Unsupervised Collaborative Learning of Keyframe Detection and Visual Odometry Towards Monocular Deep SLAM

Lu Sheng[1*] Dan Xu[2*] Wanli Ouyang[3] Xiaogang Wang[4]
[1]College of Software, Beihang University, China [2]University of Oxford, UK
[3]The University of Sydney, SenseTime Computer Vision Research Group, Australia
[4]CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong, Hong Kong
lsheng@buaa.edu.cn, danxu@robots.ox.ac.uk, wanli.ouyang@sydney.edu.au, xgwang@ee.cuhk.edu.hk

Abstract

In this paper we tackle the joint learning problem of keyframe detection and visual odometry towards monocular visual SLAM systems. As an important task in visual SLAM, keyframe selection helps efficient camera relocalization and effective augmentation of visual odometry. To benefit from it, we first present a deep network design for the keyframe selection, which is able to reliably detect keyframes and localize new frames, then an end-to-end unsupervised deep framework further proposed for simultaneously learning the keyframe selection and the visual odometry tasks. As far as we know, it is the first work to jointly optimize these two complementary tasks in a single deep framework. To make the two tasks facilitate each other in the learning, a collaborative optimization loss based on both geometric and visual metrics is proposed. Extensive experiments on publicly available datasets (i.e. *KITTI raw dataset and its odometry split [12]) clearly demonstrate the effectiveness of the proposed approach, and new state-of-the-art results are established on the unsupervised depth and pose estimation from monocular video.*

1. Introduction

While perception of 3D geometric scenes is particularly important for interaction with real-world environments, as one important topic, visual simultaneous localization and mapping (SLAM) [10] has received emerging attention in recent years. However, due to the task complexity and limited annotated data, the power of deep learning is only partially explored on existing visual SLAM systems [5, 28].

In this work, we focus on techniques for monocular visual SLAM systems, which generally contains several subtasks, such as depth prediction and camera motion estimation for local 3D scene structure recovery, and keyframe selection and management for global map construction and localization. As an important part in monocular SLAM, the keyframe selection has been widely investigated in traditional approaches for aiding the visual odometry and scene matching [30, 9]. Although various evidences have shown that using deep models with geometric constraints clearly boosts the performance of depth, camera motion and optical flow estimation [51, 49], to our best knowledge, no existing work has considered deep learning based frameworks for the keyframe selection task.

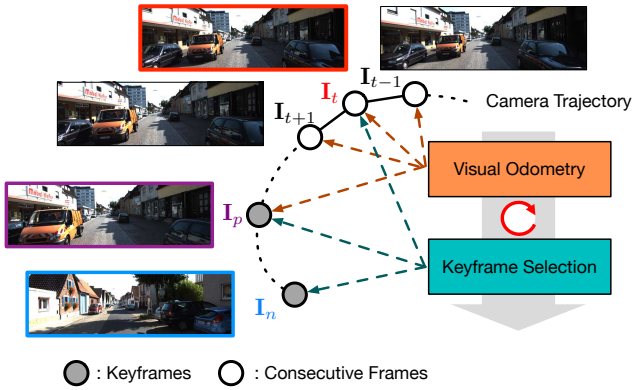

Figure 1. Illustration of our motivation: The keyframe selection and the visual odometry are intercorrelated in monocular visual SLAM. The keyframes can improve the depth prediction and camera motion estimation for the visual odometry, while inversely, the visual odometry facilitates more effective identification of keyframes. We expect that a joint learning of both tasks in a single deep model would make them benefit each other.

In this paper, we argue that a joint optimization of the keyframe selection and visual odometry should greatly benefit each other. Robust keyframe selection not only provides an efficient manner for fast localization and mapping, but also is particularly useful for effective refinement of the camera motion and depth predictions in the visual odometry task. Inversely, better visual odometry is able to facilitate more accurate keyframe identification. In addition,

*Equal contribution.

simultaneous learning of multiple tasks in deep learning has demonstrated its effectiveness in computer vision tasks such as detection and segmentation [4, 13]. It is thus a natural expectation that solving the keyframe selection and the visual odometry in a single deep network could also benefit from the advantage of the joint optimization.

Based on aforementioned observations, we propose an unsupervised deep model towards monocular SLAM, powered by three sub-networks to deal with three distinct but complementary tasks, *i.e.* keyframe selection, camera motion estimation and depth prediction. The keyframe selection sub-network learns a joint visual and geometric similarity between a pair of observed image and keyframe. If this similarity is below a threshold, the observation image is treated as a new keyframe, and is stored in a managed keyframe pool. The camera motion and depth prediction sub-networks learn to predict the depth of the observed image and relative motions from its nearby frames. To jointly learn these three tasks, we propose a collaborative learning strategy to predict a final similarity for the keyframe selection using a geometric metric estimated from the depth and the camera motion estimation networks, and a visual metric directly estimated from the keyframe selection network. A final ranking loss is added for different observation and keyframe pairs. By doing so, the whole network is trained in an unsupervised end-to-end fashion, and the three tasks constrain on each other based on their visual-geometric relationship for a better optimization of the whole model.

In summary, the contribution of this work is three-fold:

- We design a keyframe selection network, which is used to estimate a combinational similarity metric between visual and geometric cues. The learning of keyframes further provides extra supervision for the learning of the visual odometry networks.
- We propose a unified unsupervised deep learning framework to simultaneously learn the keyframe selection and visual odometry tasks in an end-to-end fashion. As far as we know, it is the first work to jointly optimize these two complementary components in a single deep model. A collaborative optimization loss is designed to enforce mutual constraints in between, enabling them to benefit each other in a joint optimization.
- We extensively demonstrate the effectiveness of the proposed approach on the KITTI raw dataset and its odometry split [12], showing the benefits of jointly learning and achieving new state-of-the art results on unsupervised monocular depth and camera motion estimation tasks.

2. Related Works

SLAM has been widely studied in recent years as a core 3D scene understanding technology. It can be roughly classified into stereo [35, 27, 15, 42], RGB-D [44, 17, 21] and monocular-based SLAM [8, 7]. We will review the most related monocular visual SLAM approaches.

Traditional Keyframe based Approaches Keyframe selection contains a detection step to identify a keyframe and a matching step for localization. The keyframe selection has been adopted in several state-of-the-art traditional SLAM approaches, such as RDSLAM [39] and ORB-SLAM [30, 31]. LSD-SLAM [8] presents a real-time visual SLAM system, which updates the keyframes by tracking the change of rigid pose, and correspondingly refines the depth map estimation. Forster *et al*. [9] applies a similar strategy to LSD-SLAM using direct tracking, while operates on semi-dense depth maps to obtain a high frame rate. More recently, Hsiao *et al*. [19] propose a keyframe-based SLAM approach based on dense plane extraction and matching, yielding superior performance on real-time SLAMs.

