GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose

Zhichao Yin and Jianping Shi
SenseTime Research
{yinzhichao, shijianping}@sensetime.com

Abstract

We propose GeoNet, a jointly unsupervised learning framework for monocular depth, optical flow and ego-motion estimation from videos. The three components are coupled by the nature of 3D scene geometry, jointly learned by our framework in an end-to-end manner. Specifically, geometric relationships are extracted over the predictions of individual modules and then combined as an image reconstruction loss, reasoning about static and dynamic scene parts separately. Furthermore, we propose an adaptive geometric consistency loss to increase robustness towards outliers and non-Lambertian regions, which resolves occlusions and texture ambiguities effectively. Experimentation on the KITTI driving dataset reveals that our scheme achieves state-of-the-art results in all of the three tasks, performing better than previously unsupervised methods and comparably with supervised ones.

1. Introduction

Understanding 3D scene geometry from video is a fundamental topic in visual perception. It includes many classical computer vision tasks, such as depth recovery, flow estimation, visual odometry, *etc*. These technologies have wide industrial applications, including autonomous driving platforms [6], interactive collaborative robotics [11], and localization and navigation systems [12], *etc*.

Traditional *Structure from Motion* (SfM) methods [34, 42] tackle them in an integrated way, which aim to simultaneously reconstruct the scene structure and camera motion. Advances have been achieved recently in robust and discriminative feature descriptors [2, 39], more efficient tracking systems [55], and better exploitation of semantic level information [4], *etc*. Even though, the proneness to outliers and failure in non-textured regions are still not completely eliminated for their inherent reliance on high-quality low-level feature correspondences.

To break through these limitations, deep models [35, 45] have been applied to each of the low-level subproblems and achieve considerable gains against traditional methods. The
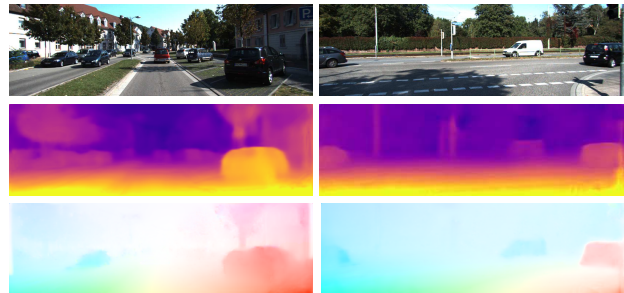

Figure 1. Example predictions by our method on KITTI 2015 [31]. Top to bottom: input image (one of the sequence), depth map and optical flow. Our model is fully unsupervised and can handle dynamic objects and occlusions explicitly.

major advantage comes from big data, which helps capturing high-level semantic correspondences for low level clue learning, thus performing better even in ill-posed regions compared with traditional methods.

Nevertheless, to preserve high performance with more general scenarios, large corpus of groundtruth data are usually needed for deep learning. In most circumstances, expensive laser-based setups and differential GPS are required, restricting the data grow to a large scale. Moreover, previous deep models are mostly tailored to solve one specific task, such as depth [26], optical flow [8], camera pose [22], *etc*. They do not explore the inherent redundancy among these tasks, which can be formulated by geometry regularities via the nature of 3D scene construction.

Recent works have emerged to formulate these problems together with deep learning. But all possess certain inherent limitations. For example, they require large quantities of laser scanned depth data for supervision [48], demand stereo cameras as additional equipment for data acquisition [15], or cannot explicitly handle non-rigidity and occlusions [50, 56].

In this paper, we propose an unsupervised learning framework GeoNet for jointly estimating monocular depth, optical flow and camera motion from video. The foundation of our approach is built upon the *nature of 3D scene geometry* (see Sec. 3.1 for details). An intuitive explanation is that most of the natural scenes are comprised of rigid staic sur-

faces, *i.e.* roads, houses, trees, *etc.* Their projected 2D image motion between video frames can be fully determined by the depth structure and camera motion. Meanwhile, dynamic objects such as pedestrians and cars commonly exist in such scenes and usually possess the characteristics of large displacement and disarrangement.

As a result, we grasp the above intuition using a deep convolutional network. Specifically, our paradigm employs a divide-and-conquer strategy. A novel cascaded architecture consisting of two stages is designed to solve the scene *rigid flow* and *object motion* adaptively. Therefore the global motion field is able to get refined progressively, making our full learning pipeline a decomposed and easier-to-learn manner. The view synthesis loss guided by such fused motion field leads to natural regularization for unsupervised learning. Example predictions are shown in Fig. 1.

As a second contribution, we introduce a novel adaptive geometric consistency loss to overcome factors not included in a pure view synthesis objective, such as occlusion handling and photo inconsistency issues. By mimicking the traditional forward-backward (or left-right) consistency check, our approach filters possible outliers and occlusions out automatically. Prediction coherence is enforced between different views in non-occluded regions, while erroneous predictions get smoothed out especially in occluded regions.

Finally, we perform comprehensive evaluation of our model in all of the three tasks on the KITTI dataset [31]. Our unsupervised approach outperforms previously unsupervised manners and achieves comparable results with supervised ones, which manifests the effectiveness and advantages of our paradigm.

2. Related Work

Traditional Scene Geometry Understanding Structure-from-Motion (SfM) is a long standing problem which infers scene structure and camera motion jointly from potentially very large unordered image collections [13, 16]. Modern approaches commonly start with feature extraction and matching, followed by geometric verification [40]. During the reconstruction process, bundle adjustment [47] is iteratively applied for refining the global reconstructed structure. Lately wide varieties of methods have been proposed in both global and incremental genres [44, 53]. However, these existing methods still heavily rely on accurate feature matching. Without good photo-consistency promise, the performance cannot be guaranteed. Typical failure cases may be caused by low texture, stereo ambiguities, occlusions, *etc.*, which may commonly appear in natural scenes.

