Neural Computing and Applications (2021) 33:3209–3220
https://doi.org/10.1007/s00521-020-05192-z

ORIGINAL ARTICLE

Geometry understanding from autonomous driving scenarios based on feature refinement

Mingliang Zhai[1] · Xuezhi Xiang[1]

Received: 2 March 2020 / Accepted: 11 July 2020 / Published online: 22 July 2020

Abstract
Nowadays, many deep learning applications benefit from multi-task learning with several related objectives. In autonomous driving scenarios, being able to infer motion and spatial information accurately is essential for scene understanding. In this paper, we propose a unified framework for unsupervised joint learning of optical flow, depth and camera pose. Specifically, we use a feature refinement module to adaptively discriminate and recalibrate feature, which can integrate local features with their global dependencies to capture rich contextual relationships. Given a monocular video, our network firstly calculates rigid optical flow by estimating depth and camera pose. Then, we design an auxiliary flow network for inferring non-rigid flow field. In addition, a forward–backward consistency check is adopted for occlusion reasoning. Extensive analyses on KITTI dataset are conducted to verify the effectiveness of our proposed approach. The experimental results show that our proposed network can produce sharper, clearer and detailed depth and flow maps. In addition, our network achieves potential performance compared to the recent state-of-the-art approaches.

Keywords Geometry understanding · Multi-task learning · Depth · Optical flow

1 Introduction

Scene geometry understanding is one of the critical problems for several computer vision applications, such as autonomous driving and augmented reality. Being able to estimate motion and depth from a monocular video is essential for scene inference, especially in autonomous driving perception. Traditional approaches for geometry understanding, such as structure from motion (SfM) [18], dense tracking and mapping (DTAM) [16] and ORB-SLAM [15], always rely on feature matching, which requires accurate image correspondence. However, these methods usually fail to match features in the regions of thin structure, non-textured and occlusion.

To address this issue, some recent approaches learn geometry understanding using deep learning technique. Recent state-of-the-art deep networks for depth and flow estimation usually depend on the availability of a large amount of labeled data, such as [3, 4, 9, 13, 20]. Although these supervised approaches achieve promising results, these networks need to be trained on a large amount of labeled data that are expensive and difficult to be acquired in real world. Therefore, many works [6, 7, 17] attempt to learn optical flow or depth in an unsupervised manner. They usually use prior knowledge constraint to define loss function and only use unlabeled data for training. Garg et al. [6] propose an encoder–decoder architecture (U-Net) similar to FlowNet [3], which is used to predict single-image depth map. The training process is guided by the photometric assumption and smoothness constraint. Based on [6], Godard et al. [7] also design an unsupervised U-Net for depth prediction, which uses left–right consistency loss additionally. However, [6, 7] are limited to use calibrated stereo image pairs for training. Moreover, although these unsupervised methods sidestep the requirement of the

✉ Xuezhi Xiang
xiangxuezhi@hrbeu.edu.cn

Mingliang Zhai
zmlshiwo@outlook.com

[1] Key Laboratory of Advanced Marine Communication and Information Technology, Ministry of Industry and Information Technology, School of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

labeled data, these works only focus on the single task and cannot estimate optical flow and depth in a unified network. In addition, real application scenarios often need multi-task collaborative work to extract more 3D scene geometry information, especially for autonomous driving.

To solve the above problem, several works [14, 27–29, 35] propose to design a multi-task network for 3D scene geometry, which can infer several tasks simultaneously, such as depth, camera pose and optical flow. Zhou et al. [35] propose a novel network for unsupervised learning of depth and ego-motion by watching a monocular video. In contrast to [6, 7], the work [35] only needs to use monocular video instead of stereo image pairs to train the network, which is conventional for real-world application. In addition, the work [35] can predict depth map and camera ego-motion from the monocular video simultaneously. Following [35], Mahjourian et al. [14] introduce a novel 3D-based loss function, which can enforce consistency of the estimated 3D point cloud and ego-motion. Yang et al. [28] present a depth-normal representation and conduct a dense edge-aware depth-normal consistency during the training process. Furthermore, Yang et al. [27] propose a novel network for learning edge with geometry, which defines an edge estimation branch to predict edge map. Then, they use the edge map to constrain the depth and camera ego-motion branches. Yin and Shi [29] introduce a residual flow network into [35] for dense motion estimation. Although [14, 27–29, 35] introduce various constraints and networks to extract more accurate depth, flow and camera motion from the monocular video, these approaches ignore the effectiveness of features representation on geometry understanding and lack the ability of discriminative learning across features channel and spatial dimensions. In addition, these methods fail to adaptively integrate local features with their global dependencies, which hinders the network from capturing rich contextual information. In particular, for real scenes, such as automatic driving, the contextual information of the scene is rich, including many high-frequency and low-frequency signals, which requires the network to discriminate the importance of features. Therefore, it is necessary to enhance the discriminative ability of feature representations for the task of geometry understanding.

One limited way to address this issue is to use skipping connection to fuse multi-level features so that the network can capture multi-level scene contexts. Garg et al. [6, 9, 14, 28] adopt the encoder–decoder architecture for depth estimation, which contains skipping connection between encoder and decoder. The feature of the encoder can be reused in the decoder via skipping connection, which enhances the diversity of features in the decoder. Nevertheless, this operation cannot enhance more useful features and suppress unimportant features adaptively, hindering the representational power of the network. Recently, self-attention is an effective solution to address the above issues and is widely used in many vision tasks [2, 21, 26, 34], which can adaptively capture meaningful feature information to guide feature learning. However, few works have been proposed to investigate the effect of the self-attention for the task of scene geometry understanding.

Motivated by the above observation, in this paper, we propose to boost the feature refinement for 3D scene geometry understanding by employing self-attention module and design a unified network for depth, flow and camera pose estimation, which can model global and local feature dependencies and emphasize meaningful features along channel and spatial dimensions adaptively. Our network can be trained end to end in an unsupervised manner and can predict optical flow, depth and camera pose from the monocular video. Since the rigid motion can be calculated by depth map and camera pose, our network is suitable for automatic driving scenarios. In summary, the main contributions of this work are as follows:

1. A novel, end-to-end trainable architecture with feature refinement module is proposed, which can jointly learn depth, optical flow and camera ego-motion from a monocular video. To the best of our knowledge, we are the first to introduce the feature refinement module into scene geometry understanding, which not only recalibrates channel-wise feature but also models rich contextual relationships over local feature representations.
2. The feature refinement is conducted on both optical flow and depth tasks for boosting the quality of flow and depth maps.
3. The qualitative and quantitative results show that the proposed framework performs substantially better than most of the existing approaches and achieves comparable results on KITTI dataset compared to the recent state-of-the-art approaches.