Traditional Visual Odometry based Approaches The monocular visual odometry estimates the 3D scene structure and ego-motion from 2D data with a monocular camera [37]. It mainly contains feature-based methods with salient feature tracking [33, 32], appearance-based methods with pixel-level image/patch matching [50, 38] and hybrid methods with a combination of the feature and appearance based strategies [34]. There are also other works exploring camera geometric modeling and regression model learning [16]. However, the traditional approaches mostly rely on hand-crafted representations or shallow models, which leads to inferior SLAM performance.

Supervised Deep Learning based Approaches To overcome the limitations of traditional approaches, more recent works focus on designing deep learning models to tackle the problem. Several supervised models have been proposed and significantly boost the performance of scene depth [45, 23, 46], camera pose [2] and scene flow estimation [26]. Eigen *et al.* [6] introduce a coarse to fine network structure with multi-scale fusion for depth prediction from single images. Kendall *et al.* [20] propose a PoseNet structure to address the 6-DoF camera relocalization problem. CNN-SLAM [40] detects keyframes and uses them to rectify the scale of the depth prediction, however, the keyframe detection is only based on an off-the-shelf method using hand-crafted features, and is not jointly learned with the other sub-tasks within a single deep model.

Unsupervised Deep Learning based Approaches Apart from the supervised models, there exists some unsupervised deep learning based approaches in the literature [24, 22, 48, 36]. Garg *et al*. [11] present an encoder-decoder disparity learning network utilizing view synthesis error for optimization. To consider mutual constraints from different views, Godard *et al*. [14] further introduce a two-branches reconstruction network and apply a left-right consistence loss to supervise each other. However, these approaches

only learn a single task in their models. SfMLearner [51] proposes to jointly unsupervised learning depth and camera pose from monocular videos using photometric synthesis loss from different nearby views. Upon SfM-Learner, GeoNet [49] further learns an optical flow task to tackle the non-rigid motion issue in the view reconstruction. Our model explores unsupervised learning from monocular videos and is more related to these two approaches, however, ours focuses on designing a keyframe selection network, and a probabilistic collaborative learning framework to make the keyframe selection and the visual-odometry benefit each other in a single deep model.

3. The Proposed Approach

We propose an end-to-end system aiming at jointly learning the keyframe selection and the visual odometry in a single deep network towards monocular SLAM. It primarily consists of a visual odometry and a keyframe selection modules implemented with neural networks. In the remainder of this section, we first introduce the designed deep keyframe selection and visual odometry modules, and then present how they are jointly learned in the proposed unsupervised collaborative learning framework.

3.1. Keyframe-based Visual Odometry

Our visual odometry model includes a monocular depth predictor $\mathcal{D}_{\phi_D}$ and a camera motion estimator $\mathcal{C}_{\phi_C}$ between a pair of frames. ϕ_D and ϕ_C are network parameters.

Network Specification For an image pair $\mathbf{I}_r$ and $\mathbf{I}_t$, $\mathcal{D}_{\phi_D}$ and $\mathcal{C}_{\phi_C}$ are defined as

$$\mathbf{D}_t = \mathcal{D}_{\phi_D}(\mathbf{I}_t), \quad \boldsymbol{\theta}_{t\to r} = \mathcal{C}_{\phi_C}(\mathbf{I}_t, \mathbf{I}_r), \tag{1}$$

where $\mathbf{D}_t$ is the predicted depth of $\mathbf{I}_t$ and $\boldsymbol{\theta}_{t\to r}$ is the camera ego-motion from the target image $\mathbf{I}_t$ to the reference image $\mathbf{I}_r$. The camera motion consists of a rotation vector $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^\top$ and a translation vector $\mathbf{t} = [t_x, t_y, t_z]^\top$. Our model follows a similar network structure as that in SfM-Learner [51], but our camera motion estimator just uses any two images $\mathbf{I}_t$ and $\mathbf{I}_r$ as its input, rather than consecutive frames. Thus our camera motion estimator is flexible and is not fixed to local adjacent frames.

Necessity of Keyframes State-of-the-art learning based visual odometry methods [51, 49] only explain small geometric changes, since they are learned by short-length consecutive frames (around $2 \sim 5$ frames). Thus they are usually failed to capture large geometric changes such as the case about the target image versus keyframes.

Thanks to the associated keyframe selection task, we find that keyframes are useful as additional training data to augment the geometry description of the visual odometry model. In this case, the camera motion estimator $\mathcal{C}_{\phi_C}$ has to capture more challenging motion patterns between the target image and the keyframes, and the depth predictor $\mathcal{D}_{\phi_D}$ must find accurate scene geometry to meet the geometric consistency between in between as well.

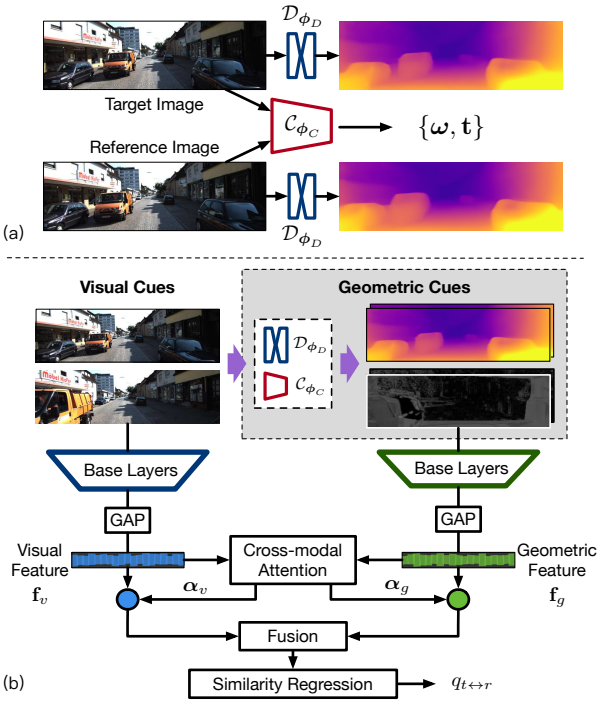

Figure 2. Network structure of the depth predictor $\mathcal{D}_{\phi_D}$, the camera motion estimator $\mathcal{C}_{\phi_C}$ and the keyframe selector $\mathcal{S}_{\phi_S}$.

3.2. Geometry-equipped Keyframe Selection

Keyframes record the most representative geometry maps or landmarks (dense depth, pose and *etc*.) among its neighboring frames. Its core function is a *keyframe selector* $\mathcal{S}_{\phi_S}$ that builds up keyframe sets by identifying a new keyframe when it includes considerably *geometric* changes or *visual* changes against the previous keyframes. It also concurrently localize any frame to its nearest keyframe (if exists) so as to fulfill the camera localization. ϕ_S indicates the network parameters of the keyframe selector.

