Sequential Adversarial Learning for Self-Supervised Deep Visual Odometry

Shunkai Li Fei Xue* Xin Wang* Zike Yan Hongbin Zha
Key Laboratory of Machine Perception (MOE), School of EECS, Peking University
PKU-SenseTime Machine Vision Joint Lab
{lishunkai, feixue, xinwang_cis, zike.yan}@pku.edu.cn zha@cis.pku.edu.cn

Abstract

We propose a self-supervised learning framework for visual odometry (VO) that incorporates correlation of consecutive frames and takes advantage of adversarial learning. Previous methods tackle self-supervised VO as a local structure from motion (SfM) problem that recovers depth from single image and relative poses from image pairs by minimizing photometric loss between warped and captured images. As single-view depth estimation is an ill-posed problem, and photometric loss is incapable of discriminating distortion artifacts of warped images, the estimated depth is vague and pose is inaccurate. In contrast to previous methods, our framework learns a compact representation of frame-to-frame correlation, which is updated by incorporating sequential information. The updated representation is used for depth estimation. Besides, we tackle VO as a self-supervised image generation task and take advantage of Generative Adversarial Networks (GAN). The generator learns to estimate depth and pose to generate a warped target image. The discriminator evaluates the quality of generated image with high-level structural perception that overcomes the problem of pixel-wise loss in previous methods. Experiments on KITTI and Cityscapes datasets show that our method obtains more accurate depth with details preserved and predicted pose outperforms state-of-the-art self-supervised methods significantly.

1. Introduction

The ability for an agent to understand 3D environment and infer ego-motion is crucial for many real-world applications, such as autonomous driving [7], robotics [14], and virtual/augmented reality [30]. As the problem of simultaneous localization and mapping (SLAM) and visual odometry (VO) has a clear meaning in 3D geometry, VO/SLAM has been studied as a multi-view geometric problem for decades. These classic methods [11, 12, 15, 25, 29] perform well in regular scenes, but fail in challenging conditions due to their inherent reliance on low-level feature correspondences. Since deep learning captures structural perception by extracting high-level features, a number of learning-based VO methods have been applied to break through the limitations of classic approaches [19, 33, 34, 36, 37, 41].

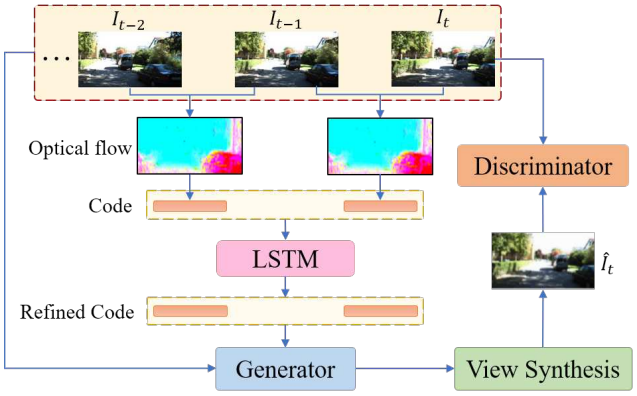

Figure 1. Overview of our method. The network extracts optical flow into a compact code, which is incorporated by LSTM to aggregate historical information and refine previous estimations. Depth and pose estimation is regarded as an image conditioned generative task, and the refined code is provided as input signal. The geometric inference is used to reconstruct a warped image by view synthesis and evaluated by a discriminator.

However, supervised learning requires substantial labeled data, which is either tedious or impractical to obtain. Recent work has been trying to address this problem by coupling depth and pose estimation in a self-supervised manner [39, 42]. As image sequence is the only input, all the estimations should be mapped to image space for self-supervision. The mapping is typically made by view synthesis and photometric loss is defined to minimize the difference between synthesized image and the real one.

In self-supervised VO, estimation of depth and pose are simultaneously learned in a coupled way, accurate depth contributes to precise pose estimation and vice versa. Previous works on self-supervised VO estimate depth from single

*equal contribution

view. As an ill-posed problem, the output depth is vague, hence the predicted pose is also inaccurate. However, uncertainties in depth estimation can be eliminated by exploiting correlations between consecutive frames. Nonetheless, due to the data redundancy of image sequence, it is inefficient to integrate the information of multiple frames by stacking them along the RGB channel. In this paper, we propose to learn a compact representation (referred to as 'code') of the correlation between frames, and sequential information is accumulated by integrating codes via Long Short-Term Memory (LSTM). The code provides correlations of consecutive frames that help generate clear depth maps and reduce accumulated error over a long sequence.

On the other hand, inaccurate depth and pose leads to distortion artifacts in synthesized images (Fig. 3), which are difficult to be eliminated by photometric loss due to the pixel-level correspondence. A new evaluation criterion with structural perception is needed for accurate depth estimation. In this paper, we tackle VO as a self-supervised image generation task and take advantage of Generative Adversarial Networks (GAN) [18]. The generator learns to estimate depth and pose to synthesize a warped image, while the discriminator evaluates the quality of synthesized image with structural perception and higher-level understandings. This two-player game impels the generator to estimate more accurate depth and pose, while the discriminator is able to distinguish distortion artifacts with structural perception.

The overview of our method is shown in Fig. 1. Different from single-view estimation, our method generates clear depth with additional information which cannot be retrieved from a single image. The information is obtained by encoding optical flow into a compact code, and codes of multiple frames are incorporated and refined by LSTM. The overall framework is treated as a generative model with adversarial learning. During training, the spatial-temporal consistency is enforced as self-supervision. The main contributions of our paper can be summarized as follows:

- We propose to exploit spatial-temporal correlations over long sequence to significantly reduce estimation errors and scale drift for self-supervised VO.
- We treat self-supervised VO as a generative model and take the advantage of adversarial learning for self-supervised pose and depth estimation.