The rest of the paper is organized as follows: Section 2 reviews the related works of scene geometry understanding. Section 3 mainly describes the details of feature refinement module, the entire architecture of our proposed network and multi-task learning loss function. Section 4 introduces the training details and the datasets used in our experiments and reports the experimental results on public benchmarks. Finally, we give the conclusion and future work in Sect. 5.

2 Related work

In this section, we first introduce the traditional approaches for scene geometry understanding briefly. Then, we review supervised geometry understanding based on CNNs. Furthermore, we mainly describe unsupervised learning of geometry understanding. Finally, we review the attention-based image analysis methods.

2.1 Traditional approaches for geometry understanding

Structure from motion (SfM) is a long-standing problem, which predicts scene structure and camera motion jointly from adjacent images or video. Traditional approaches always rely on feature matching [15, 16, 18], which require accurate image correspondence. However, these methods fail at the regions of low texture, stereo ambiguities and occlusion and cannot handle single view reconstruction. Recently, many deep learning-based methods are proposed to address the above issues.

2.2 Supervised geometry understanding using CNNs

Recent advances in CNNs have achieved significant success in scene geometry understanding. Many methods attempt to learn depth in a supervised manner. Eigen et al. [4] propose a single-image depth estimation network, which incorporates coarse-scale depth prediction with fine-scale prediction. To explore the relationships among image features, Liu et al. [13] propose a novel network, which combines CNNs and conditional random field (CRF). Laina et al. [11] introduce residual network into depth estimation and mainly focus on indoor scene. Fu et al. [5] develop a deep ordinal regression network, which conducts the ordinal competition among depth values. Although the supervised deep learning greatly improves the performance of depth estimation, these methods require a large amount of labeled data for training, which is particularly difficult to be collected in real world.

2.3 Unsupervised geometry understanding using CNNs

To address the above problems, recently, many works tend to learn geometry in an unsupervised manner. Garg et al. [6] propose a classic unsupervised network for depth estimation, which uses image reconstruction loss and depth smoothness loss to guide the training process. The work [6] shows that the unsupervised learning of depth can be seen as an image reconstruction problem. Based on [6], Godard et al. [7] propose a depth estimation network with left–right consistency loss. However, these two networks only can be used to estimate the depth map and rely on stereo image pairs for training. Zhou et al. [35] propose a novel network for geometry understanding from a monocular video, which can learn depth and camera pose in a joint manner. Based on [35], Yang et al. [28] introduce an edge-aware depth-normal consistency, which improves the geometry consistency between different projections of the space. Further, Yang et al. [27] introduce an edge network into geometry understanding, which is used to constrain the estimated depth map and camera ego-motion. Mahjourian et al. [14] introduce a novel iterative closest point (ICP) loss into geometry understanding, which can enforce consistency of the estimated 3D point clouds and ego-motion. Zhan et al. [31] introduce deep feature reconstruction into depth estimation. Yin and Shi [29] propose a residual flow network for handling non-rigid region. Although geometry understanding based on CNNs has achieved great progress, existing methods ignore to explore the effectiveness of self-attention for geometry understanding.

2.4 Attention-based image analysis methods

The core goal of attention mechanism is to exploit the global information of features, which can adaptively enhance important features and suppress useless features. The attention mechanism is widely used in many vision tasks, such as image classification [8, 21, 26], image super-resolution [12, 34] and object detection [2, 19, 37]. Wang et al. [21] propose a residual attention network for image classification. However, the computational burden is huge due to the residual architecture. Hu et al. [8] propose squeeze-and-excitation network for image classification, which can model independences among feature channels. Zhang et al. [34] use the channel attention module proposed in [8] for image super-resolution task. Nevertheless, the channel attention module [8] only considers the relationship between channels and cannot capture the spatial dependencies. Woo et al. [26] propose a convolutional block attention module for image classification, which not only can adaptively refine channel-wise features by considering interdependencies among channels but also can utilize global contextual information to emphasize or suppress features in different spatial locations. However, few works introduce spatial–channel combinational attention into geometry understanding. In this paper, we explore the effectiveness of self-attention for scene geometry understanding.

2.5 Video analysis

Video analysis is an important task in the field of computer vision. Wang and Wang [22] propose a cross-agent for action recognition, which uses the transfer learning technique to model the relationship between the source agent and the target agent. Yu et al. [30] propose a weakly semantic guided method for both environment-constrained and cross-domain action recognition. Zhang et al. [32] propose a causal recurrent flow-based method for online video object detection, which uses temporal context information to enhance the ability of feature representation. Wang et al. [23] propose a frame-subsampled and drift-resilient method for video object tracking, which only needs to handle the subsampled video frames. Zhang et al. [33] propose a sitcom-star-based clothing retrieval method for video content-based advertising. They design a deep learning framework for human-body detection, human pose selection, face verification, clothing detection and retrieval from advertisements. In this paper, we mainly focus on the task of scene geometry understanding from monocular video.

3 Approach

The goal of this work is to estimate depth, flow and camera pose from an unlabeled monocular video sequence. Given a monocular video sequence, our network can predict the depth map D, flow map W and camera ego-motion T directly in a joint way. The entire of our framework contains three sub-networks: DepthNet, PoseNet and FlowNet. Despite being jointly trained, the depth, flow and camera pose estimation networks can be used independently during inference. Figure 1 shows the entire architecture of our proposed network. In the following, we first introduce the structure of feature refinement module and then describe the neural network architecture as well as its multi-task learning loss in detail.

3.1 Feature refinement module

We assume that given original feature map $F = \{F_1, F_2, \ldots, F_c\}$, where c is the number of channels, the feature refinement module (FR) can produce the channel attention map A_c and spatial attention map A_s directly, which are used to refine the feature map F. The width and height of F are W and H. Figure 2 shows the details of FR. The FR contains two parts: channel and spatial attention modules. We can see that channel attention contains four parts, max-pooling, avg-pooling, multi-layer perceptron (MLP) and sigmoid function. The max-pooling and avg-pooling operations can aggregate the global information of input features. The global max-pooling process can be defined as

$$F_c^{max} = Max(F_c(i,j)), \tag{1}$$

where $Max(\cdot)$ is the maximization function and $F_c(i,j)$ is the value at position (i, j) of cth feature F_c. The global average pooling process can be defined as

$$F_c^{avg} = \frac{1}{H * W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_c(i,j), \tag{2}$$

where $F_c(i,j)$ is the value at position (i, j) of channel feature F_c. H and W are the width and height of input features. Then, these two features F^{max} and F^{avg} are fed into multilayer perception (MLP) with one hidden layer to reduce the channel number with radio of r. In our experiments, r is set to 16.