Network Specification Assume the target image as $\mathbf{I}_t$ and the reference image as $\mathbf{I}_r$ (possibly the existing keyframes). The keyframe selector is to measure whether $\mathbf{I}_t$ and $\mathbf{I}_r$ are similar both in visual and geometric viewpoints. $\mathcal{S}_{\phi_S}$ has a two-stream structure and adaptively combines the visual and geometric similarities for the final decision, as depicted in Fig. 2, in which (1) the visual stream applies the concatenated $\mathbf{I}_t$ and $\mathbf{I}_r$ as its input. (2) the geometric stream receives the channel-wise concatenation of a series of geometric data obtained from the visual odometry module. It includes the predicted depth maps $\mathbf{D}_t$ and $\mathbf{D}_r$, and warping residual maps $\Delta\mathbf{I}_{t\leftarrow r}$ from $\mathbf{I}_r$ to $\mathbf{I}_t$ and $\Delta\mathbf{I}_{r\leftarrow t}$ *vice versa*.

The warping residual maps (take $\Delta\mathbf{I}_{t\leftarrow r}$ as an example) are

$$\Delta\mathbf{I}_{t\leftarrow r}(\mathbf{x}) = |\mathbf{I}_r(\mathcal{W}(\mathbf{x}; \mathbf{D}_t, \boldsymbol{\theta}_{t\to r})) - \mathbf{I}_t(\mathbf{x})| \quad (2)$$

where $\mathcal{W}(\mathbf{x}; \mathbf{D}_t, \boldsymbol{\theta}_{t\to r})$ is the rigid warping field explained by the predicted depth $\mathbf{D}_t$ and the camera motion $\boldsymbol{\theta}_{t\to r}$. In summary, the keyframe selector is

$$q_{t\leftrightarrow r} = \mathcal{S}_{\boldsymbol{\phi}_S}(\mathbf{I}_t, \mathbf{I}_r; \mathbf{D}_t, \mathbf{D}_r, \Delta\mathbf{I}_{t\leftarrow r}, \Delta\mathbf{I}_{r\leftarrow t}), \quad (3)$$

where the similarity $q_{t\leftrightarrow r}$ is robust to the order of $\mathbf{I}_t$ and $\mathbf{I}_r$.

Both streams share a same network architecture but not their network parameters. The base layers in each stream are copied from ResNet-18 [18], while each of them is followed by a global average pooling and several fully-connected layers. The extracted visual feature $\mathbf{f}_c$ and geometric feature $\mathbf{f}_g$ from each stream are fused together with the help of cross-modal attention. The attentions $\boldsymbol{\alpha}_c$ and $\boldsymbol{\alpha}_g$ are learned via an additional fully-connected layer using concatenated $\mathbf{f}_c$ and $\mathbf{f}_g$ as the input. The attended visual features $\mathbf{f}_c \circ \boldsymbol{\alpha}_c$ and $\mathbf{f}_g \circ \boldsymbol{\alpha}_g$ are further combined using fully-connected layers to generate the final similarity score.

3.3. Keyframe Online Updating and Management

We design an online keyframe updating and management strategy to maintain a key frame pool $\mathcal{P}^{\mathcal{K}}$ during the training stage. At the beginning of the training, $\mathcal{P}^{\mathcal{K}}$ uses several randomly selected frames as initialization. Each keyframe $\mathcal{F}_k^{\mathcal{K}}$ is represented with a three-tuple containing the frame index t_k corresponding to input video sequence, the RGB image $\mathbf{I}_k^{\mathcal{K}}$ and the depth estimation map $\mathbf{D}_k^{\mathcal{K}}$ produced from the visual odometry network, *i.e.* $\mathcal{F}_k^{\mathcal{K}} = \{t_k, \mathbf{I}_k^{\mathcal{K}}, \mathbf{D}_k^{\mathcal{K}}\}$ and $\mathcal{P}^{\mathcal{K}} = \{{\mathcal{F}_k}^{\mathcal{K}}\}_{k=1}^{K}$ where K is the total number of key frames. The keyframe updating consists inserting and merging operations. After several training iterations, a determination on an input target frame is conducted. We used 200 iterations in our implementation. If its similarity scores between the nearest keyframes are above a threshold, it would be insert into $\mathcal{P}_{\mathcal{K}}$. And after each epoch, we start merging the selected keyframes given by a trained better keyframe matching network. Adjacent keyframes in $\mathcal{P}_{\mathcal{K}}$ are organized into pairs and are passed into the network for similarity measurement. If the two are similar enough, only one of them is kept. The keyframe depth estimation is also used to help the optimization of visual odometry sub-network. The depth map of the closest keyframe to the target frame, is used to refine the depth prediction from the depth estimation net via a weighted averaging operation. In the testing phase, the latest keyframe always compares the target frame, if their dissimilarity is above a threshold, new keyframe is inserted into $\mathcal{P}^{\mathcal{K}}$. Please check Fig. 3(a) for illustration.

3.4. Unsupervised Collaborative Learning

As aforementioned, the keyframe selection and visual odometry are complementary to each other. It is thus beneficial to collaboratively learn these tasks together. But how to merge them into a unified learning framework is not trivial and requires a special design of the training procedure. As shown in Fig. 3(b), the proposed collaborative learning scheme will be depicted in details in the following text.

Training Data Preparation. Keyframe selection and visual odometry require different training data constructions as they follow different learning logics. In each training example, we have a short training sequence $\mathcal{I}_s$ ($|\mathcal{I}_s| = 3$), in which the center frame is the target image $\mathbf{I}_t$. And we gather one intra-class sample $\mathbf{I}_p$ that is picked as the temporally nearest keyframe in the keyframe set $\mathcal{P}^{\mathcal{K}}$, and select a second temporally nearest sample $\mathbf{I}_n$ as the hard negative sample. Therefore, the training example is $\mathcal{I}_{\mathcal{K}} = \{\mathcal{I}_s, \mathbf{I}_p, \mathbf{I}_n\}$.

