788.

[6] David Eigen, Christian Puhrsch, and Rob Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in neural information processing systems*, 2014, pp. 2366–2374.

[7] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox, "Demon: Depth and motion network for learning monocular stereo," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, vol. 5.

[8] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017, vol. 2, p. 7.

[9] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe, "Unsupervised learning of depth and ego-motion from video," in *CVPR*, 2017, vol. 2, p. 7.

[10] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki, "Sfm-net: Learning of structure and motion from video," *arXiv preprint arXiv:1704.07804*, 2017.

[11] Markus Kuderer, Shilpa Gulati, and Wolfram Burgard, "Learning driving styles for autonomous vehicles from demonstration," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2641–2646.

[12] Laura Eboli, Gabriella Mazzulla, and Giuseppe Pungillo, "Combining speed and acceleration to define car users safe or unsafe driving behaviour," *Transportation research part C: emerging technologies*, vol. 68, pp. 113–125, 2016.

[13] Arpan Mehar, Satish Chandra, and Senathipathi Velmurugan, "Speed and acceleration characteristics of different types of vehicles on multi-lane highways," *European Transport*, vol. 55, pp. 1825–3997, 2013.

[14] Andreas Geiger, Philip Lenz, and Raquel Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[15] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4040–4048.

[16] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al., "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.

[17] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[18] Reza Mahjourian, Martin Wicke, and Anelia Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5667–5675.

[19] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[20] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.

[21] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu, "Undeepvo: Monocular visual odometry through unsupervised deep learning," *arXiv preprint arXiv:1709.06841*, 2017.