Then, O_1 and O_2 are merged by using element-wise summation. We use a sigmoid function to normalize

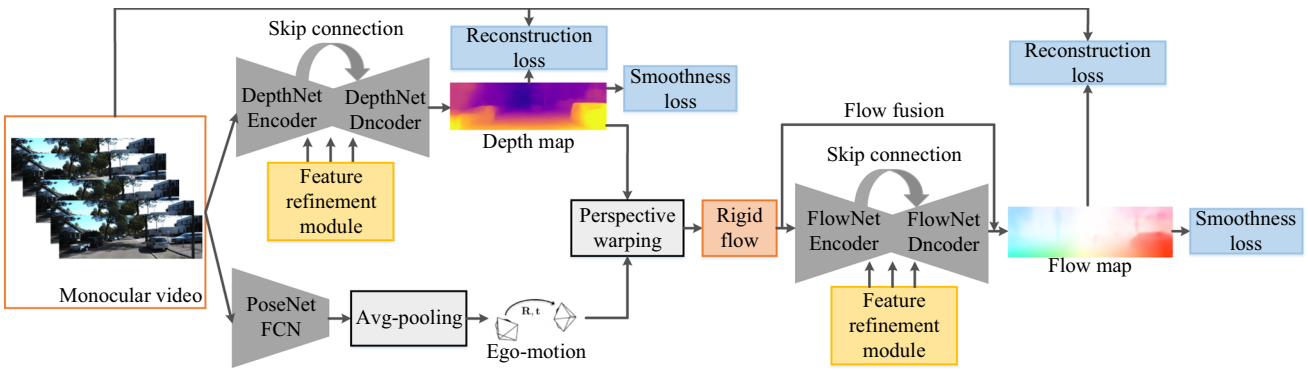

Fig. 1 Overview of our proposed unsupervised learning framework for monocular depth, flow and camera pose estimation. Our network contains three parts: DepthNet, PoseNet and FlowNet. This network takes target and source view images I_t and I_s as inputs, and outputs per-pixel depth map D_t, optical flow map $W_{t\to s}$ and camera ego-motion $T_{t\to s}$. Both DepthNet and FlowNet adopt encoder–decoder architecture. PoseNet adopts fully convolution network (FCN). The feature refinement module is conducted on both DepthNet and FlowNet for improving the quality of depth and flow maps

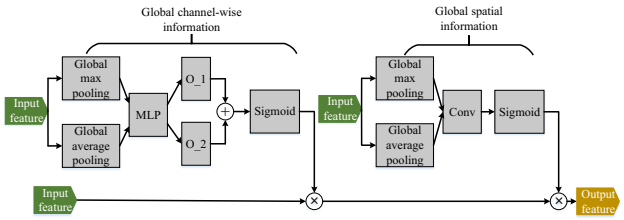

Fig. 2 The architecture of feature refinement module (FR)

features after the summation operation. The final channel attention map A_c can be defined as

$$A_c = Sig(Ac1 + Ac2), \tag{3}$$

where $Sig(\cdot)$ denotes the sigmoid function. The feature refined by channel attention can be defined as

$$F^1_{ref} = F \times A_c. \tag{4}$$

The spatial attention mainly contains four parts, including max-pooling, avg-pooling, convolution layer and sigmoid function. Similar to channel attention, the max-pooling and average-pooling operations can integrate local features with their global dependencies. Thus, we obtain F^{max} and F^{avg} using Eqs. (1) and (2).

Then, we concatenate F^{max}_c and F^{avg}_c. The concatenated feature F^{con} is convolved by a standard convolution layer with a kernel of $7 * 7$ followed a sigmoid function for normalization, which produces a 2D spatial attention map A_s. The final refined feature can be defined as

$$F^2_{ref} = F^1_{ref} \times A_s. \tag{5}$$

With the feature refinement module, the original feature can be recalibrated adaptively.

3.2 Network architecture

The entire network architecture is shown in Fig. 1. Our network can be divided into three parts: DepthNet, PoseNet and FlowNet. Specially, we embed feature refinement module (FR) into both flow and depth sub-networks, which can recalibrate features alone two aspects and can produce more useful and important features for improving the quality of depth and optical flow maps.

Figure 3 shows the network architecture of depth and camera pose estimation. For monocular depth estimation, we adopt residual network [7] as the backbone, which contains the contracting part and expanding part. The contracting part is composed of one standard convolutional layer, max-pooling layer and several residual blocks each of which consists of $3 * 3$ convolution layers. The contracting part is similar to most encoder–decoder architectures but incorporates the feature refinement module (FR). The numbers of feature channels for layers from "Conv1d" to "Refined4" are 64, 64, 256, 512, 1024 and 2048, respectively. The kernel of the first convolution "Conv1d" is set to $7 * 7$. The encoder part is composed of six stages, and each stage downsamples the image or feature to half input resolution. The layer numbers of residual blocks at each stage are set to 3, 4, 6, 3 after the second stage. Figure 4 shows the expanding part of depth network, which contains six deconvolution layers followed by FR. The numbers of feature channels for layers from "Deconv1" to "Refined10" are 512, 256, 128, 63, 32 and 16, respectively. The decoder part upsamples the output of the encoder using successive deconvolution layers. Moreover, we use multiple skip connections to input the output layers at each scale of the encoder at the respective decoder scales.

The PoseNet receives target and source view pair I_s and I_t as input and predicts the relative camera pose between two input views. We conduct eight convolution layers followed by an avg-pooling layer before final prediction to regress 6-DoF camera pose. The details of the connection are shown in Fig. 3. The numbers of feature channels for layers from "Conv1p" to "Conv7p" are 16, 32, 64, 128, 256, 256, 256 and 6. The PoseNet is similar to [35], but we do not share the weights of layers "Conv1p" to "Conv6p" with depth network. Each convolution layer is followed by batch normalization and ReLU function except the last layer.