Optimization Loss of Visual Odometry. The visual odometry module is learned among a combined training image set $\mathcal{I}_{vo} = \{\mathcal{I}_s, \mathbf{I}_p\}$. For each image pairs $\{\mathbf{I}_t, \mathbf{I}_r\}$ in $\mathcal{I}_{vo}$, we optimize the photometric consistency to both images, within the regions that rigid correspondences exist:

$$\mathcal{L}_{pc} = \sum_{\{\mathbf{I}_t, \mathbf{I}_r\}\in\mathcal{I}_{vo}} \sum_{\mathbf{x}} (1 - \mathbf{M}_t(\mathbf{x})) \cdot \rho(\mathbf{I}_{t\leftarrow r}(\mathbf{x}), \mathbf{I}_t) + (1 - \mathbf{M}_r(\mathbf{x})) \cdot \rho(\mathbf{I}_{r\leftarrow t}(\mathbf{x}), \mathbf{I}_r) + (\mathbf{M}_r(\mathbf{x}) + \mathbf{M}_t(\mathbf{x})) \cdot \tau \quad (4)$$

where $\rho(x, y) = \frac{\alpha}{2}(1 - \text{SSIM}(x, y)) + (1 - \alpha)\sigma(x - y)$ is a robust perceptual image similarity measurement. SSIM is the structural similarity index [43] and $\sigma(x) = (x^2 + \varepsilon^2)^{0.45}$ is the robust Charbonnier loss [3]. $\mathbf{I}_{t\leftarrow r}(\mathbf{x}) = \mathbf{I}_r(\mathcal{W}(\mathbf{x}; \mathbf{D}_t, \boldsymbol{\theta}_{t\to r}))$ is the backward warped reference image, and $\mathbf{I}_{r\leftarrow t}$ is defined as the backward warped target image $\mathbf{I}_{r\leftarrow t}(\mathbf{x}) = \mathbf{I}_t(\mathcal{W}(\mathbf{x}; \mathbf{D}_r, \boldsymbol{\theta}_{r\to t}))$. Note that $\boldsymbol{\theta}_{r\to t}$ is the inverse motion of $\boldsymbol{\theta}_{t\to r}$, which is calculated analytically but not through the camera motion estimator $\mathcal{C}(\mathbf{I}_r, \mathbf{I}_t)$ again. The non-rigid mask $\mathbf{M}_t$ in $\mathbf{I}_t$ is generated by detecting the regions where the cycle consistency between the bi-directional warping fields, *i.e.*, $\Delta\mathcal{W}_t(\mathbf{x}) = |\mathcal{W}(\mathbf{x}; \mathbf{D}_t, \boldsymbol{\theta}_{t\to r}) + \mathcal{W}(\mathcal{W}(\mathbf{x}; \mathbf{D}_t, \boldsymbol{\theta}_{t\to r}); \mathbf{D}_r, \boldsymbol{\theta}_{r\to t})|$ is violated. The non-rigid mask $\mathbf{M}_r$ in $\mathbf{I}_r$ is calculated in a similar way to threshold $\Delta\mathcal{W}_r(\mathbf{x}) = |\mathcal{W}(\mathbf{x}; \mathbf{D}_r, \boldsymbol{\theta}_{r\to t}) + \mathcal{W}(\mathcal{W}(\mathbf{x}; \mathbf{D}_r, \boldsymbol{\theta}_{r\to t}); \mathbf{D}_t, \boldsymbol{\theta}_{t\to r})|$. The threshold is scaled according to the per-pixel magnitude of the warping fields, similarly as [29]. The additional constant τ is added to remove trivial solutions that any pixel is non-rigid.

To enhance the geometric consistency, we enforce the cycle consistency in the rigid regions as well

$$\mathcal{L}_{cc} = \sum_{\{\mathbf{I}_t, \mathbf{I}_r\}\in\mathcal{I}_{vo}} \sum_{\mathbf{x}} (1 - \mathbf{M}_t(\mathbf{x})) \cdot \Delta\mathcal{W}_t(\mathbf{x}) + (1 - \mathbf{M}_r(\mathbf{x})) \cdot \Delta\mathcal{W}_r(\mathbf{x}). \quad (5)$$

The depth maps are further smoothed by $\mathcal{L}_{ds} = \sum_{\mathbf{I}_t\in\mathcal{I}_{vo}} \sum_{\mathbf{x}} |\nabla d_t(\mathbf{x})|^{\top} \exp(-|\nabla\mathbf{I}_t(\mathbf{x})|)$, which is an

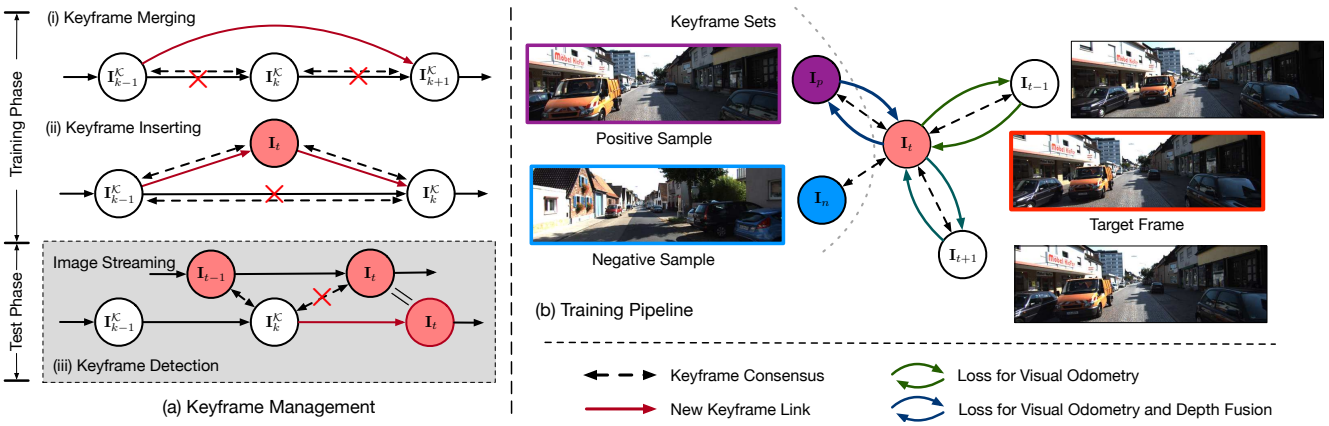

Figure 3. (a) The keyframe management, including keyframe merging and keyframe inserting in the training phase, and keyframe set construction in the testing phase. (b) The collaborative learning scheme. The training image tuples consists of consecutive frames around the target image, and two randomly sampled intra-class image $\mathbf{I}_p$ and inter-class image $\mathbf{I}_n$. The *keyframe selection* task uses the complete training tuple, but the *visual odometry* does not use the inter-class sample.

edge-aware disparity smoothness loss, where the disparity is simply defined as $d_t(\mathbf{x}) = 1/\mathbf{D}_t(\mathbf{x})$.