Our method outperforms state-of-the-art self-supervised approaches significantly and gives comparable results with supervised manners. Extensive experiments manifest the advantages of our model. Besides, the idea of self-supervised adversarial learning with spatial-temporal consistency may also bring insight into VO/SLAM and video-based computer vision researches.

2. Related works

Humans are capable of perceiving 3D environment and inferring ego-motion in a short time, but it is hard for an agent to be equipped with similar capabilities. VO/SLAM has been considered as a multi-view geometric problem for decades. It is traditionally solved by minimizing photometric [12] or geometric [29] reprojection errors and works well in regular environments, but fails in challenging conditions like dynamic objects and abrupt motions. In light of these limitations, VO has been studied with learning techniques in recent years and many approaches with promising performance have been proposed.

Supervised methods formulate VO as a supervised learning problem and many methods with good results have been proposed. DeMoN [33] jointly estimates pose and depth in an end-to-end manner. Inspired by the practice of parallel tracking and mapping in classic VO/SLAM, DeepTAM [41] utilizes two networks for pose and depth estimation. DeepVO [34] treats VO as a sequence-to-sequence learning problem by estimating poses recurrently. The limitation of supervised learning is that it requires a large amount of labeled data. The acquisition of ground truth often requires expensive equipment or highly manual labeling, and some gathered data are inaccurate. Depth obtained by LIDAR is sparse, and the output depth of Kinect contains a lot of noise. Furthermore, some ground truth is unable to obtain (*e.g.* optical flow). Previous works have tried to address these problems with synthetic datasets [9], but there is always a gap between synthetic and real-world data.

Self-supervised methods In order to alleviate the reliance on ground truth, recently many self-supervised methods have been proposed for VO. The key to self-supervised learning is to find the internal correlations and constraints in the training data. SfMLearner [42] leverages the geometric correlation of depth and pose to learn both of them in a coupled way, with a learned mask to mask out regions that don't meet static scene assumption. As the first self-supervised approach for VO, SfMLearner couples depth and pose estimations with image warping, which becomes the problem of minimizing photometric loss. Inherited from this idea, many self-supervised VO have been proposed, including modifications on loss functions [22, 26], network architectures [3, 4, 22, 28, 40], predicted contents [39], and combination with classic VO/SLAM [5, 38]. For example, GeoNet [39] extends the framework to jointly estimate optical flow with forward-backward consistency to infer unstable regions and achieves state-of-the-art performance among self-supervised VO methods.

Despite its feasibility, self-supervised VO still underperforms supervised ones. Apart from the effectiveness of direct supervision, a key reason is that they focus mainly on geometric properties [42] but pay little attention to the sequential nature of the problem. In these methods, only a

few frames (no more than 5) are processed in the network, while previous estimations are discarded and the current estimation is made from scratch. Instead, the performance can be enhanced by taking geometric relations of sequential observations into account.

Our approach differs from previous art in formulating self-supervised VO as a sequential learning problem. The frame-to-frame correlation is represented as a compact code, and sequential information are integrated via LSTM. In contrast to prevalent single-view depth estimation, our framework estimates depth with the code conditioned on a single image and treat VO as a generative task. By means of adversarial learning, our method provides sharper depth and more accurate pose estimations.

3. Method

In this section, we will introduce our method in detail. The entire framework consists of four components (Fig. 2). The encoder extracts high-level features from optical flow into a compact code in Sec. 3.1, and the codes are aggregated and further refined by LSTM in Sec. 3.2. The generator estimates depth and pose conditioned on refined codes and images in Sec. 3.3-3.4. The discriminator in Sec. 3.5 judges the authenticity of a synthesized view. Finally, loss functions used in training are defined in Sec. 3.6.

3.1. Encoder

Visual odometry estimates camera motion between consecutive image pairs. This estimation is computed by feature correspondence or photometric consistency in classic VO/SLAM. Different from previous self-supervised methods that estimate directly from raw images, we provide the network with a representation of frame-to-frame correspondence for depth and pose estimation.

As a way of frame-to-frame correspondence, parallax and motion of each pixel can be obtained by computing optical flow between consecutive images. In our framework, we compute optical flow [13] and extract it into a compact representation (referred to as 'code') c_t with a size of 128

$$c_t = \mathcal{C}(\mathcal{F}(I_{t-1}, I_t)). \tag{1}$$

The extracted c_t will be incorporated with historical information and used as side input for depth and pose estimation.

3.2. Sequential information aggregation

Estimating depth and pose from only a few frames is prone to error accumulation and scale drift. The problem can be mitigated by exploiting correlations over long sequence. This formulation is appealing for self-supervised sequential estimations since it utilizes incoming observations and spatial-temporal consistency as self-supervision.

In our framework, we use LSTM [20] to model VO as a self-supervised sequential learning problem. As an extension of recurrent neural networks (RNN), LSTM introduces a *cell* to remember and forget information adaptively. LSTM fuses the code c_t of current frame I_t into accumulated information. Intuitively, the long-term information is remembered as a prior, and short-term memory is used to infer the current state. The feature flow passing through recurrent units carries rich information of previous states, enabling refined outputs to improve the current estimation

$$c_t^{'}, h_t = \mathcal{U}(c_t, h_{t-1}), \tag{2}$$

where $c_t^{'}$ denotes the refined code that incorporates historical information, and h_{t-1}, h_t are hidden states at time $t-1$, t, respectively.

3.3. Depth estimation

In the existing literature, depth is estimated from a single image I

$$\hat{D} = \mathcal{D}(I). \tag{3}$$

As an ill-posed problem, the estimated depth is reasonable on the whole but vague in details. On the other hand, simply stacking multiple frames does not improve the result of depth estimation [42]. In order to obtain a clear depth, correlations of multiple views should be provided as additional information which cannot be retrieved from a single image.