The architecture of FlowNet is similar to the DepthNet, which is designed to estimate non-rigid flow fields. We also use FR module to refine the feature of optical flow. The final optical flow is the addition of the rigid flow and non-rigid flow.

3.3 Multi-learning loss function

The supervision signals for training our depth, camera pose and optical flow prediction CNNs mainly come from image reconstruction, depth regularization and flow regularization. Here, we first introduce the rigid reconstruction from the estimated depth and camera pose. And then, we introduce the flow reconstruction from the estimated flow field.

3.3.1 Rigid reconstruction

The DepthNet and PoseNet are constrained by a perspective warping formula. Let I_t denote a target view and I_s denote a source view. Given monocular images, the DepthNet and PoseNet can predict depth map D_t and camera motion $T_{t\to s}$ between the target and source views. These two branches are constrained by a perspective warping formulation. Let p_t denote the homogeneous coordinates of a pixel in the target view I_t. The projected coordinates p^w_s on the source view I_s can be defined as

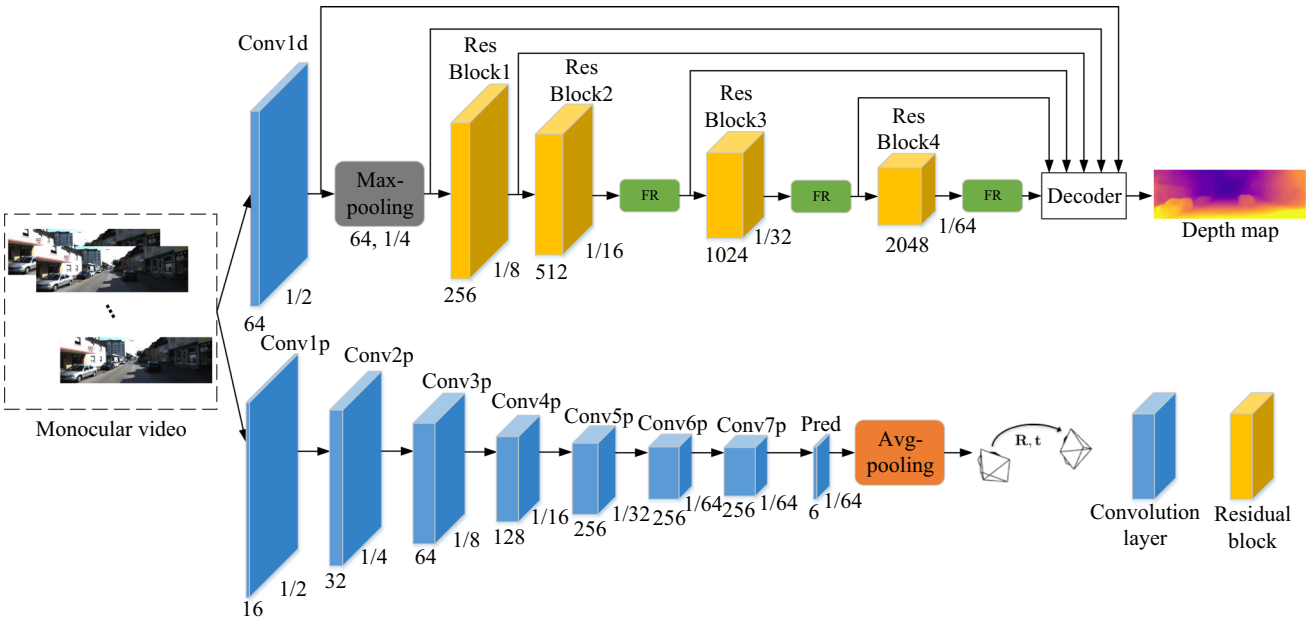

Fig. 3 The architecture of DepthNet and PoseNet. The DepthNet is composed of a series of convolution and deconvolution layers. From ResBlock2, we use FR to refine the feature. The PoseNet is composed of successive convolution layer

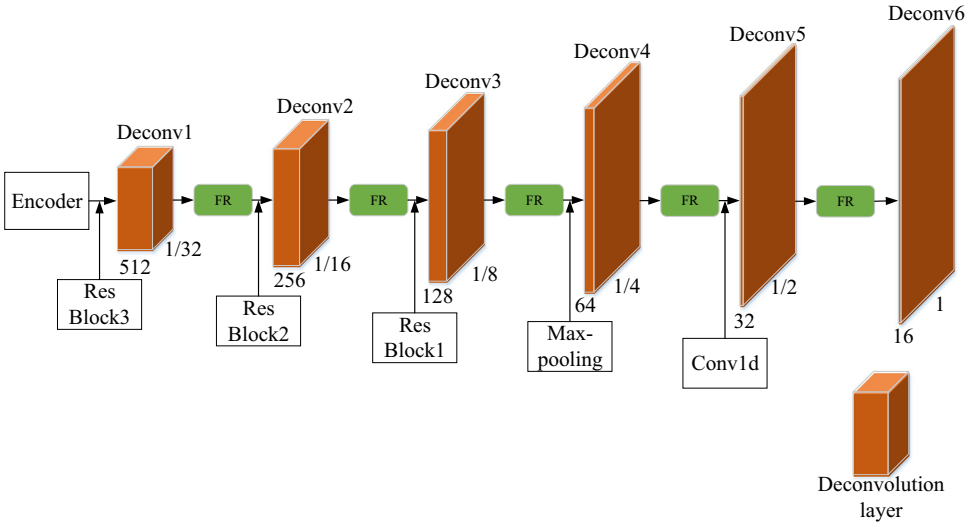

Fig. 4 The architecture of decoder part. Deconvolutional layers in decoder are introduced for the purpose of recovering spatial and detail information

$$p_s^w = KT_{t\to s}D_t(p_t)K^{-1}p_t, \tag{6}$$

where K denotes the camera intrinsic matrix. Then, the relative 2D rigid flow from target image I_t to source image I_s can be defined as

$$f_{t\to s}^{rig}(p_t) = p_s^w - p_t. \tag{7}$$

Further, let I_s^w denote the synthesized view using the rigid flow $f_{t\to s}^{rig}$. The rigid reconstruction loss can be defined as

$$L_{br} = \alpha\frac{1 - SSIM(I_t, I_s^w)}{2} + (1-\alpha)||I_t - I_s^w||_1, \tag{8}$$

where $SSIM$ denotes the image similarity measurement [25] and $||\cdot||_1$ denotes the L1 norm. Weight α is introduced to maintain the balance between perceptual similarity and robustness to outliers. The above rigid reconstruction loss is typically not sufficient to constrain the depth in textureless regions. Therefore, we introduce the smoothness loss of depth map, which can be defined as

$$L_{sd} = \sum_{x_t} |\bigtriangledown D(x_t)|e^{-|I(x_t)|}, \tag{9}$$

where $D(x_t)$ is the estimated depth and $\bigtriangledown$ is the vector differential operator. $|\cdot|$ denotes absolute operation.