Optimization Loss for Keyframe Selection The learning of keyframe selection intensively applies the triplet losses to measure the similarity between frames. Specifically, two kinds of triplets are constructed: (1) $\langle \mathbf{I}_t, \mathbf{I}_s, \mathbf{I}_p \rangle$, in which $\mathbf{I}_s$ is one image in the training sequence $\mathcal{I}_s$, (2) $\langle \mathbf{I}_t, \mathbf{I}_s, \mathbf{I}_n \rangle$, in which $\mathbf{I}_s$ is one image in the visual odometry training image set $\mathcal{I}_{vo}$. The first triplet is used to rank the similarities w.r.t. $\mathbf{I}_t$ among samples inside the interval around the target image $\mathbf{I}_t$, where $\mathbf{I}_t$ should be more similar to $\mathbf{I}_s$ than $\mathbf{I}_p$ with a small margin γ_p. The second one ranks the similarities with a large margin γ_n, it suggests a much larger similarity between $\mathbf{I}_t$ and any sample in $\mathcal{I}_{vo}$ than the negative sample $\mathbf{I}_n$. To this end, the keyframe loss is written as

$$\mathcal{L}_{kf} = \sum_{\mathbf{I}_s \in \mathcal{I}_s / \{\mathbf{I}_t\}} \max\{0, \gamma_p - q_{t \leftrightarrow s} + q_{t \leftrightarrow p}\} + \sum_{\mathbf{I}_s \in \mathcal{I}_{vo} / \{\mathbf{I}_t\}} \max\{0, \gamma_n - q_{t \leftrightarrow s} + q_{t \leftrightarrow n}\}. \quad (6)$$

The similarity score is generated according to our keyframe selector in Eq. (3). The intra-sample margin is $\gamma_p = 0.1$ and the inter-sample margin is $\gamma_n = 0.8$.

Overall Optimization Objectives The final loss for our collaborative learning is a weighted combination of the aforementioned losses, written as

$$\mathcal{L}_{total} = \lambda_{pc}\mathcal{L}_{pc} + \lambda_{cc}\mathcal{L}_{cc} + \lambda_{ds}\mathcal{L}_{ds} + \lambda_{kf}\mathcal{L}_{kf}. \quad (7)$$

The weights are to balance the contribution of each term. In our experiments, we set $\lambda_{pc} = 1.0, \lambda_{cc} = 0.05, \lambda_{ds} = 0.5$ and $\lambda_{kf} = 1.0$. Note that our depth predictor $\mathcal{D}_{\phi_D}$ uses multi-scale depth predictions to release the local gradient issue [51], thus the losses $\mathcal{L}_{pc}$, $\mathcal{L}_{cc}$ and $\mathcal{L}_{ds}$ are also applied in coarser scales, but the weights are decayed accordingly.

4. Experiments

4.1. Experimental Setup

Network Architecture. Our model mainly contains three components, the depth predictor $\mathcal{D}_{\phi_D}$, the camera motion estimator $\mathcal{C}_{\phi_C}$ and the keyframe selector $\mathcal{S}_{\phi_S}$. The depth predictor follows the skip-connected encoder-decoder structure as SfMLearner [51], and outputs 4-scale depth predictions. The camera motion estimator regresses the 6-DoF camera motions by 8 convolution layers followed by a global average pooling, as the structure in [51]. The structure of the keyframe selector has two parallel branches, its network specification is depicted in Sec. 3.2. We adopt batch normalization and ReLU activation function after all the convolution layers except the output layers.

Datasets. We train our system on the train split by Eigen *et al.* [6] on the KITTI raw dataset with all static frames excluded. This dataset contains stereo views, and we use them independently. The train/val ratio is $9:1$, following Zhou *et al.* [51]. To test the performance of our visual odometry and keyframe selection, we also transfer our system onto the KITTI odometry dataset. We employ the $00 \sim 08$ sequences for training, and the $09 \sim 10$ for testing.

Training Details. Our experiments are conducted using the TensorFlow framework [1]. We train our model in an end-to-end fashion with our special designed training data preparation. During training, we resize the image sequences to a resolution of 128×416, and perform several preprocessing tricks such as random cropping and resizing, and random brightness [49, 51]. The network is trained by Adam solver with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is simply fixed at 0.0002 and the batch size is 8. The network is trained on a single NVIDIA Titan X GPU. The training process typically takes around 30 epochs.

Evaluation Protocol. The depth prediction performance on

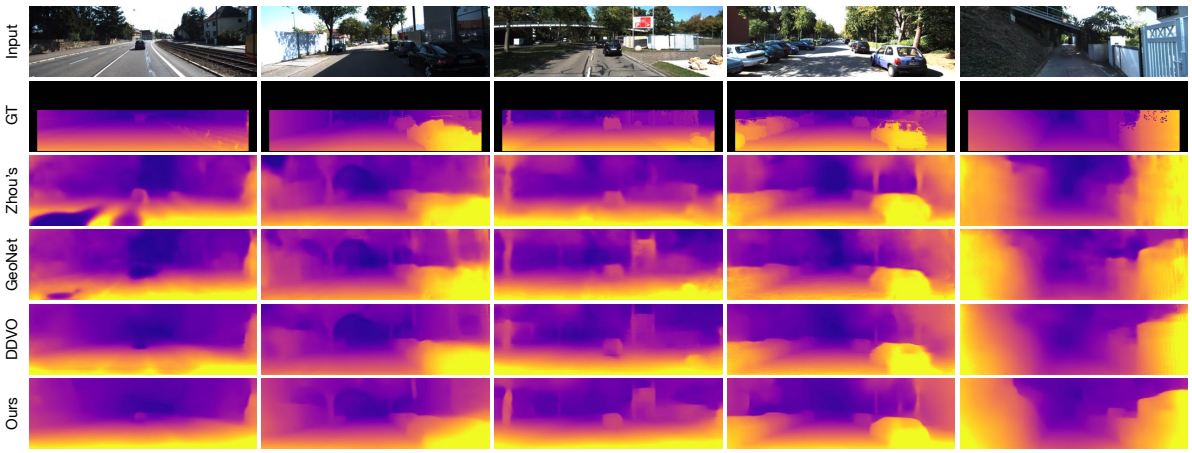

Figure 4. Monocular depth prediction comparison with Zhou *et al*. [51], GeoNet [49] and DDVO [41].The ground-truth is interpolated for visualization. Our method captures more geometric details, preserves the structure consistency and avoids artifacts in texture-less area.

Method	Setting	Cap	Data	abs rel	sq rel	RMSE	RMSE(log)	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen *et al*. [6]	depth-gt	80m	-	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu *et al*. [25]	depth-gt	80m	-	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Godard *et al*. [14]	stereo	80m	-	0.148	1.344	5.927	0.247	0.803	0.922	0.964
Xu *et al*. [47]	depth-gt	80m	-	0.133	0.896	4.718	0.195	0.828	0.952	0.984
Zhou *et al*. [51]	mono	80m	$\mathcal{I}_s$	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Zhou *et al*. [51]*	mono	80m	$\mathcal{I}_s$	0.183	1.595	6.709	0.270	0.734	0.902	0.959
GeoNet [49]	mono	80m	$\mathcal{I}_s$	0.164	1.303	6.090	0.247	0.765	0.919	0.968
DDVO [41]	mono	80m	$\mathcal{I}_s$	0.151	1.257	**5.583**	0.228	**0.810**	0.936	0.974
Klodt *et al*. [22]	mono/SfM	80m	$\mathcal{I}_s$	0.166	1.490	5.998	–	0.778	0.919	0.966
Ours w/o CC	mono	80m	$\mathcal{I}_\mathcal{K}$	0.168	1.259	5.937	0.247	0.755	0.920	0.969
Ours	mono	80m	$\mathcal{I}_\mathcal{K}$	**0.139**	**1.021**	5.418	**0.209**	0.803	**0.937**	**0.976**
Godard *et al*. [14]	stereo	50m	-	0.140	0.976	4.471	0.232	0.818	0.931	0.969
Garg *et al*. [11]	pose-gt	50m	-	0.169	1.080	5.104	0.273	0.740	0.904	0.962
Zhou *et al*. [51]	mono	50m	$\mathcal{I}_s$	0.201	1.391	5.181	0.264	0.696	0.900	0.966
GeoNet [49]	mono	50m	$\mathcal{I}_s$	0.157	0.990	4.600	0.231	0.781	0.931	0.974
Ours	mono	50m	$\mathcal{I}_\mathcal{K}$	**0.131**	**0.805**	**4.021**	**0.202**	**0.820**	**0.947**	**0.982**