Because of the high degree of order and regularity of 3D scenes, depth can be effectively represented by a compact feature with a single image [6]. As motion parallax of two frames reflects the distance of each part of the scene, we provide the refined code $c_t^{'}$ as side input for depth estimation

$$\hat{D}_t = \mathcal{D}(I_t, c_t^{'}). \tag{4}$$

As an image conditioned depth generation process, I_t is extracted into a feature map by convolutional layers, which is further concatenated with $c_t^{'}$ in the network. It is then followed by up-sampling layers with skip connections.

3.4. Pose and mask estimation

Most self-supervised VO methods regress pose directly from images but fail to exploit the depth of two views. In classic methods, pose regression from images and depth is solved by RGBD registration, such as using image feature detection for initial guess and robust 3D correspondence for pose refinement [23, 31]. In order to exploit both color and depth information, we stack images and depth maps into 2 RGBD images for pose estimation from $t-1$ to t

$$\hat{T}_{t-1}^{t} = \mathcal{P}((I_{t-1}, \hat{D}_{t-1}), (I_t, \hat{D}_t)). \tag{5}$$

After the acquisition of pose and depth, image warping is used for view synthesis. The homogeneous coordinate of

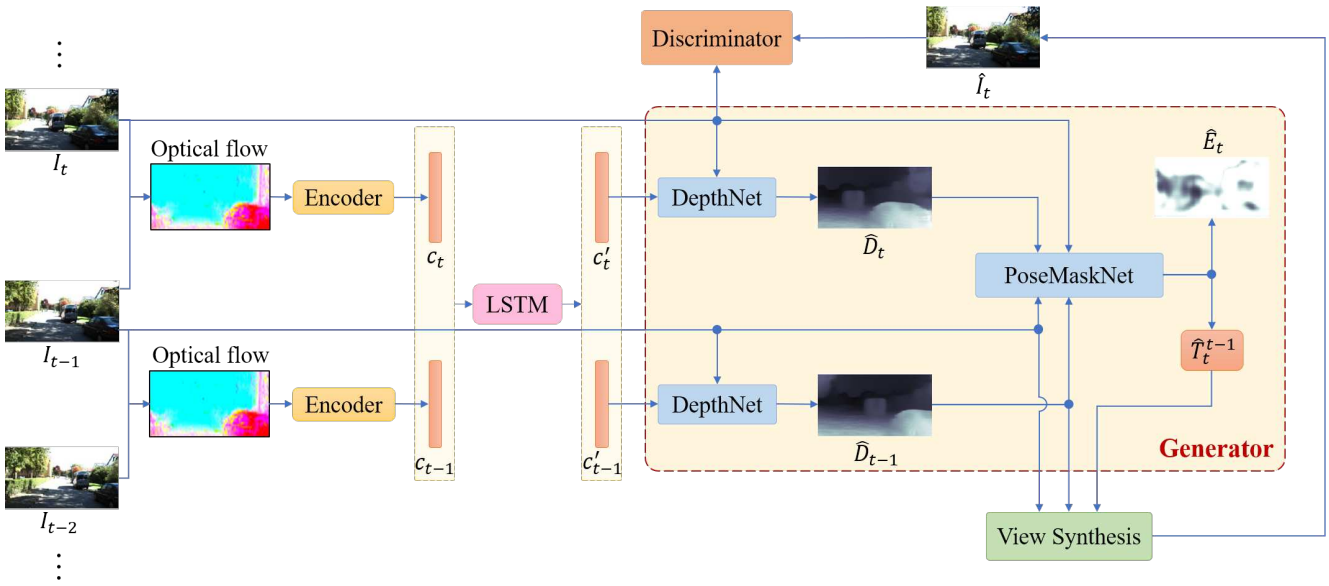

Figure 2. Illustration of our framework. The encoder compresses optical flow of two consecutive images into a compact code, which is aggregated and refined by LSTM. The DepthNet estimates depth conditioned on the refined code and input image. The estimated depth is concatenated with image for pose and mask prediction, while the authenticity of the warped image is judged by the discriminator. The discriminator is excluded during the test phase.

a pixel in the target view p_t and the source view p_{t-1} are correlated by [42]

$$p_{t-1} \sim K\hat{T}_t^{t-1}\hat{D}_t(p_t)K^{-1}p_t, \tag{6}$$

where K denotes camera intrinsics. We use differentiable bilinear sampling as [42]. In this way, the synthesized image $\hat{I}_t$ and I_t can be used for self-supervision.

Nonetheless, view synthesis builds on the assumption that the scene is static without illumination change and occlusions, which is often violated in practice. To overcome this problem, our framework learns to predict a per-pixel mask $\hat{M}_t$ as a belief in how successful a target pixel is rendered during view synthesis [42]. Consequently, the weighted photometric loss is

$$\mathcal{L}_{pho} = \sum_{<I_1,\dots,I_N>} \sum_{p} \hat{M}_t(p)\|\hat{I}_t(p) - I_t(p)\|_1. \tag{7}$$

3.5. Discriminator

Photometric loss is widely used in self-supervised VO and the warped results are shown in Fig. 3. Despite convolutional neural networks (CNN) extract high-level features that prevent low-level feature problem in classic VO/SLAM, the loss function is still based on pixel-level instead of evaluating on a larger receptive field with higher-level understandings. Due to the pixel-level correspondence and photometric consistency assumption, photometric loss is not robust to occlusion, texture-less regions, dynamic objects and illumination change. In these challenging conditions, there are multiple local minima with similar magnitudes. The network tends to trap into any of them during training with vague depth and wrong pose, leading to inaccurate reconstruction (Fig. 3). Some of previous research have realized this problem [39, 40] and try to eliminate this disturbing factor by explicitly modeling motion segmentation and optical flow, but achieve limited improvement.