3.4 Flow reconstruction

The residual flow sub-network is used to calculate the motion of non-rigid region. Let $f_{t\to s}^{non}$ denote the non-rigid flow. The full flow can be calculated as follows:

$$f_{t\to s}^{full} = f_{t\to s}^{non} + f_{t\to s}^{rig}. \tag{10}$$

Further, let I_s^f denote the synthesized view using the estimated full flow $f_{t\to s}^{full}$. The flow reconstruction loss can be defined as

$$L_{bf} = \alpha \frac{1 - SSIM(I_t, I_s^f)}{2} + (1-\alpha)||I_t - I_s^f||_1. \tag{11}$$

The smoothness loss of full flow can be defined as

$$L_{sf} = \sum_{x_t} |\bigtriangledown W(x_t)| e^{-|I(x_t)|}, \tag{12}$$

where $W(x_t)$ is the estimated full flow.

3.5 Occlusion reasoning

A forward–backward consistency check is widely used in many works [24, 36] to identify invalid regions. Here, we use the forward–backward consistency check to extract the occlusion region. This process can be defined as

$$|W^f + W^b| < \alpha_1(|W^f|^2 + |W^b|^2) + \alpha_2, \tag{13}$$

where W^f and W^b denote the forward and backward flow fields. α_1 and α_2 are threshold. When the network calculates the loss, the occlusion point is eliminated. Furthermore, to filter possible outliers and occlusions out automatically, we conduct geometric consistency check during training, which can be defined as

$$L_{fg} = \sum_{p_t} [\delta_{p_t}] || \bigtriangleup f_{t\to s}^{full}(p_t)||_1, \tag{14}$$

where $f_{t\to s}^{full}(p_t)$ is the full flow difference at pixel p_t computed by forward–backward consistency check. δ_{p_t} denotes the condition of

$$||f_{t\to s}^{full}(p_t)||_2 < max\{a, b||f_{t\to s}^{full}(p_t)||_2\}, \tag{15}$$

where a and b are set to 3 and 0.05. $||\cdot||_2$ denotes L2 norm.

3.6 Total loss function

Based on the above discussions, the total loss for training our multi-task neural network can be defined as

$$L_{total} = \lambda_{br}L_{br} + \lambda_{fr}L_{fr} + \lambda_{ds}L_{ds} + \lambda_{fs}L_{fs} + \lambda_{fg}L_{fg}, \tag{16}$$

where λ_{br}, λ_{fr}, λ_{ds}, λ_{fs} and λ_{fg} represent respective loss weights.

4 Experiments

In this section, we evaluate our depth results on KITTI Eigen split set and compare our models to recent supervised and unsupervised approaches. Then, we evaluate our flow results on KITTI2015 dataset. Moreover, we evaluate our pose results on KITTI odometry dataset. Table 1 shows the results of depth evaluation. Table 2 reports the results of optical flow evaluation. In Table 1 and Table 2, bold in each column represents the best result in each category. Table 3 gives the results of camera pose estimation. Table 4 shows the results of ablation study. In Table 3 and Table 4, bold in each column represents the best result.

4.1 Evaluation criteria

For evaluation, we use the following metrics used by previous work:

- Mean absolute relative difference (Abs Rel):

$$AbsRel = \frac{1}{|N|}\sum_{y\in N}\frac{|y - y^*|}{y^*}. \tag{17}$$

- Mean squared relative difference (Seq Rel):

$$SeqRel = \frac{1}{|N|}\sum_{y\in N}\frac{|y - y^*|^2}{y^*}. \tag{18}$$

- Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{y\in N}|y - y^*|^2}. \tag{19}$$

- Log RMSE:

$$RMSElog = \sqrt{\frac{1}{N}\sum_{y\in N}|logy - logy^*|^2}. \tag{20}$$

- The accuracy with threshold t:

$$\delta = max(\frac{y}{y^*}, \frac{y^*}{y}) < t, \tag{21}$$

  where $t \in [1.25, 1.25^2, 1.25^3]$.
- Endpoint error (EPE):

$$EPE = \frac{1}{N}\sqrt{(u-u^*)^2 + (v-v^*)^2}, \tag{22}$$

  where u and v denote the estimated horizontal and vertical flow vectors. u^* and v^* denote the horizontal and vertical ground truth vectors.

From Eqs. (15) to (19), y denotes the estimated depth, and y^* denotes the ground truth of depth.

Table 1 Performance comparison on KITTI Eigen split dataset

Method	Training data	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
		Lower the better				Higher the better		
Eigen et al. (C) [4]	Single image	0.214	1.605	6.563	0.292	0.673	0.884	0.957
Eigen et al. (F) [4]	Single image	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu et al. [13]	Single image	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Godard et al. [7]	Stereo pair	**0.148**	1.344	5.927	0.247	**0.803**	**0.922**	0.964
Garg et al. [6]	Stereo pair	0.152	**1.226**	**5.849**	**0.246**	0.784	0.921	**0.967**
Zhou et al. [35]	Monocular video	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Yang et al. [28]	Monocular video	0.156	1.360	6.641	0.248	0.750	0.914	0.969
Mahjourian et al. [14]	Monocular video	0.163	1.240	6.220	0.250	0.762	0.916	0.968
Yang et al. [27]	Monocular video	0.162	1.352	6.276	0.252	–	–	–
Yin and Shi [29]	Monocular video	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Ours	Monocular video	**0.152**	**1.103**	**5.608**	**0.230**	**0.796**	**0.935**	**0.974**

Table 2 Performance comparison on KITTI2015 flow training dataset

Method	Supervised	KITTI2015 Train (AEPE)
FlowNetS [3]	Yes	14.19
FlowNetC [3]	Yes	11.49
FlowNet2.0 [9]	Yes	**10.06**
PWC-Net [20]	Yes	10.35
Ren et al. [17]	No	16.79
Yin and Shi [29]	No	10.81
Ours	No	**10.19**