Scene flow estimation is another closely related topic to our work, which solves the dense 3D motion field of a scene from stereoscopic image sequences [49]. Top ranked methods on the KITTI benchmark typically involve the joint reasoning of geometry, rigid motion and segmentation [3, 51].

MRFs [27] are widely adopted to model these factors as a discrete labeling problem. However, since there exist large quantities of variables to optimize, these off-the-shelf approaches are usually too slow for practical use. On the other hand, several recent methods have emphasized the rigid regularities in generic scene flow. Taniai *et al.* [46] proposed to segment out moving objects from the rigid scene with a binary mask. Sevilla-Lara *et al.* [41] defined different models of image motion according to semantic segmentation. Wulff *et al.* [54] modified the Plane+Parallax framework with semantic rigid prior learned by a CNN. Different from the above mentioned approaches, we employ deep neural networks for better exploitation of high level cues, not restricted to a specific scenario. Our end-to-end method only takes on the order of milliseconds for geometry inference on a consumer level GPU. Moreover, we robustly estimate high-quality ego-motion which is not included in the classical scene flow conception.

Supervised Deep Models for Geometry Understanding With recent development of deep learning, great progress has been made in many tasks of 3D geometry understanding, including depth, optical flow, pose estimation, *etc.*

By utilization of a two scale network, Eigen *et al.* [9] demonstrated the capability of deep models for single view depth estimation. While such monocular formulation typically has heavy reliance on scene priors, a stereo setting is preferred by many recent methods. Mayer *et al.* [29] introduced a correlation layer to mimic traditional stereo matching techniques. Kendall *et al.* [24] proposed 3D convolutions over cost volumes by deep features to better aggregate stereo information. Similar spirits have also been adopted in learning optical flow. E. Ilg *et al.* [18] trained a stacked network on large corpus of synthetic data and achieved impressive result on par with traditional methods.

Apart from the above problems as dense pixel prediction, camera localization and tracking have also proven to be tractable as a supervised learning task. Kendall *et al.* [23] cast the 6-DoF camera pose relocalization problem as a learning task, and extended it upon the foundations of multi-view geometry [22]. Oliveira *et al.* [36] demonstrated how to assemble visual odometry and topological localization modules and outperformed traditional learning-free methods. Brahmbhatt *et al.* [5] exploited geometric constraints from a diversity of sensory inputs for improving localization accuracy on a broad scale.

Unsupervised Learning of Geometry Understanding For alleviating the reliances on expensive groundtruth data, various unsupervised approaches have been proposed recently to address the 3D understanding tasks. The core supervision typically comes from a view synthesis objective based on geometric inferences. Here we briefly review on

the most closely related ones and indicate the crucial differences between ours.

Garg *et al*. [14] proposed a stereopsis based auto-encoder for single view depth estimation. While their differentiable inverse warping is based on Taylor expansion, making the training objective sub-optimal. Both Ren *et al*. [37] and Yu *et al*. [21] extended the image reconstruction loss together with a spatial smoothness loss for unsupervised optical flow learning, but took no advantage of geometric consistency among predictions. By contrast, Godard *et al*. [15] exploited such constraints in monocular depth estimation by introducing a left-right consistency loss. However, they treat all the pixels equally, which would affect the effectiveness of geometric consistency loss in occluded regions. Concurrent to our work, Meister *et al*. [30] also independently introduce a bidirectional census loss. Different to their stacked structure focusing on unsupervised learning of optical flow, we tackle several geometry understanding tasks jointly. Zhou *et al*. [56] mimicked the traditional structure from motion by learning the monocular depth and ego-motion in a coupled way. Building upon the rigid projective geometry, they do not consider the dynamic objects explicitly and in turn learn a explainability mask for compensation. Similarly, Vijayanarasimhan *et al*. [50] learned several object masks and corresponding rigid motion parameters for modelling moving objects. In contrast, we introduce a residual flow learning module to handle non-rigid cases and emphasize the importance of enforcing geometric consistency in predictions.

3. Method

In this section, we start by the nature of 3D scene geometry. Then we give an overview of our GeoNet. It follows by its two components: rigid structure reconstructor and non-rigid motion localizer respectively. Finally, we raise the geometric consistency enforcement, which is the core of our GeoNet.

3.1. Nature of 3D Scene Geometry

Videos or images are the screenshots of 3D space projected into certain dimensions. The 3D scene is naturally comprised of static background and moving objects. The movement of static parts in a video is solely caused by camera motion and depth structure. Whereas movement of dynamic objects is more complex, contributed by both homogeneous camera motion and specific object motion.

Understanding the homogeneous camera motion is relatively easier compared to complete scene understanding, since most of the region is bounded by its constraints. To decompose the problem of 3D scene understanding by its nature, we would like to learn the scene level consistent movement governed by camera motion, namely the *rigid flow*, and the *object motion* separately.

Here we briefly introduce the notations and basic concepts used in our paper. To model the strictly restricted rigid flow, we define the static scene geometries by a collection of depth maps D_i for frame i, and the relative camera motion $T_{t \to s}$ from target to source frame. The relative 2D rigid flow from target image I_t to source image I_s can be represented by[1]

$$f_{t \to s}^{rig}(p_t) = KT_{t \to s}D_t(p_t)K^{-1}p_t - p_t, \qquad (1)$$

where K denotes the camera intrinsic and p_t denotes homogeneous coordinates of pixels in frame I_t. On the other hand, we model the unconstrained object motion as classical optical flow conception, *i.e*. 2D displacement vectors. We learn the *residual flow* $f_{t \to s}^{res}$ instead of the full representation for non-rigid cases, which we will explain later in Sec. 3.4. For brevity, we mainly illustrate the cases from target to source frames in the following, which one can easily generalize to the reversed cases. Guided by these positional constraints, we can apply *differentiable inverse warping* [20] between nearby frames, which later become the foundation of our fully unsupervised learning scheme.