Instead, the distortion artifacts are easily detectable by a discriminator. The compelling results achieved by GAN have been successfully demonstrated in many image generation tasks [1, 21, 43]. The adversarial learning impels the network to learn more flexible distributions to tackle underfitting issues and overcome gradient locality. In the self-supervised paradigm, VO can be regarded as a conditional image generation task

$$\hat{I}_t = G(c'_{t-1}, c'_t | I_{t-1}, I_t). \tag{8}$$

I_t is a sample from distribution p_{real}, and $\hat{I}_t$ is generated from c'_{t-1}, c'_t on the latent space p_{code}.

During training, the generator tries to fool the discriminator by generating better pose and depth. Meanwhile, given I_t as side information, the discriminator tries to distinguish the fake $\hat{I}_t$ by predicting a probability of authenticity $D(\hat{I}_t|I_t)$. The adversarial training overcomes the problem of Eq. (7) to produce accurate depth and pose without explicit modeling of motion segmentations and optical flow.

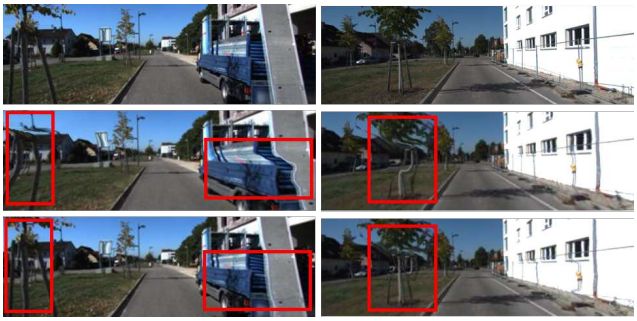

Figure 3. Example of warped images according to the estimated depth and pose. Top row: captured images, medium row: warped images of SfMLearner [42], bottom row: warped images of our method. It can be seen that inaccurate predictions will lead to distortion artifacts on the warped image. Compared to the existing literature, our method synthesizes more accurate warped images.

The value function of this min-max game can be formulated according to [21]

$$\begin{aligned}\mathcal{L}_{GAN} =& \min_G \max_D V(G,D) \\ =& \mathbb{E}_{I_t \sim p_{real}}[\log(D(I_t|I_t))] + \\ & \mathbb{E}_{c'_{t-1}, c'_t \sim p_{code}}[\log(1 - D(\hat{I}_t|I_t))].\end{aligned} \tag{9}$$

3.6. Loss functions

Appearance loss In order to overcome the pixel-level correspondence problem, we measure the reconstructed images from both weighted photometric loss and structural similarity metric (SSIM) [35]

$$\begin{aligned}\mathcal{L}_{ap} =& \mathcal{L}_{reg}(\hat{M}) + (1-\alpha)\mathcal{L}_{pho} \\ & + \frac{1}{N}\sum_{x,y} \alpha \frac{SSIM(\hat{I}(x,y), I(x,y))}{2},\end{aligned} \tag{10}$$

$\mathcal{L}_{reg}(\hat{M})$ is a regularization term to prevent the network converges to a trivial solution, which is detailed in [42]. N is the number of images in the training minibatch. The filter size of SSIM is set 10×10 and α is set 0.85.

Depth regularization Discontinuity of depth usually happens where strong image gradients are present. Similar to [4, 40], we introduce an edge-aware smoothness loss to enforce discontinuity and local smoothness in depth

$$\begin{aligned}\mathcal{L}_{smo} =& \frac{1}{N}\sum_{x,y} \|\nabla_x \hat{D}(x,y)\| e^{-\|\nabla_x I(x,y)\|} + \\ & \|\nabla_y \hat{D}(x,y)\| e^{-\|\nabla_y I(x,y)\|}.\end{aligned} \tag{11}$$

Trajectory consistency Although LSTM-based framework is suffice to provide more accurate poses by filtering out the noise between consecutive transformations, the estimated $\hat{T}_t^{t-1}$ are still relative poses. There are no relations and no geometric consistency among them. Actually, these relative poses can be transformed into a unified coordinate by accumulating them along the trajectory. According to rigid-body transformation, given a set of transformations such as $A \to B \to C \to D$, the relative poses T_A^B, T_B^C, T_C^D satisfies the following constraints [22]

$$\begin{aligned}T_A^B \cdot T_B^C \cdot T_C^D &= T_A^D, \\ T_A^B \cdot T_B^C &= T_A^C, \\ T_B^C \cdot T_C^D &= T_B^D,\end{aligned} \tag{12}$$

In order to enforce trajectory consistency, we compute the following loss on three scales for every eight frames

$$\mathcal{L}_{TC} = \frac{1}{N}\sum_{i=1}^{N} \sum_{t\in[2,4,8]} \|\hat{p_d}_i^{i+t} - \hat{p_r}_i^{i+t}\|_1, \tag{13}$$

where $\hat{p_d}_i^{i+t}$ is the 6-DoF pose directly estimated from (I_i, $c_i^{'}$) and (I_{i+t}, $c_{i+t}^{'}$), and $\hat{p_r}_i^{i+t}$ is the concatenated 6-DoF pose of successive relative transformations.

GAN loss in Eq. (9) acts as an auxiliary self-supervision for the synthesized image. The final loss function becomes

$$\mathcal{L}_{final} = \lambda_a \mathcal{L}_{ap} + \lambda_s \mathcal{L}_{smo} + \lambda_t \mathcal{L}_{TC} + \lambda_g \mathcal{L}_{GAN}. \tag{14}$$

4. Experiments

In this section, we will introduce the implementation details and show both qualitative and quantitative results compared with other methods. In the end, an ablation study is employed to test the effectiveness of each component in our framework.