Table 1. Monocular depth prediction results on the KITTI dataset [12] using the split of Eigen *et al*. [6]. We reports 7 metrics as suggested by Eigen *et al*. [6]. We also indicate the training setting, and the training data structure. $\mathcal{I}_s$ means consecutive frames, $\mathcal{I}_\mathcal{K}$ is our keyframe augmented sequences. **Bold** means the overall best results. "w/o CC" means the our visual odometry module trained without cycle consistency. *Updated results provided on the website of Zhou *et al*. [51]. In setting, depth-gt and post-gt mean that the methods require depth and pose groundtruth in a supervised setting.

is evaluated on the 697 images from the test split of Eigen *et al*. [6], which covers 29 scenes in the KITTI raw dataset. Following [51], the predicted depth maps are scaled with a factor by matching its median to its ground-truth data, *i.e*., $\mathbf{D}_{pred} = \text{median}(\mathbf{D}_{gt})/\text{median}(\mathbf{D}_{pred})$. We use the same depth evaluation metrics as in Eigen *et al*. [6]. Note that most reference monocular depth prediction methods use consecutive $|\mathcal{I}_s| = 3$ frames, but our method requires two additional intra-/inter-class samples, such as the set $\mathcal{I}_K$.

The camera motion evaluation is on the $09 \sim 10$ sequences in the KITTI odometry split. Following [51], all of the reported results are evaluated in terms of 5-frame snippets. To resolve scale ambiguities that frequently occur in monocular visual odometry or SLAM systems, we adjust the scaling factors of the results to optimally align with the ground-truth trajectories. We use Absolute Trajectory Error (ATE) to evaluate trajectory drift for 5-frame snippet.

For keyframe selection evaluation, we gather snippets whose starting frame is a reference keyframe and a pseudo-GT keyframe is located in the middle of the snippet. The pseudo-GT keyframe is detected if the ratio of the overlapping area with the reference keyframe is just below 50%, in which the overlapping area is defined by the ground-truth camera motion and interpolated depth maps. We apply this strategy to the KITTI odometry test split.

4.2. Overall Performance Analysis

Performance of Monocular Depth Estimation. As shown in Tab. 1, if truncating the depth predictions by 80m, our proposed unsupervised approach outperforms all the com-

Method	Absolute Trajectory Error	
	sequence 09	sequence 10
ORB-SLAM (full)	0.014 ± 0.008	0.012 ± 0.011
ORB-SLAM (short)	0.064 ± 0.141	0.064 ± 0.130
Mean Odom.	0.032 ± 0.026	0.028 ± 0.023
Zhou *et al*. [51]	0.021 ± 0.017	0.020 ± 0.015
Zhou *et al*. [51]*	0.016 ± 0.009	0.013 ± 0.009
GeoNet [49]	0.012 ± 0.007	0.012 ± 0.009
Klodt *et al*. [22]	0.014 ± 0.007	0.013 ± 0.009
Ours$^{\times}$	$\mathbf{0.012 \pm 0.006}$	$\mathbf{0.010 \pm 0.008}$

Table 2. Absolute Trajectory Error (ATE) on the KITTI odometry test split averaged over all 5-frame snippets (lower is better). $^{\times}$Our method does not trained by 5-frame snippets but 3-frame snippets with two additional intra-/inter-samples. *Updated results.

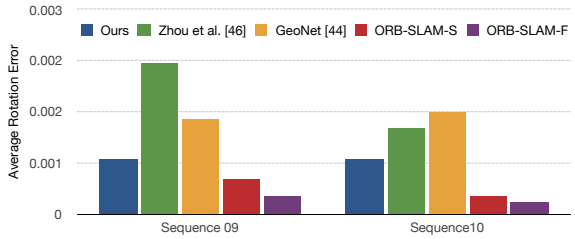

Figure 5. Average rotation errors on the KITTI odometry test split (lower is better). ORB-SLAM-S and ORB-SLAM-F is the short and the full of ORB-SLAM respectively.

pared unsupervised monocular depth prediction methods on most of the evaluation metrics, including [51, 49, 22], and even some recent methods with calibrated stereo data [11, 14] or directly supervised by ground-truth depth [25, 6]. DDVO [41] has marginal improvements on the scores $\delta < 1.25$ and RMSE, but our method has a significant gain over the squared relative difference (sq rel) from 1.257 to 1.021. If truncating the predictions by 50m, our model achieves the best performance on all the metrics.

Performance of Camera Pose Estimation. We compare our method with a traditional monocular SLAM system named ORB-SLAM (full) [30] and its local version ORB-SLAM (short) for 5-frame snippets, their results are borrowed from the website of Zhou *et al*. [51]. We also compare with SfMLearner [51] and GeoNet [49]. As in Tab. 2, our method outperforms a naïve baseline (mean odometry) and the conventional method ORB-SLAM (short) and ORB-SLAM (full). With respect to deep learning based approaches, our camera motion estimator is better than SfMLearner proposed by Zhou [51], but its performance is slightly inferior to GeoNet [49]. We believe this gap could be eliminated if our model is trained by longer snippets.

We also show the average rotation errors over all 5-frame snippets, in which the error is calculated as ℓ_2 norm between the rotation angles from the predictions and the ground-truths, as shown in Fig. 5. Although our method was only trained on shorter sequences, its predicted rotations are more accurate to the other learning based approaches [51, 49], which reveals the significance of the keyframes in helping regularize the odometry learning, especially the case with geometric changes from rotations. Note that two variants of ORB-SLAM offer better rotation predictions than learning based models. That is probably because that the results of ORB-SLAM gathered from Zhou *et al*. [51] are shorter than 5 frames and thus only contain smaller camera motions.