3.2. Overview of GeoNet

Our proposed GeoNet perceives the 3D scene geometry by its nature in an unsupervised manner. In particular, we use separate components to learn the rigid flow and object motion by rigid structure reconstructor and non-rigid motion localizer respectively. The image appearance similarity is adopted to guide the unsupervised learning, which can be generalized to infinite number of video sequences without any labeling cost.

An overview of our GeoNet has been depicted in Fig. 2. It contains two stages, the rigid structure reasoning stage and the non-rigid motion refinement stage. The first stage to infer scene layout is made up of two sub-networks, *i.e*. the DepthNet and the PoseNet. Depth maps and camera poses are regressed respectively and fused to produce the rigid flow. Furthermore, the second stage is fulfilled by the ResFlowNet to handle dynamic objects. The residual non-rigid flow learned by ResFlowNet is combined with rigid flow, deriving our final flow prediction. Since each of our sub-networks targets at a specific sub-task, the complex scene geometry understanding goal is decomposed to some easier ones. View synthesis at different stage works as fundamental supervision for our unsupervised learning paradigm.

Last but not the least, we conduct geometric consistency check during training, which significantly enhances the coherence of our predictions and achieves impressive performance.

---

[1] Similar to [56], we omit the necessary conversion to homogeneous coordinates here for notation brevity.
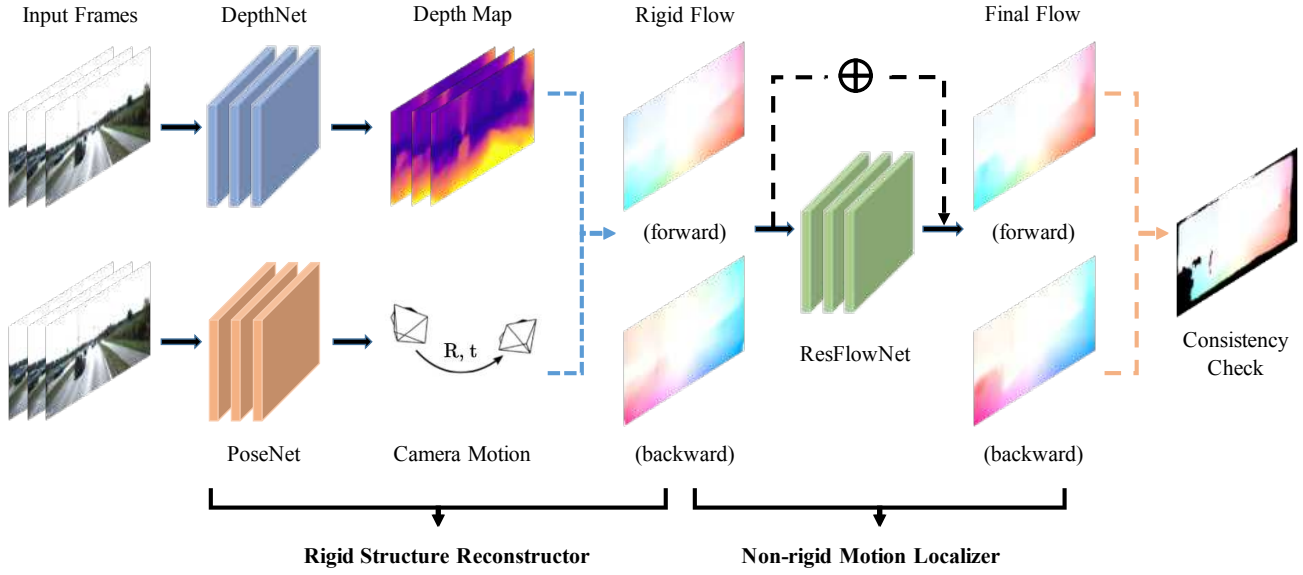
Figure 2. Overview of GeoNet. It consists of rigid structure reconstructor for estimating static scene geometry and non-rigid motion localizer for capturing dynamic objects. Consistency check within any pair of bidirectional flow predictions is adopted for taking care of occlusions and non-Lambertian surfaces.

3.3. Rigid Structure Reconstructor

Our first stage aims to reconstruct the rigid scene structure with robustness towards non-rigidity and outliers. The training examples are temporal continuous frames $I_i(i = 1 \sim n)$ with known camera intrinsics. Typically, a target frame I_t is specified as the reference view, and the other frames are source frames I_s. Our DepthNet takes single view as input and exploits accumulated scene priors for depth prediction. During training, the entire sequence is treated as a mini-batch of independent images and fed into the DepthNet. In contrast, to better utilize the feature correspondences between different views, our PoseNet takes the entire sequence concated along channel dimension as input to regress all the relative 6DoF camera poses $T_{t \to s}$ at once. Building upon these elementary predictions, we are able to derive the global rigid flow according to Eq. (1). Immediately we can synthesize the other view between any pair of target and source frames. Let us denote $\tilde{I}_s^{rig}$ as the inverse warped image from I_s to target image plane by $f_{t \to s}^{rig}$. Thereby the supervision signal for our current stage naturally comes in form of minimizing the dissimilarities between the synthesized view $\tilde{I}_s^{rig}$ and original frame I_t (or inversely).