4.1. Implementation details

As shown in Fig. 2, our framework includes 4 sub-networks. Both DepthNet and PoseMaskNet consist of encoding and decoding parts. The encoders are made up of 6 convolutional downsampling layers with stride 2, and decoders transform the extracted features into depth or masks with deconvolutional layers. Both depth and masks are predicted in 4 scales. In order to preserve both high-level and detailed information of the image, skip connections are used between encoders and decoders at corresponding resolutions [42]. Meanwhile, the encoding part of PoseMaskNet is also followed by 2 fully-connected layers to regress Euler angles and translations of 6-DoF pose, respectively. The Encoder and discriminator follow the same architecture as the encoding part of DepthNet. The extracted feature from Encoder then passes through an average pooling layer to output a 128-channel vector. Batch normalization and ReLUs are adopted in each layer except for the output layers.

Our model is implemented by PyTorch [32] on a single NVIDIA GTX 1080Ti GPU. All sub-networks are trained

Method	Supervision	Dataset	Cap	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Train set mean	-	K	80m	0.361	4.826	8.102	0.377	0.638	0.804	0.894
Eigen *et al*. [10] Coarse	Depth	K	80m	0.214	1.605	6.563	0.292	0.673	0.884	0.957
Eigen *et al*. [10] Fine	Depth	K	80m	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu *et al*. [27]	Depth	K	80m	0.201	1.584	6.471	0.273	0.680	0.898	0.967
SfMLearner [42]	-	K	80m	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Vid2Depth [28]	-	K	80m	0.163	1.240	6.220	0.250	0.762	0.916	0.968
GeoNet [39]	-	K	80m	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Zhan *et al*. [40]	Stereo	K	80m	**0.135**	1.132	5.585	0.229	0.820	0.933	0.971
Ours	-	K	80m	0.150	**1.127**	**5.564**	**0.229**	**0.823**	**0.936**	**0.974**
Garg *et al*. [16]	Stereo	K	50m	0.169	1.080	5.104	0.273	0.740	0.904	0.962
SfMLearner [42]	-	K	50m	0.201	1.391	5.181	0.264	0.696	0.900	0.966
Vid2Depth [28]	-	K	50m	0.155	0.927	4.549	0.231	0.781	0.931	0.975
GeoNet [39]	-	K	50m	0.147	0.936	4.348	0.218	0.810	0.941	0.977
Zhan *et al*. [40]	Stereo	K	50m	**0.128**	**0.815**	4.204	0.216	**0.835**	0.941	0.975
Ours	-	K	50m	0.146	0.927	**4.107**	**0.216**	0.819	**0.943**	**0.981**
SfMLearner [42]	-	CS+K	80m	0.198	1.836	6.565	0.275	0.718	0.901	0.960
Vid2Depth [28]	-	CS+K	80m	0.159	1.231	5.912	0.243	0.784	0.923	0.970
GeoNet [39]	-	CS+K	80m	0.153	1.328	5.737	0.232	0.802	0.934	0.972
Ours	-	CS+K	80m	**0.136**	**1.064**	**5.176**	**0.289**	**0.830**	**0.942**	**0.976**

Table 1. Monodular depth estimation results on KITTI dataset by the split of Eigen *et al*. [10]. K and CS refer to KITTI and Cityscapes datasets, respectively. As for supervision, 'Depth' means the ground truth depth is used during training, 'Stereo' means stereo image sequences with known baselines between two cameras are used during training, and '-' means no supervision is provided. The results are capped at 80m and 50m, respectively. As for error metrics Abs Rel, Seq Rel, RMSE and RMSE log, lower value is better; as for accuracy metrics $\delta < 1.25$, $\delta < 1.25^2$ and $\delta < 1.25^3$, higher value is better.

together in an self-supervised manner. During training, images are resized to 128×416 and data augmentation (random rotation, zoom, color jitter) is applied to prevent overfitting. As suggested in WGAN [2], the stochastic gradient descent is used for the discriminator, and Adam [24] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.99$ is used for all the other networks. The length of LSTM is set 15, and weighting factors λ_a, λ_s, λ_t, λ_g are set 0.75, 0.1, 0.14 and 0.01, respectively. The training batch size is set 4 with a weight decay of 3×10^{-4} for 100,000 iterations. The initial learning rate is set 10^{-4} and reduced by half for every 15,000 iterations. The network infers depth and pose at the speed of $18ms$ per frame during the test.

4.2. Depth estimation

We take the split of Eigen *et al*. [10] and use monocular images to train and test depth estimation. Ground truth depth is obtained by projecting sparse laser-scanned depth points into images, and depth predictions are interpolated to be the same size as ground truth for evaluation. In order to solve the scale ambiguity problem, the predicted depth is multiplied by a scaling factor to match the median with ground truth. Following the evaluation protocol in [17], both 50m and 80m thresholds of maximum depth are used for evaluation. As with previous methods, we also pre-train the network on Cityscapes dataset [8] and fine-tune on KITTI to test its adaptability across different environments.

We provide a comparison with related works which have depth supervision [10] or calibrated stereo images with known camera baseline for self-supervision. As shown in Table 1, our method outperforms all self-supervised methods and achieves comparable results with supervised ones. In particular, KITTI and Cityscapes datasets differ not only in scene contents but also in camera intrinsics. Results in the bottom rows of Table 1 show that our method generalizes well in different environments. Since enhanced edges and details only take up a small proportion of depth maps, the improvement on depth accuracy is therefore limited.