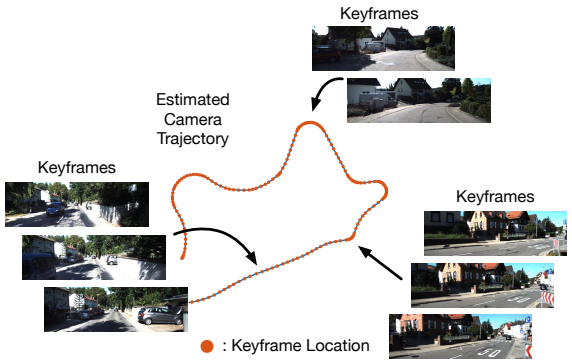

Figure 6. Keyframe selection accompanied with the visual odometry. The test sequence is 09 in KITTI odometry test split.

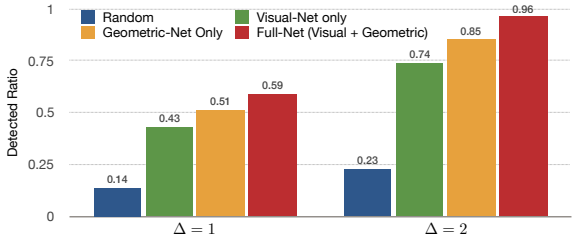

Figure 7. Detection ratio evaluation of Keyframe detection combined with the visual odometry. The test sequence is 09 in KITTI odometry test split.

Method	Error (lower is better)			
	rel	sq rel	RMSE	RMSE (log)
SFM-Learner [51] w/o KFS	0.208	1.768	6.856	0.283
Ours w/o KFS	0.181	1.587	6.689	0.264
Ours w/ KFS (off-line)	0.171	1.389	6.237	0.251
Ours w/ KFS (fixed-length)	0.151	1.127	5.941	0.223
Ours w/ KFS (online updating)	**0.139**	**1.021**	**5.418**	**0.209**

Table 3. Quantitative comparison of different variants of the proposed approach w.r.t. the error evaluation metrics on the task of monocular depth estimation. KFS means key-frame selection.

Method	Absolute Trajectory Error	
	sequence 09	sequence 10
SFM-Learner [51] w/o KFS	0.021 ± 0.017	0.020 ± 0.015
Ours w/o KFS	0.018 ± 0.012	0.017 ± 0.012
Ours w/ KFS (off-line)	0.015 ± 0.008	0.014 ± 0.011
Ours w/ KFS (fixed-length)	0.014 ± 0.007	0.012 ± 0.009
Ours w/ KFS (online updating)	$\mathbf{0.012 \pm 0.006}$	$\mathbf{0.010 \pm 0.008}$

Table 4. Quantitative comparison of different variants of the proposed approach w.r.t Absolute Trajectory Error (ATE) on the KITTI odometry test split.

Performance of Keyframe Selection. We also give a few

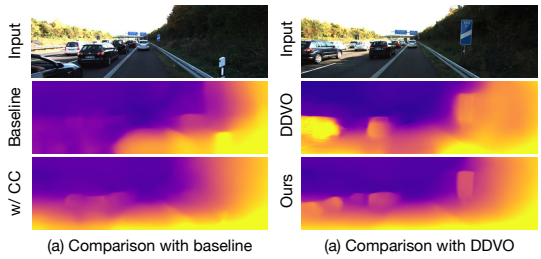

Figure 8. (a) Cycle consistency improves the geometry reliability of the predicted depth maps. Some examples of DDVO [41] (b) show texture-copying artifacts and unwanted texture blurs.

exemplar experiments on the quality of our keyframe selection module. First, we show the selected keyframe set when we execute the complete visual odometry to the test sequence 09 in the KITTI odometry test split. The selected keyframes are usually uniformly distributed when the car drives straight along the street, such as the keyframes shown in the left of Fig. 6. However, when the car turns left/right, the sudden geometric changes induce large visual dissimilarity in the captured frames, and thus our keyframe selection encourages more frequent keyframes in a short interval at the turning corner. It also reveals that our keyframe selection is more sensitive to geometry-based visual changes. We also qualitatively test our keyframe selector by reporting the ratio that the detected keyframes are within a fixed range $[-\Delta, \Delta]$ around the pseudo-GT keyframes, as shown in Fig. 7. The proposed keyframe selector combines the merits from both the visual and the geometric cues, and improves their sole models with a large margin.

Qualitative Evaluation. We compare our predicted depth maps with those by Zhou *et al*. [51], GeoNet [49] and DDVO [41], as in Fig. 4. The ground-truth depth maps are re-projected sparse point clouds from velodyne laser scanner. The proposed method has the best visual quality among the prior arts. It can successfully recover reliable depths inside challenging regions (*e.g*., texture-less area in the first column of Fig. 4), preserve piece-wise smooth structural details but not introduce texture-mapping from the input image. DDVO has the most comparable performance and more often resolves small objects, but it usually suffers severe texture-mapping artifacts, as show in Fig 8(b).

4.3. Model Component Analysis

Baseline Models. To evaluation the performance effect of different modules, we have several baselines: (i) SFM-Learner [51], which does not use any keyframe selection in the visual odometry; (ii) Ours w/o KFS, which improves the performance over [51], and still not using keyframe selection; (iii) Ours w/ KFS (off-line), which pretrains the keyframe selection sub-network using only the visual clue, and produce a set of keyframes for training with the visual odometry sub-network; (iv) Ours w/ KFS (fixed-length), which use a fixed frame length to determine the keyframe, and the keyframe selection sub-network is jointly optimized with the visual odometry sub-network; (iv) Ours w/ KFS (online updating) which uses the proposed keyframe management strategy to online update the keyframe pool and the multiple sub-tasks are jointly learned.

Effect of Keyframe Selection on Visual Odometry. Tab. 3 and 4 show the results of different baseline models on monocular depth and pose estimation tasks. It can be observe that, even if we use off-line keyframe information, we can still improve the performance on both depth and pose estimations. By jointly learning two tasks, especially employing the online keyframe updating, a clear performance gain is obtained, demonstrating the effectiveness of the proposed keyframe detection onto the task of visual odometry.

Effect of Visual Odometry on Keyframe Selection. Fig. 5 shows the average rotation errors of different methods on the KITTI odometry. To compare with our direct competitor [51], our model with keyframe selection significantly outperforms their method by reducing the errors with a large margin, which means that the visual odometry network provides better geometric output helping learning better keyframe detector, confirming our initial intuition.

Effectiveness of Cycle Consistency. We also conduct a piece of ablation study about the cycle consistency, as shown in Tab. 1. Without cycle consistency, the learning of our depth predictor is similar to Zhou *et al*. [51] but with additional long-term connections from "keyframes". Its performance is superior to Zhou *et al*. [51] and comparable to recent advanced methods, showing the advances of keyframes in helping visual odometry module. Cycle consistency clearly increases the prediction reliability, as shown in Fig. 8(a), and it boosts the quantitative results of our model to a large margin. But we need to mention that the cycle consistency may not optimal in detecting non-rigid motion regions, thus we may inevitably find depth distortions around moving objects, such as cars in Fig. 4.