However, it should be pointed out that rigid flow only dominates the motion of non-occluded rigid region while becomes invalid in non-rigid region. Although such negative effect is slightly mitigated within the rather short sequence, we adopt a robust image similarity measurement [15] for the photometric loss, which maintains the bal-

ance between appropriate assessment of perceptual similarity and modest resilience for outliers, and is differentiable in nature as follows

$$\mathcal{L}_{rw} = \alpha \frac{1 - SSIM(I_t, \tilde{I}_s^{rig})}{2} + (1-\alpha)\|I_t - \tilde{I}_s^{rig}\|_1, \quad (2)$$

where SSIM denotes the structural similarity index [52] and α is taken to be 0.85 by cross validation. Apart from the rigid warping loss $\mathcal{L}_{rw}$, to filter out erroneous predictions and preserve sharp details, we introduce an edge-aware depth smoothness loss $\mathcal{L}_{ds}$ weighted by image gradients

$$\mathcal{L}_{ds} = \sum_{p_t} |\nabla D(p_t)| \cdot (e^{-|\nabla I(p_t)|})^T, \quad (3)$$

where $|\cdot|$ denotes elementwise absolute value, ∇ is the vector differential operator, and T denotes the transpose of image gradient weighting.

3.4. Non-rigid Motion Localizer

The first stage provides us with a stereoscopic perception of rigid scene layout, but ignores the common existence of dynamic objects. Therefore, we raise our second component, *i.e.* the ResFlowNet to localize non-rigid motion.

Intuitively, generic optical flow can directly model the unconstrained motion, which is commonly adopted in off-the-shelf deep models [8, 18]. But they do not fully exploit the well-constrained property of rigid regions, which we have already done in the first stage actually. Instead, we formulate our ResFlowNet for learning the *residual non-rigid*
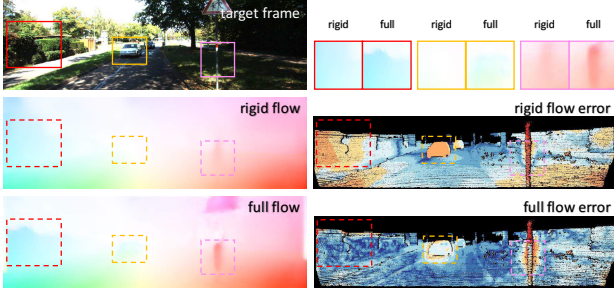
Figure 3. Comparison between flow predictions at different stages. Rigid flow gives satisfactory result in most static regions, while residual flow module focuses on localizing non-rigid motion such as cars, and refining initial prediction in challenging cases such as dark illuminations and thin structures.

flow, the shift solely caused by relative object movement to the world plane. Specifically, we cascade the ResFlowNet after the first stage in a way recommended by [18]. For any given pair of frames, the ResFlowNet takes advantage of output from our rigid structure reconstructor, and predicts the corresponding residual signal $f_{t \to s}^{res}$. The final full flow prediction $f_{t \to s}^{full}$ is constituted by $f_{t \to s}^{rig} + f_{t \to s}^{res}$.

As illustrated in Fig. 3, our first stage, rigid structure reconstructor, produces high-quality reconstruction in most rigid scenes, which sets a good starting point for our second stage. Thereby, our ResFlowNet in motion localizer simply focuses on other non-rigid residues. Note that ResFlowNet can not only rectify wrong predictions in dynamic objects, but also refine imperfect results from first stage thanks to our end-to-end learning protocol, which may arise from high saturations and extreme lighting conditions.

Likewise, we can extend the supervision in Sec. 3.3 to current stage with slight modifications. In detail, following the full flow $f_{t \to s}^{full}$, we perform image warping between any pair of target and source frames again. Replacing the $\tilde{I}_s^{rig}$ with $\tilde{I}_s^{full}$ in Eq. (2), we obtain the full flow warping loss $\mathcal{L}_{fw}$. Similarly, we extend the smoothness loss in Eq. (3) over 2D optical flow field, which we denote as $\mathcal{L}_{fs}$.

3.5. Geometric Consistency Enforcement

Our GeoNet takes rigid structure reconstructor for static scene, and non-rigid motion localizer as compensation for dynamic objects. Both stages utilize the view synthesis objective as supervision, with the implicit assumption of photometric consistency. Though we employ robust image similarity assessment such as Eq. (2), occlusions and non-Lambertian surfaces still cannot be perfectly handled in practice.

To further mitigate these effects, we apply a forward-backward consistency check in our learning framework without changing the network architecture. The work by Godard *et al*. [15] incorporated similar idea into their depth

learning scheme with the left-right consistency loss. However, we argue that such consistency constraints, as well as the warping loss, should not be imposed at occluded regions (see Sec. 4.3). Instead we optimize an adaptive consistency loss across the final motion field.

Concretely, our geometric consistency enforcement is fulfilled by optimizing the following objective

$$\mathcal{L}_{gc} = \sum_{p_t} [\delta(p_t)] \cdot \|\Delta f_{t \to s}^{full}(p_t)\|_1, \qquad (4)$$

where $\Delta f_{t \to s}^{full}(p_t)$ is the full flow difference computed by forward-backward consistency check at pixel p_t in I_t, $[\cdot]$ is the Iverson bracket, and $\delta(p_t)$ denotes the condition of

$$\|\Delta f_{t \to s}^{full}(p_t)\|_2 < max\{\alpha, \beta\|f_{t \to s}^{full}(p_t)\|_2\}, \qquad (5)$$

in which (α, β) are set to be $(3.0, 0.05)$ in our experiment. Pixels where the forward/backward flows contradict seriously are considered as possible outliers. Since these regions violate the photo consistency as well as geometric consistency assumptions, we handle them only with the smoothness loss $\mathcal{L}_{fs}$. Therefore both our full flow warping loss $\mathcal{L}_{fw}$ and geometric consistency loss $\mathcal{L}_{gc}$ are weighted by $[\delta(p_t)]$ pixelwise.