Table 3 Absolute trajectory error (ATE) on the KITTI odometry dataset

Method	Seq.09	Seq.10
ORB-SLAM [15]	0.014 ± 0.008	0.012 ± 0.011
Zhou et al. [35]	0.021 ± 0.017	0.020 ± 0.015
Mahjourian et al. [14]	0.013 ± 0.010	0.012 ± 0.011
Ours	$\mathbf{0.012 \pm 0.013}$	$\mathbf{0.012 \pm 0.007}$

Table 4 Ablation study

Model	Depth			Flow
	Abs Rel	Sq Rel	RMSE	AEPE
Ours (w/o FR)	0.155	1.296	5.857	10.81
Ours (full)	**0.152**	**1.103**	**5.608**	**10.19**

4.2 Training details

We use TensorFlow [1] framework for the training and test phases on a single Nvidia 1080Ti GPU and use the Adam optimizer [10] with $\beta_1 = 0.9$, $\beta_2 = 0.999$. Our network is trained and evaluated on KITTI raw dataset with a different split. The entire training strategy can be divided into two stages. The first stage is the joint learning of depth and camera pose. The total number of iterations is set to 350k. We use KITTI Eigen split to train and test our network. The second stage is joint learning of depth, camera pose and flow. The total number of iterations is set to 1500k. The input image sequences are cropped to $128 * 416$. The learning rate is set to $2 * 10^{-4}$. The batch size is set to 4. According to the previous works [7, 29, 36], the parameters are set to $\alpha = 0.85$, $\alpha_1 = 0.01$ and $\alpha_2 = 0.5$. According to the previous work [29], the loss weights are set to $\lambda_{br} = 1$, $\lambda_{fr} = 1$, $\lambda_{ds} = 0.5$, $\lambda_{fs} = 0.2$ and $\lambda_{fc} = 0.2$.

4.3 Dataset

We use the KITTI dataset as training and test sets. The dataset provides videos, 3D point clouds from LIDAR and vehicle moving trajectory in automatic driving scenes captured by stereo RGB cameras. In this paper, we only use the monocular RGB image sequences to train our network. Moreover, we use the split proposed in [4] consisting of about 40000 frames for training, 4000 frames for validation and 697 frames for testing, which covers a total of 29 scenes. We crop the images with a size of $128 * 416$. To make a fair comparison, we use the same cropping size as other approaches.

4.4 Depth evaluation

In Table 1, we compare our network to single-task-based methods [4, 6, 7, 13] and recent multi-task-based methods [14, 27–29, 35]. The error and accuracy metrics are computed over the Eigen [4] test set. Note that "C" and "S" of [4] denote coarse and fine versions. According to training data, we roughly divide these approaches into three categories: single image, stereo image pair and monocular video. We can find that Garg et al. [6] and Godard et al. [7] can achieve promising results among the single-task-based methods. In terms of two metrics of Abs Rel and $\delta < 1.25$, our model achieves comparable performance against Godard et al. [7]. The [7] is a stereo-based approach, which needs various calibrated stereo image pairs during training. In contrast, our network only needs monocular image sequences during training. Compared to the supervised methods [4, 13], our model can outperform these methods by a large margin. Our approach outperforms all of the fives prior multi-task-based methods including Zhou et al. [35], Yang et al. [28], Mahjourian et al. [14], Yang et al. [27] and Yin and Shi [29], which demonstrates the effectiveness of our proposed method. A qualitative comparison is visualized in Fig. 5. We compare our model to the work [35]. We can find that our model can produce sharper, detailed and accurate depth map. Moreover, our model can preserve details at object regions, such as motion boundaries, trees, poles and vehicles.

4.5 Optical flow evaluation

We evaluate the optical flow prediction performance on the KITTI2015 dataset. The results are shown in Table 2. We compare our method to both supervised and unsupervised methods including FlowNetS [3], FlowNetC [3], FlowNet2.0 [9], PWC-Net [20], Ren et al. [17] and Yin and Shi [29]. Compared to the supervised methods FlowNetS [3], FlowNetC [3], FlowNet2.0 [9] and PWC-Net [20], our model Our method performs slightly worse than FlowNet2.0 [9]. However, FlowNet2.0 [9] is a computationally complex neural network architecture for optical flow estimation, which stacks several networks to form a large network. Moreover, it needs a large amount labeled data for training. Compared to the unsupervised methods Ren et al. [17] and Yin and Shi [29], our model shows significant performance. Figure 6 shows the visual comparison. We can find that our method produces clearer object contours and motion boundary than the work [29].

4.6 Pose evaluation

For completeness, we provide a comparison of the pose network. Table 3 reports the ego-motion comparison of our model and [14, 15, 35] on the KITTI odometry dataset. We divide the 11 sequences with ground truth into two parts: the 00-08 sequences for training and the 09-10 sequences for testing. In Table 3, ORB-SLAM [15] is a traditional method, and Zhou et al. [35] and Mahjourian et al. [14] are deep learning-based methods. We find that our proposed method significantly outperforms [35] and achieves the results close to [14, 15].

4.7 Ablation study

Furthermore, we design two different variants of our network to prove the effectiveness of feature refinement module (FR). In the first variant "Ours (w/o FR)," we remove the feature refinement module. In the second variant "Ours," we add the feature refinement module. The results prove that using the FR module can reduce the error of both depth and flow tasks. The results of the two variants are shown in Table 4. Experimental results prove that