Fig. 4 shows the qualitative examples of depth estimated by different methods. It can be seen that some methods have difficulty in recovering the depth of cars and mistake the depth of several objects. As the code provides frame-to-frame correspondence, our method produces clearer depth compared with single-view depth estimation approaches. Additionally, benefited from adversarial learning, the estimated depth preserves boundaries and thin structures, which is more accurate in details.

4.3. Pose estimation

In addition, we apply our method to KITTI odometry dataset for pose estimation. The dataset contains 11 driving scenes with ground truth poses. In order to make fair comparison, we follow the same train/test split as [39, 42] by using sequences 00-08 for training and 09-10 for test.

The performance of pose estimation is evaluated using Absolute Trajectory Error (ATE) for both translation and rotation. Our method is compared with SfMLearner [42], GeoNet [39], Vid2Dpeth [28], Zhan *et al*. [40] and ORB-SLAM, a representative framework in classic SLAM. ORB-SLAM (short) is emplemented by tracking module with lo-

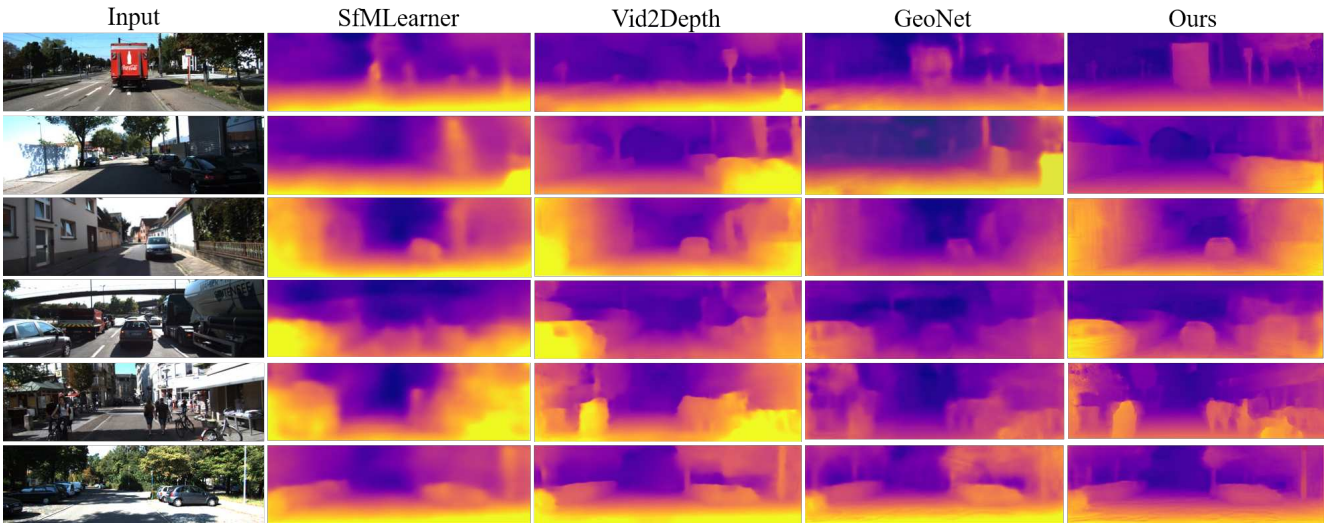

Figure 4. Selected depth estimations from the test on KITTI dataset. Our method shows better prediction on detailed structures, low texture regions and shaded areas than the other self-supervised VO approaches. The estimated depth is clear in both close and distant areas.

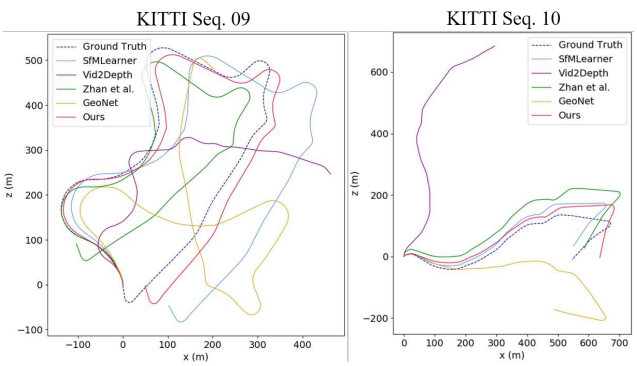

Figure 5. Trajectories of different methods on KITTI dataset. Our method shows a better odometry in both rotation and translation.

Method	Seq.09	Seq.10
ORB-SLAM [29] (short)	0.064±0.141	0.064±0.130
ORB-SLAM [29] (full)	0.014±0.008	0.012±0.011
SfMLearner [42]	0.021±0.017	0.020±0.015
SfMLearner [42] modified	0.016±0.009	0.013±0.009
Zhan *et al*. [40]	0.013±0.009	0.013±0.008
Vid2Depth [28]	0.013±0.010	0.012±0.011
GeoNet [39]	0.012±0.007	0.012±0.009
Ours	**0.0030±0.0014**	**0.0029±0.0012**

Table 2. Absolute Trajectory Error (ATE) on sequence 09 and 10 in KITTI odometry dataset. Our method outperforms all the other baselines by a large margin.

cal bundle adjustment, and ORB-SLAM (full) processes the entire sequence with loop closure and global bundle adjustment. Both versions of ORB-SLAM use a single scale map which is beneficial to an accurate trajectory with consistent scale. In order to solve the scale ambiguity problem in monocular VO, a scaling factor is used to align the trajectories with ground truth [40].