To summarize, our final loss through the entire pipeline becomes

$$\mathcal{L} = \sum_l \sum_{\langle t,s \rangle} \{\mathcal{L}_{rw} + \lambda_{ds}\mathcal{L}_{ds} + \mathcal{L}_{fw} + \lambda_{fs}\mathcal{L}_{fs} + \lambda_{gc}\mathcal{L}_{gc}\},$$
$$(6)$$

where λ denotes respective loss weight, l indexes over pyramid image scales, and $\langle t, s \rangle$ indexes over all the target and source frame pairs and their inverse combinations.

4. Experiments

In this section, we firstly introduce our network architecture and training details. Then we will show qualitative and quantitative results in monocular depth, optical flow and camera pose estimation tasks respectively.

4.1. Implementation Details

Network Architecture Our GeoNet mainly contains three subnetworks, the DepthNet, the PoseNet, together to form the rigid structure reconstructor, and the ResFlowNet, incorporated with the output from previous stage to localize non-rigid motion. Since both the DepthNet and the ResFlowNet reason about pixel-level geometry, we adopt the network architecture in [15] as backbone. Their structure mainly consists of two components: the encoder and the decoder parts. The encoder follows the basic structure of ResNet50 as its more effective residual learning manner. The decoder is made up of deconvolution layers to enlarge the spatial feature maps to full scale as input. To preserve
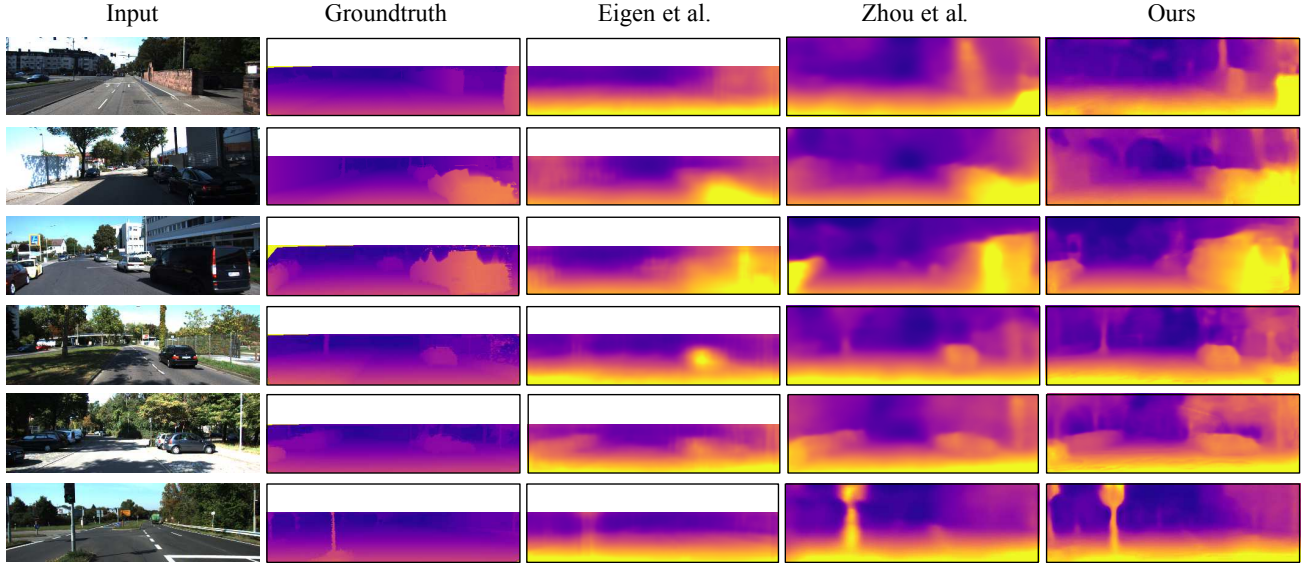
| Input | Groundtruth | Eigen et al. | Zhou et al. | Ours |

Figure 4. Comparison of monocular depth estimation between Eigen *et al*. [9] (supervised by depth), Zhou *et al*. [56] (unsupervised) and ours (unsupervised). The groundtruth is interpolated for visualization purpose. Our method captures details in thin structures and preserves consistently high-quality predictions both in close and distant regions.

both global high-level and local detailed information, we use skip connections between encoder and decoder parts at different corresponding resolutions. Both the depth and residual flow are predicted in a multi-scale scheme. The input to ResFlowNet consists of batches of tensors concated in channel dimension, including the image pair I_s and I_t, the rigid flow $f_{t \to s}^{rig}$, the synthesized view $\tilde{I}_s^{rig}$ and its error map compared with original frame I_t. Our PoseNet regresses the 6-DoF camera poses, *i.e.* the euler angles and translational vectors. The architecture is same as in [56], which contains 8 convolutional layers followed by a global average pooling layer before final prediction. We adopt batch normalization [19] and ReLUs [33] interlaced with all the convolutional layers except the prediction layers.