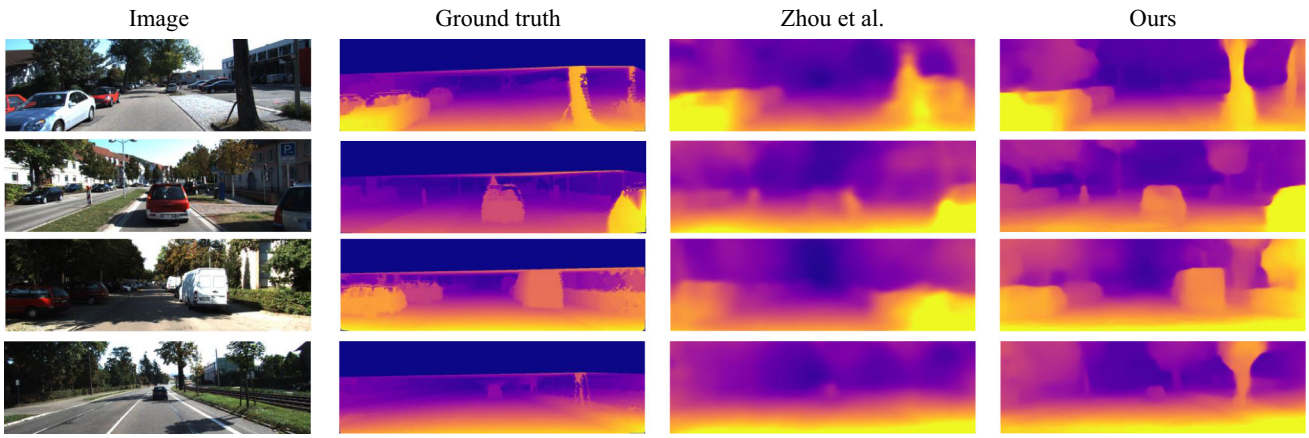

Fig. 5 Qualitative results of depth estimation on KITTI dataset. In each row left to right: image, ground truth flow and two predictions: Zhou et al. [35] and Ours

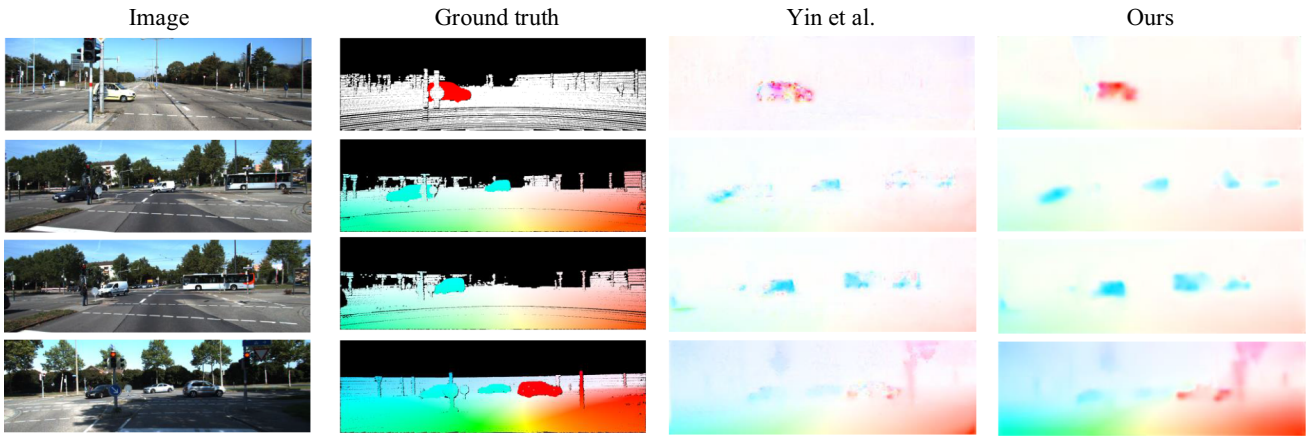

Fig. 6 Qualitative results of optical flow estimation on KITTI dataset. In each row left to right: image, ground truth flow and two predictions: Yin and Shi [29] and Ours

adding feature refinement module improves the performance of depth and optical flow estimation significantly.

4.8 Running time

Efficiency is crucial for real-world applications. We test our model on a single Nvidia 1080Ti GPU on the KITTI 2015 dataset. The running time listed in Table 5 is the average time on all frames. We show the total time consumed by a forward calculation. In Table 5, "Ours (w/o FR)" denotes the model without using the FR module. From Table 5, we can find that using the FR module increases the running time of about 27ms. Moreover, our model cannot achieve real-time performance. For real-world applications, the model only can be used for post-processing of the collected data. The efficiency problem for real-time applications will be considered in our future work.

4.9 Limitations

Figures 7 and 8 show some failure cases of optical flow estimation and depth prediction. In Fig. 7, from the first row, we can find that some estimation errors occur when the vehicular gap is close. From the second row, we can see that our model cannot handle the small displacement in the scene well. The low-speed motion of the vehicle is not extracted. From the third row, we can find that the estimated flow map is blurry in the night scenario. In Fig. 8, from the first and second rows, we can find that the depth information in the region of the thin rod cannot be extracted well. From the third row, we can find that the estimated depth map loses texture details in the tree region. The above experiments show some limitations of our proposed approach. We leave these limitations as future works.

Table 5 Running time

Model	Running time
Ours (w/o FR)	232ms
Ours	259ms

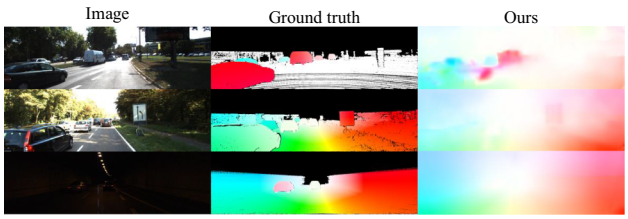

Fig. 7 Some failure cases of optical flow estimation. In each row left to right: image, ground truth flow and our result

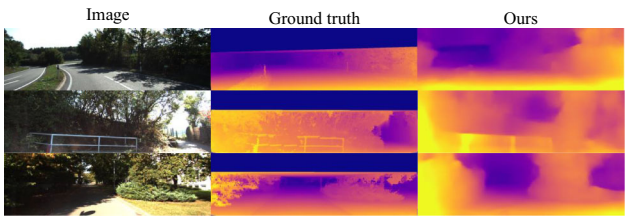

Fig. 8 Some failure cases of depth estimation. In each row left to right: image, ground truth depth and our result

5 Conclusion and future work

This paper focuses on geometry understanding from autonomous driving scenarios, which is an under-studied and very challenging problem. In this paper, we propose a novel deep learning framework based on feature refinement for estimating depth, optical flow and camera ego-motion.

It employs a feature refinement module to discriminate and recalibrate features adaptively alone two aspects: channel and spatial. The feature refinement module is embedded in both flow estimation sub-network and depth prediction sub-network, which can provide global channel-wise and spatial information for these two tasks. The experimental results on KITTI dataset prove the effectiveness of the proposed approach.