5. Conclusion

In this paper we have proposed a learning approach towards monocular visual SLAM. In detail, we designed a deep network for the keyframe selection, which is able to to detect keyframes, manage the keyframes and localize new frames. And we further proposed an end-to-end unsupervised learning framework to simultaneously optimize the keyframe selection and the visual odometry tasks in a single deep model. To constrain and benefit each task during the network learning, a unsupervised collaborative learning strategy was designed. We clearly demonstrated the effectiveness of the proposed approach on KITTI raw and KITTI Odometry datasets with a significant gain over the baseline models, and created new state-of-the-art results on depth and pose estimation from monocular videos.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016. 5

[2] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *CVPR*, 2018. 2

[3] Pierre Charbonnier, Laure Blanc-Féraud, Gilles Aubert, and Michel Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *ICIP*, volume 2, pages 168–172. IEEE, 1994. 4

[4] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *CVPR*, 2016. 2

[5] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Toward geometric deep SLAM. *arXiv preprint arXiv:1707.07410*, 2017. 1

[6] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 2, 5, 6, 7

[7] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *TPAMI*, 40(3):611–625, 2017. 2

[8] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *ECCV*, 2014. 2

[9] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *ICRA*, 2014. 1, 2

[10] Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43(1):55–81, 2015. 1

[11] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016. 2, 6, 7

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *IJRR*, 32(11):1231–1237, 2013. 1, 2, 6

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *TPAMI*, 38(1):142–158, 2016. 2

[14] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 2, 6, 7

[15] Ruben Gomez-Ojeda, David Zuñiga-Noël, Francisco-Angel Moreno, Davide Scaramuzza, and Javier Gonzalez-Jimenez. PL-SLAM: a stereo slam system through the combination of points and line segments. *arXiv preprint arXiv:1705.09479*, 2017. 2

[16] Vitor Guizilini and Fabio Ramos. Semi-parametric models for visual odometry. In *ICRA*, 2012. 2

[17] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J Davison. A benchmark for rgb-d visual odometry, 3D reconstruction and SLAM. In *ICRA*, 2014. 2

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4

[19] Ming Hsiao, Eric Westman, Guofeng Zhang, and Michael Kaess. Keyframe-based dense planar SLAM. In *ICRA*, 2017. 2

[20] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-DOF camera relocalization. In *ICCV*, 2015. 2

[21] Christian Kerl, Jürgen Sturm, and Daniel Cremers. Robust odometry estimation for RGB-D cameras. In *ICRA*, pages 3748–3754. IEEE, 2013. 2

[22] Maria Klodt and Andrea Vedaldi. Supervising the new with the old: learning SFM from SFM. In *ECCV*, 2018. 2, 6, 7

[23] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 2

[24] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. UnDeepVO: Monocular visual odometry through unsupervised deep learning. In *ICRA*, 2018. 2

[25] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian D Reid. Learning depth from single monocular images using deep convolutional neural fields. *TPAMI*, 38(10):2024–2039, 2016. 6, 7

[26] 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 *CVPR*, 2016. 2

[27] Christopher Mei, Gabe Sibley, Mark Cummins, Paul Newman, and Ian Reid. RSLAM: A system for large-scale mapping in constant-time using stereo. *IJCV*, 94(2):198–214, 2011. 2

[28] Christopher Mei, Gabe Sibley, Mark Cummins, Paul M Newman, and Ian D Reid. A constant-time efficient stereo SLAM system. In *BMVC*, 2009. 1

[29] Simon Meister, Junhwa Hur, and Stefan Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, New Orleans, Louisiana, Feb. 2018. 4

[30] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular SLAM system. *TRO*, 31(5):1147–1163, 2015. 1, 2, 7

[31] 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*, 33(5):1255–1262, 2017. 2

[32] Oleg Naroditsky, Xun S Zhou, Jean Gallier, Stergios I Roumeliotis, and Kostas Daniilidis. Two efficient solutions for visual odometry using directional correspondence. *TPAMI*, 34(4):818–824, 2012. 2

[33] David Nistér, Oleg Naroditsky, and James Bergen. Visual odometry for ground vehicle applications. *JFR*, 23(1):3–20, 2006. 2

[34] Navid Nourani-Vatani and Paulo Vinicius Koerich Borges. Correlation-based visual odometry for ground vehicles. *JFR*, 28(5):742–768, 2011. 2

[35] Lina M Paz, Pedro Piniés, Juan D Tardós, and José Neira. Large-scale 6-DOF SLAM with stereo-in-hand. *TRO*, 24(5):946–957, 2008. 2

[36] Andrea Pilzer, Dan Xu, Mihai Puscas, Elisa Ricci, and Nicu Sebe. Unsupervised adversarial depth estimation using cycled generative networks. In *3DV*. IEEE, 2018. 2

[37] Davide Scaramuzza and Friedrich Fraundorfer. Visual odometry [tutorial]. *IEEE Robotics & Automation Magazine*, 18(4):80–92, 2011. 2

[38] Davide Scaramuzza and Roland Siegwart. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *TRO*, 24(5):1015–1026, 2008. 2

[39] Wei Tan, Haomin Liu, Zilong Dong, Guofeng Zhang, and Hujun Bao. Robust monocular slam in dynamic environments. In *ISMAR*, 2013. 2

[40] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. CNN-SLAM: Real-time dense monocular slam with learned depth prediction. In *CVPR*, 2017. 2

[41] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *CVPR*, pages 2022–2030, 2018. 6, 7, 8

[42] Rui Wang, Martin Schworer, and Daniel Cremers. Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras. In *ICCV*, pages 3903–3911, 2017. 2

[43] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 13(4):600–612, 2004. 4

[44] Thomas Whelan, Michael Kaess, Hordur Johannsson, Maurice Fallon, John J Leonard, and John McDonald. Real-time large-scale dense RGB-D slam with volumetric fusion. *IJRR*, 34(4-5):598–626, 2015. 2

[45] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *CVPR*, 2017. 2

[46] D Xu, E Ricci, Ouyang Wanli, Wang Xiaogang, and N Sebe. Monocular depth estimation using multi-scale continuous crfs as sequential deep networks. *TPAMI*, 41(6):1426–1440, 2019. 2

[47] Dan Xu, Wei Wang, Hao Tang, Hong Liu, Nicu Sebe, and Elisa Ricci. Structured attention guided convolutional neural fields for monocular depth estimation. In *CVPR*, 2018. 6

[48] Nan Yang, Rui Wang, Jörg Stückler, and Daniel Cremers. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *ECCV*, 2018. 2

[49] Zhichao Yin and Jianping Shi. GeoNet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018. 1, 3, 5, 6, 7, 8

[50] Alan M Zhang and Lindsay Kleeman. Robust appearance based visual route following for navigation in large-scale outdoor environments. *IJRR*, 28(3):331–356, 2009. 2

[51] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 1, 3, 5, 6, 7, 8