Training Details Our experiment is conducted using the TensorFlow framework [1]. Though the sub-networks can be trained together in an end-to-end fashion, there is no guarantee that the local gradient optimization could get the network to that optimal point. Therefore, we adopt a stage-wise training strategy, reducing computational cost and memory consumption at meantime. Generally speaking, we first train the DepthNet and the PoseNet, then by fixing their weights, the ResFlowNet is trained thereafter. We also evaluated finetuning the overall network with a smaller batch size and learning rate afterwards, but achieved limited gains. During training, we resize the image sequences to a resolution of 128×416. We also perform random resizing, cropping, and other color augmentations to prevent overfitting. The network is optimized by Adam [25], where $\beta_1 = 0.9$, $\beta_2 = 0.999$. The loss weights are set to be

$\lambda_{ds} = 0.5$, $\lambda_{fs} = 0.2$ and $\lambda_{gc} = 0.2$ for all the experiments. We take an initial learning rate of 0.0002 and mini-batch size of 4 at both stages. The network is trained on a single TitanXP GPU and infers depth, optical flow and camera pose with the speed of $15ms$, $45ms$ and $4ms$ per example at test time. The training process typically takes around 30 epochs for the first stage and 200 epochs for the second stage to converge. To make a fair evaluation, we compare our method with different training/test split for each task on the popular KITTI dataset [31].

4.2. Monocular Depth Estimation

To evaluate the performance of our GeoNet in monocular depth estimation, we take the split of Eigen *et al*. [9] to compare with related works. Visually similar frames to the test scenes as well as static frames are excluded following [56]. The groundtruth is obtained by projecting the Velodyne laser scanned points into image plane. To evaluate at input image resolution, we resize our predictions by interlinear interpolation. The sequence length is set to be 3 during training.

As shown in Table 1, "Ours VGG" trained only on KITTI shares the same network architecture with "Zhou *et al*. [56] without BN", which reveals the effectiveness of our loss functions. While the difference between "Ours VGG" and "Ours ResNet" validates the gains achieved by different network architectures. Our method significantly outperforms both supervised methods [9, 28] and previously unsupervised work [14, 56]. A qualitative comparison has been visualized in Fig. 4. Interestingly, our result is slightly inferior to Godard *et al*. [15] when trained on KITTI and Cityscapes

Method	Supervised	Dataset	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen *et al.* [9] Coarse	Depth	K	0.214	1.605	6.563	0.292	0.673	0.884	0.957
Eigen *et al.* [9] Fine	Depth	K	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu *et al.* [28]	Depth	K	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Godard *et al.* [15]	Pose	K	**0.148**	1.344	5.927	0.247	**0.803**	0.922	0.964
Zhou *et al.* [56]	No	K	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Zhou *et al.* [56] updated[2]	No	K	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Ours VGG	No	K	0.164	1.303	6.090	0.247	0.765	0.919	0.968
Ours ResNet	No	K	0.155	**1.296**	**5.857**	**0.233**	0.793	**0.931**	**0.973**
Garg *et al.* [14] cap 50m	Pose	K	0.169	1.080	5.104	0.273	0.740	0.904	0.962
Ours VGG cap 50m	No	K	0.157	0.990	4.600	0.231	0.781	0.931	0.974
Ours ResNet cap 50m	No	K	**0.147**	**0.936**	**4.348**	**0.218**	**0.810**	**0.941**	**0.977**
Godard *et al.* [15]	Pose	CS + K	**0.124**	**1.076**	**5.311**	**0.219**	**0.847**	**0.942**	**0.973**
Zhou *et al.* [56]	No	CS + K	0.198	1.836	6.565	0.275	0.718	0.901	0.960
Ours ResNet	No	CS + K	0.153	1.328	5.737	0.232	0.802	0.934	0.972

Table 1. Monocular depth results on KITTI 2015 [31] by the split of Eigen *et al.* [9]. For training, K is the KITTI dataset [31] and CS is Cityscapes [7]. Errors for other methods are taken from [15, 56]. We show the best result trained only on KITTI in bold. The results of Garg *et al.* [14] are capped at 50m and we seperately list them for comparison.

datasets both. We believe this is due to the profound distinctions between training data characteristics, *i.e.* rectified stereo image pairs and monocular video sequences. Still, the results manifest the geometry understanding ability of our GeoNet, which successfully captures the regularities among different tasks out of videos.

4.3. Optical Flow Estimation

The performance of optical flow component is validated on the KITTI stereo/flow split. The official 200 training images are adopted as testing set. Thanks to our unsupervised nature, we could take the raw images without groundtruth for training. All the related images in the 28 scenes covered by testing data are excluded. To compare our *residual flow learning* scheme with *direct flow learning*, we specifically trained modified versions of FlowNetS [8] with the unsupervised losses: "Our DirFlowNetS (no GC)" is guided by the warping loss and smoothness loss as in Sec. 3.4, while "Our DirFlowNetS" further incorporates the geometric consistency loss as in Sec. 3.5 during training. Moreover, we conduct ablation study in adaptive consistency loss versus naive consistency loss, *i.e.* without weighting in Eq. (4).

As demonstrated in Table 2, our GeoNet achieves the lowest EPE in overall regions and comparable result in non-occluded regions against other unsupervised baselines. The comparison between "Our DirFlowNetS (no GC)" and "Our DirFlowNetS" already manifests the effectiveness of our geometric consistency loss even in a variant architecture. Futhermore, "Our GeoNet" adopts the same losses but beats "Our DirFlowNetS" in overall regions, demonstrating the advantages of our architecture based on nature of 3D scene geometry (see Fig. 5 for visualized comparison). Neverthe-

Method	Dataset	Noc	All
EpicFlow [38]	-	4.45	9.57
FlowNetS [8]	C+S	8.12	14.19
FlowNet2 [18]	C+T	4.93	10.06
DSTFlow [37]	K	6.96	16.79
Our DirFlowNetS (no GC)	K	6.80	12.86
Our DirFlowNetS	K	**6.77**	12.21
Our Naive GeoNet	K	8.57	17.18
Our GeoNet	K	8.05	**10.81**

Table 2. Average end-point error (EPE) on KITTI 2015 flow training set over non-occluded regions (Noc) and overall regions (All). The handcrafted EpicFlow takes 16s per frame at runtime; The supervised FlowNetS is trained on FlyingChairs and Sintel; Likewise the FlowNet2 is trained on FlyingChairs and FlyingThings3D.

less, naively enforcing consistency loss proves to deteriorate accuracy as shown in "Our Naive GeoNet" entry.