In the future, we plan to introduce a context-aware module into our model to extract more texture details. Moreover, we plan to further design an additional module to extract motion and depth from the night scenario. In addition, it will be interesting to investigate a lightweight network to improve the performance of real-time process.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grant 61401113, in part by the Natural Science Foundation of Heilongjiang Province of China under Grant LC201426, in part by the Fundamental Research Funds for the Central Universities of China under Grant 3072020CF0807 and in part by the Ph.D. Student Research and Innovation Fund of the Fundamental Research Funds for the Central Universities under Grant 3072019GIP0807.

Compliance with ethical standards

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, et al (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. ArXiv: 1603.04467
2. Chen S, Tan X, Wang B, Hu X (2018) Reverse attention for salient object detection. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer vision—ECCV 2018. Springer International Publishing, Cham, pp 236–252
3. Dosovitskiy A, Fischer P, Ilg E, Häusser P, Hazirbas C, Golkov V, vd Smagt P, Cremers D, Brox T (2015) Flownet: Learning optical flow with convolutional networks. In: 2015 IEEE international conference on computer vision (ICCV). pp 2758–2766
4. Eigen D, Puhrsch C, Fergus R (2014) Depth map prediction from a single image using a multi-scale deep network. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ (eds) Advances in neural information processing systems 27. Curran Associates, Inc., Red Hook, pp 2366–2374
5. Fu H, Gong M, Wang C, Batmanghelich K, Tao D (2018) Deep ordinal regression network for monocular depth estimation. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 2002–2011
6. Garg R, Bg VK, Carneiro G, Reid I (2016) Unsupervised CNN for single view depth estimation: geometry to the rescue. In: Leibe B, Matas J, Sebe N, Welling M (eds) Computer vision—ECCV 2016. Springer International Publishing, Cham, pp 740–756
7. Godard C, Aodha OM, Brostow GJ (2017) Unsupervised monocular depth estimation with left-right consistency. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). pp 6602–6611
8. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 7132–7141
9. Ilg E, Mayer N, Saikia T, Keuper M, Dosovitskiy A, Brox T (2017) Flownet 2.0: Evolution of optical flow estimation with deep networks. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). pp 1647–1655
10. Kingma D, Ba J (2018) Adam: a method for stochastic optimization. ArXiv: 1412.6980
11. Laina I, Rupprecht C, Belagiannis V, Tombari F, Navab N (2016) Deeper depth prediction with fully convolutional residual networks. In: 2016 Fourth international conference on 3D vision (3DV). pp 239–248
12. Lee W, Chuang P, Wang YF (2019) Perceptual quality preserving image super-resolution via channel attention. In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp 1737–1741
13. Liu F, Shen C, Lin G, Reid I (2016) Learning depth from single monocular images using deep convolutional neural fields. IEEE Trans Pattern Anal Mach Intell 38(10):2024–2039
14. Mahjourian R, Wicke M, Angelova A (2018) Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 5667–5675
15. Mur-Artal R, Montiel JMM, Tardós JD (2015) ORB-SLAM: A versatile and accurate monocular slam system. IEEE Trans Robot 31(5):1147–1163
16. Newcombe RA, Lovegrove SJ, Davison AJ (2011) DTAM: Dense tracking and mapping in real-time. In: 2011 International conference on computer vision. pp 2320–2327
17. Ren Z, Yan J, Ni B, Liu B, Yang X, Zha H (2017) Unsupervised deep learning for optical flow estimation. In: AAAI conference on artificial intelligence (AAAI)
18. Snavely N, Seitz SM, Szeliski R (2008) Modeling the world from internet photo collections. Int J Comput Vis 80(2):189–210
19. Song K, Yang H, Yin Z (2019) Multi-scale attention deep neural network for fast accurate object detection. IEEE Trans Circ Syst Video Technol 29(10):2972–2985
20. Sun D, Yang X, Liu M, Kautz J (2018) PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: 2018 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp 8934–8943
21. Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). pp 6450–6458
22. Wang H, Wang L (2018) Cross-agent action recognition. IEEE Trans Circ Syst Video Technol 28(10):2908–2919
23. Wang X, Hu Y, Radwin RG, Lee JD (2018) Frame-subsampled, drift-resilient video object tracking. In: 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp 1573–1577
24. Wang Y, Yang Y, Yang Z, Zhao L, Wang P, Xu W (2018) Occlusion aware unsupervised learning of optical flow. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 4884–4893
25. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
26. Woo S, Park J, Lee JY, Kweon IS (2018) CBAM: Convolutional block attention module. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer vision—ECCV 2018. Springer International Publishing, Cham, pp 3–19
27. Yang Z, Wang P, Wang Y, Xu W, Nevatia R (2018) Lego: Learning edge with geometry all at once by watching videos. In:

2018 IEEE/CVF conference on computer vision and pattern recognition. pp 225–234
28. Yang Z, Wang P, Xu W, Zhao L, Nevatia R (2018) Unsupervised learning of geometry from videos with edge-aware depth-normal consistency. In: AAAI conference on artificial intelligence
29. Yin Z, Shi J (2018) Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 1983–1992
30. Yu T, Wang L, Da C, Gu H, Xiang S, Pan C (2019) Weakly semantic guided action recognition. IEEE Trans Multimed 21(10):2504–2517
31. Zhan H, Garg R, Weerasekera CS, Li K, Agarwal H, Reid IM (2018) Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In: 2018 IEEE/CVF conference on computer vision and pattern recognition. pp 340–349
32. Zhang C, Kim J (2019) Modeling long- and short-term temporal context for video object detection. In: 2019 IEEE international conference on image processing (ICIP). pp 71–75
33. Zhang H, Ji Y, Huang W, Liu L (2019) Sitcom-star-based clothing retrieval for video advertising: a deep learning framework. Neural Comput Appl 31(11):7361–7380
34. Zhang Y, Li K, Li K, Wang L, Zhong B, Fu Y (2018) Image super-resolution using very deep residual channel attention networks. In: Computer vision—ECCV 2018. Springer International Publishing, Cham, pp 294–310
35. Zhou T, Brown M, Snavely N, Lowe DG (2017) Unsupervised learning of depth and ego-motion from video. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). pp 6612–6619
36. Zhu Y, Newsam S (2018) Learning optical flow via dilated networks and occlusion reasoning. In: 2018 25th IEEE international conference on image processing (ICIP). pp 3333–3337
37. Zhu Y, Zhao C, Guo H, Wang J, Zhao X, Lu H (2019) Attention couplenet: fully convolutional attention coupling network for object detection. IEEE Trans Image Process 28(1):113–126