As shown in Table 2, our method significantly outperforms all the other baselines, and trajectories of sequences 09-10 are plotted in Fig. 5. In addition, although only a limited number of frames can be processed by LSTM, our method still performs better than ORB-SLAM (full) without any need of global optimization (such as loop closure, bundle adjustment and re-localization) [29]. This reveals that our method is able to produce accurate pose estimations by incorporating short-term correspondences and long-term dependences in odometry.

4.4. Ablation studies

In order to study the importance of each component, we perform ablation studies on various versions of our method. The baseline is our framework removing code, LSTM, trajectory consistency loss and discriminator. All the experiments are conducted on KITTI dataset and results are shown in Table 3, 4 and Fig. 6.

As shown in Fig. 6 (b), single view depth estimation is prone to be misled by the texture and color distributions in RGB images. The depth of poles is not recovered, and the depth of the sky is regarded the same as the white wall due to similar colors. In contrast, our method avoids these problems by taking additional information into account. The code encodes frame-to-frame correspondence which provides a significant improvement in depth estimation. The recovered depth is much sharper in contours and preserves tiny objects in both close and distant areas. In addition, adversarial learning gives the performance a further boost, and

Method	Dataset	Cap	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Baseline	K	50m	0.218	1.462	5.837	0.275	0.723	0.908	0.967
Baseline+code	K	50m	0.162	1.178	4.533	0.236	0.811	0.933	0.973
Baseline+code+GAN	K	50m	0.152	0.937	4.120	0.217	0.816	0.939	0.979
Baseline+code+LSTM	K	50m	0.148	0.939	4.271	0.217	0.816	0.941	0.977
Baseline+code+GAN+LSTM	K	50m	0.150	0.931	4.116	0.216	0.819	0.943	0.979
Baseline+code+GAN+LSTM+TC	K	50m	**0.146**	**0.927**	**4.107**	**0.216**	**0.819**	**0.943**	**0.981**

Table 3. Ablation study on depth estimation for various versions of our method. Baseline denotes our framework without code, LSTM, discriminator (*i.e.* GAN) and trajectory consistency (TC) loss.

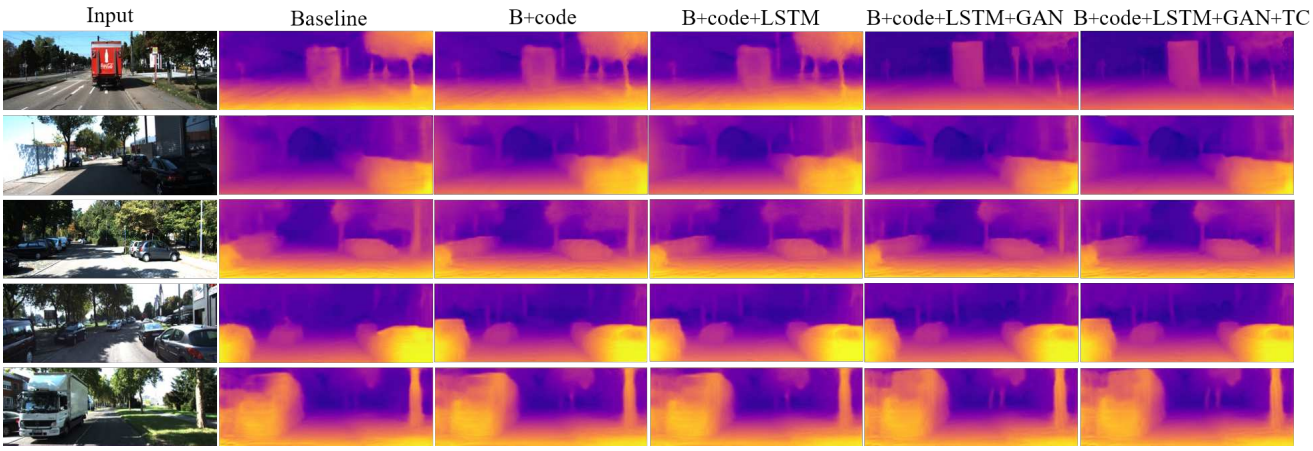

Figure 6. Ablation study on depth estimation of our method. B denotes our baseline method, which is our framework without code, LSTM, discriminator (*i.e.* GAN) and trajectory consistency (TC) loss.

Method	Seq.09	Seq.10
Baseline	0.0072±0.0025	0.0070±0.0023
B+code	0.0069±0.0021	0.0065±0.0020
B+code+GAN	0.0064±0.0019	0.0062±0.0019
B+code+LSTM	0.0045±0.0015	0.0043±0.0015
B+code+GAN+LSTM	0.0036±0.0013	0.0036±0.0012
B+code+GAN+LSTM+TC	**0.0030±0.0014**	**0.0029±0.0012**

Table 4. Ablation study on pose estimation for various versions of our method on KITTI sequence 09 and 10. B denotes baseline.

the temporal information actually improves depth.

As for pose estimation in Table 4, our baseline method performs much better than the other self-supervised VO approaches in literature (Table 2). This may mainly because of the joint use of depth and image for pose estimation (Eq. (5)). In addition, the accuracy is significantly improved by LSTM which incorporates historical information of multiple frames. The enforcement of trajectory consistency also brings about promising improvements in that it enforces geometric consistency among multiple pose estimations. Since depth is improved mainly on edges and details which takes up a small proportion, the accuracy gain is therefore limited. Yet the improved details are very important to RGBD matching for pose regression. Therefore, a slight increase in depth accuracy causes a big improvement in pose estimation.

5. Conclusions

We proposed an self-supervised VO framework that reduces accumulated errors over long sequence to achieve accurate pose and depth estimation. Benefited from spatial-temporal consistency among consecutive frames, the proposed framework incorporates historical information to reduce estimation errors in a self-supervised manner. In addition, we proposed to tackle VO as an self-supervised image generation task by means of a GAN paradigm. Our method outperforms both self-supervised and traditional VO baselines in literature, and ablation studies validate the effectiveness of each component of our framework.