Gradient Locality of Warping Loss However, the direct unsupervised flow network DirFlowNetS performs better in non-occluded regions than GeoNet, which seems unreasonable. We investigate into the end-point error (EPE) distribution over different magnitudes of groundtruth residual flow *i.e.* $\|f^{gt} - f^{rig}\|$, where f^{gt} denotes the groundtruth full flow. As shown in Fig. 6, our GeoNet achieves much lower error in small displacement relative to f^{rig}, while the error increases with large displacement. Experimentally, we find that GeoNet is extremely good at rectifying small errors from rigid flow. However, the predicted residual flow tends to prematurely converge to a certain range, which is in consistency with the observations of [15]. It is because the gradients of warping based loss are derived by local pixel intensity differences, which would be amplified in a more complicated cascaded architecture, *i.e.* the GeoNet. We have experimented by replacing the warping loss with a nu-

---

[2] Results are updated from https://github.com/tinghuiz/SfMLearner with improved implementation.

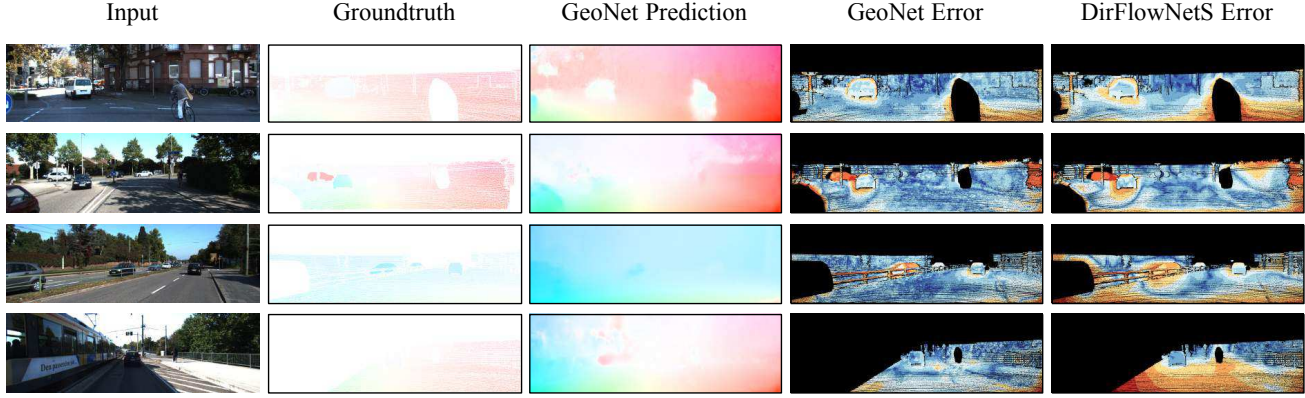| Input | Groundtruth | GeoNet Prediction | GeoNet Error | DirFlowNetS Error |

Figure 5. Comparison of direct flow learning method DirFlowNetS (geometric consistency loss enforced) and our GeoNet framework. As shown in the figure, GeoNet shows clear advantages in occluded, texture ambiguous regions, and even in shaded dim area.
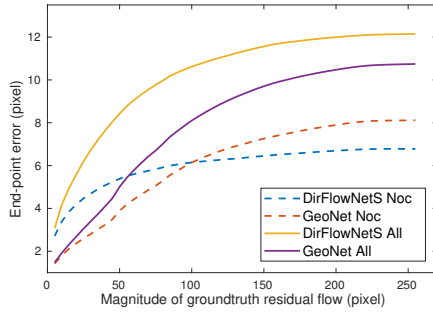

Figure 6. Average EPE at different magnitude of groundtruth residual flow. In total regions (All), GeoNet consistently outperforms direct flow regression; but in non-occluded regions (Noc), the advantage of GeoNet is restricted to the neighbourhood of rigid flow.

merically supervised one (guided by groundtruth or knowledge distilled from the DirFlowNetS [17]) without changing network architecture, and found such issue disappeared. Investigating practical solution to avoid the gradient locality of warping loss is left as our future work.

4.4. Camera Pose Estimation

We have evaluated the performance of our GeoNet on the official KITTI visual odometry split. To compare with Zhou *et al.* [56], we divide the 11 sequences with groundtruth into two parts: the 00-08 sequences are used for training and the 09-10 sequences for testing. The sequence length is set to be 5 during training. Moreover, we compare our method with a traditional representative SLAM framework: ORB-SLAM [32]. It involves global optimization steps such as loop closure detection and bundle adjustment. Here we present two versions: "The ORB-SLAM (short)" only takes 5 frames as input and "ORB-SLAM (long)" takes the entire sequence as input. All of the results are evaluated in terms of 5-frame trajectories, and scaling factor is optimized to align with groundtruth

to resolve scale ambiguity [43]. As shown in Table 3, our method outperforms all of the competing baselines. Note that even though our GeoNet only utlizes limited information within a rather short sequence, it still achieves better result than "ORB-SLAM(full)". This reveals again that our geometry anchored GeoNet captures additional high level cues other than sole low level feature correspondences. Finally, we analyse the failure cases and find the network sometimes gets confused about the reference system when large dynamic objects appear nearby in front of the camera, which commonly exist in direct visual SLAM [10].

Method	Seq.09	Seq.10
ORB-SLAM (full)	0.014 ± 0.008	0.012 ± 0.011
ORB-SLAM (short)	0.064 ± 0.141	0.064 ± 0.130
Zhou *et al.* [56]	0.021 ± 0.017	0.020 ± 0.015
Zhou *et al.* [56] updated	0.016 ± 0.009	0.013 ± 0.009
Our GeoNet	$\mathbf{0.012 \pm 0.007}$	$\mathbf{0.012 \pm 0.009}$

Table 3. Absolute Trajectory Error (ATE) on KITTI odometry dataset. The results of other baselines are taken from [56]. Our method outperforms all of the other methods.

5. Conclusion

We propose the jointly unsupervised learning framework GeoNet, and demonstrate the advantages of exploiting geometric relationships over different previously "isolated" tasks. Our unsupervised nature profoundly reveals the capability of neural networks in capturing both high level cues and feature correspondences for geometry reasoning. The impressive results compared to other baselines including the supervised ones indicate possibility of learning these low level vision tasks without costly collected groundtruth data.

For future work, we would like to tackle the gradient locality issue of warping based loss, and validate the possible improvement of introducing semantic information into our GeoNet.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, 2016. 6

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *CVIU*, 2008. 1

[3] A. Behl, O. H. Jafari, S. K. Mustikovela, H. A. Alhaija, C. Rother, and A. Geiger. Bounding boxes, segmentations and object coordinates: How important is recognition for 3d scene flow estimation in autonomous driving scenarios? In *ICCV*, 2017. 2

[4] M. Blaha, C. Vogel, A. Richard, J. Wegner, K. Schindler, and T. Pock. Large-scale semantic 3d reconstruction: an adaptive multi-resolution model for multi-class volumetric labeling. In *CVPR*, 2016. 1

[5] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz. Mapnet: Geometry-aware learning of maps for camera localization. *CoRR*, 2017. 2

[6] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015. 1

[7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 7

[8] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015. 1, 4, 7

[9] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 2, 6, 7

[10] J. Engel, T. Schöps, and D. Cremers. LSD-SLAM: Large-scale direct monocular slam. In *ECCV*, 2014. 8

[11] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 2003. 1

[12] F. Fraundorfer, C. Engels, and D. Nistér. Topological mapping, localization and navigation using image collections. In *IROS*, 2007. 1

[13] Y. Furukawa and C. Hernndez. Multi-view stereo: A tutorial. *Found. Trends. Comp. Graphics and Vision*, 2015. 2

[14] R. Garg, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016. 3, 6, 7

[15] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 1, 3, 4, 5, 6, 7

[16] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 2

[17] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *CoRR*, 2015. 8

[18] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017. 2, 4, 5, 7

[19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 6

[20] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015. 3

[21] J. Y. Jason, A. W. Harley, and K. G. Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *ECCV Workshops*, 2016. 3

[22] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. *CVPR*, 2017. 1, 2

[23] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, 2015. 2

[24] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-end learning of geometry and context for deep stereo regression. In *CVPR*, 2017. 2

[25] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, 2014. 6

[26] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 1

[27] S. Z. Li. Markov random field models in computer vision. In *ECCV*, 1994. 2

[28] F. Liu, C. Shen, G. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *PAMI*, 2016. 6, 7

[29] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 2

[30] S. Meister, J. Hur, and S. Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, 2018. 3

[31] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *CVPR*, 2015. 1, 2, 6, 7

[32] R. Mur-Artal, J. D. Tards, J. M. M. Montiel, and D. Glvez-Lpez. ORB-SLAM: a versatile and accurate monocular SLAM system. *Transactions on Robotics*, 2015. 8

[33] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010. 6

[34] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. DTAM: Dense tracking and mapping in real-time. In *ICCV*, 2011. 1

[35] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 1

[36] G. L. Oliveira, N. Radwan, W. Burgard, and T. Brox. Topometric localization with deep learning. In *ISRR*, 2017. 2

[37] Z. Ren, J. Yan, B. Ni, Y. Yuan, X. Yang, and H. Zha. Unsupervised deep learning for optical flow estimation. In *AAAI*, 2017. 3, 7

[38] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015. 7

[39] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *ICCV*, 2011. 1

[40] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2

[41] L. Sevilla-Lara, D. Sun, V. Jampani, and M. J. Black. Optical flow with semantic segmentation and localized layers. In *CVPR*, 2016. 2

[42] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *IJCV*, 2008. 1

[43] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IROS*, 2012. 8

[44] C. Sweeney, T. Sattler, T. Hollerer, M. Turk, and M. Pollefeys. Optimizing the viewing graph for structure-from-motion. In *ICCV*, 2015. 2

[45] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1

[46] T. Taniai, S. N. Sinha, and Y. Sato. Fast multi-frame stereo scene flow with motion segmentation. In *CVPR*, 2017. 2

[47] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustmenta modern synthesis. In *International Workshop on Vision Algorithms*, 1999. 2

[48] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *CVPR*, 2017. 1

[49] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. In *ICCV*, 1999. 2

[50] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki. Sfm-net: Learning of structure and motion from video. *CoRR*, 2017. 1, 3

[51] C. Vogel, K. Schindler, and S. Roth. 3d scene flow estimation with a piecewise rigid scene model. *IJCV*, 2015. 2

[52] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004. 4

[53] C. Wu. Towards linear-time incremental structure from motion. In *3DTV-CON*, 2013. 2

[54] J. Wulff, L. Sevilla-Lara, and M. J. Black. Optical flow in mostly rigid scenes. In *CVPR*, 2017. 2

[55] G. Zhang, H. Liu, Z. Dong, J. Jia, T.-T. Wong, and H. Bao. Efficient non-consecutive feature tracking for robust structure-from-motion. *TIP*, 2016. 1

[56] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 1, 3, 6, 7, 8