In the future, we will extend our framework to unsupervised end-to-end SLAM. It is also worthwhile to investigate the code learned by our framework, which may help semantic segmentation, surface normal estimation and dense 3D reconstruction. In addition, developing an self-supervised online refinement technique to adaptively learn new environments on the fly is also an interesting issue of VO/SLAM and other 3D computer vision researches.

Acknowledgments. The work is supported by the National Key Research and Development Program of China (2017YFB1002601) and National Natural Science Foundation of China (61632003, 61771026).

References

[1] Filippo Aleotti, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia. Generative Adversarial Networks for Unsupervised Monocular Depth Prediction. In *ECCV*, 2018.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein Generative Adversarial Networks. In *ICML*, 2017.

[3] V Babu, Anima Majumder, Kaushik Das, Swagat Kumar, et al. A Deeper Insight into the UnDEMoN: Unsupervised Deep Network for Depth and Ego-Motion Estimation. *arXiv preprint arXiv:1809.00969*, 2018.

[4] V Madhu Babu, Kaushik Das, Anima Majumdar, and Swagat Kumar. UnDEMoN: Unsupervised Deep Network for Depth and Ego-Motion Estimation. In *IROS*, 2018.

[5] Dan Barnes, Will Maddern, Geoffrey Pascoe, and Ingmar Posner. Driven to Distraction: Self-Supervised Distractor Learning for Robust Monocular Visual Odometry in Urban Environments. In *ICRA*, 2018.

[6] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J Davison. CodeSLAM: Learning a Compact, Optimisable Representation for Dense Visual SLAM. In *CVPR*, 2018.

[7] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving. In *ICCV*, 2015.

[8] 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 *CVPR*, 2016.

[9] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning Optical Flow with Convolutional Networks. In *ICCV*, 2015.

[10] David Eigen, Christian Puhrsch, and Rob Fergus. Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network. In *NIPS*, 2014.

[11] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct Sparse Odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):611–625, 2018.

[12] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-Scale Direct Monocular SLAM. In *ECCV*, 2014.

[13] Gunnar Farnebck. Two-Frame Motion Estimation Based on Polynomial Expansion. In *Scandinavian Conference on Image Analysis*, 2003.

[14] Christian Forster, Simon Lynen, Laurent Kneip, and Davide Scaramuzza. Collaborative Monocular SLAM with Multiple Micro Aerial Vehicles. In *IROS*, 2013.

[15] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast Semi-Direct Monocular Visual Odometry. In *ICRA*, 2014.

[16] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue. In *ECCV*, 2016.

[17] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised Monocular Depth Estimation with Left-Right Consistency. In *CVPR*, 2017.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *NIPS*, 2014.

[19] Joao F Henriques and Andrea Vedaldi. MapNet: An Allocentric Spatial Memory for Mapping Environments. In *CVPR*, 2018.

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

[21] Phillip Isola, Junyan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *CVPR*, 2017.

[22] Ganesh Iyer, J Krishna Murthy, Gunshi Gupta, Madhava Krishna, and Liam Paull. Geometric Consistency for Self-Supervised End-to-End Visual Odometry. In *CVPR Workshops*, 2018.

[23] Christian Kerl, Jurgen Sturm, and Daniel Cremers. Dense visual SLAM for RGB-D cameras. In *IROS*, 2014.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for Stochastic Optimization. In *ICLR*, 2015.

[25] Georg Klein and David Murray. Parallel Tracking and Mapping on a Camera Phone. In *ISMAR*, 2009.

[26] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. UndeepVO: Monocular Visual Odometry through Unsupervised Deep Learning. In *ICRA*, 2018.

[27] 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*, 38(10):2024–2039, 2016.

[28] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Unsupervised Learning of Depth and Ego-Motion from Monocular Video Using 3D Geometric Constraints. In *CVPR*, 2018.

[29] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[30] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *ISMAR*, 2011.

[31] Jaesik Park, Qian Yi Zhou, and Vladlen Koltun. Colored Point Cloud Registration Revisited. In *ICCV*, 2017.

[32] Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan. PyTorch. `https://github.com/pytorch/pytorch`, 2017.

[33] 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 *CVPR*, 2017.

[34] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. DeepVO: Towards End-to-End Visual Odometry with Deep Recurrent Convolutional Neural Networks. In *ICRA*, 2017.

[35] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image Quality Assessment: from Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[36] Fei Xue, Qiuyuan Wang, Xin Wang, Wei Dong, Junqiu Wang, and Hongbin Zha. Guided Feature Selection for Deep Visual Odometry. In *ACCV*, 2018.

[37] Fei Xue, Xin Wang, Shunkai Li, Qiuyuan Wang, Junqiu Wang, and Hongbin Zha. Beyond Tracking: Selecting Memory and Refining Poses for Deep Visual Odometry. In *CVPR*, 2019.

[38] Nan Yang, Rui Wang, Jorg Stuckler, and Daniel Cremers. Deep Virtual Stereo Odometry: Leveraging Deep Depth Prediction for Monocular Direct Sparse Odometry. In *ECCV*, 2018.

[39] Zhichao Yin and Jianping Shi. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose. In *CVPR*, 2018.

[40] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid. Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction. In *CVPR*, 2018.

[41] Huizhong Zhou, Benjamin Ummenhofer, and Thomas Brox. DeepTAM: Deep Tracking and Mapping. In *ECCV*, 2018.

[42] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised Learning of Depth and Ego-Motion from Video. In *CVPR*, 2017.

[43] Jun Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In *ICCV*, 2017.