IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 31, 2020, accepted January 25, 2021, date of publication March 17, 2021, date of current version April 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3062673

Learning Instance Motion Segmentation With Geometric Embedding

ZHEN LENG, JING CHEN, (Member, IEEE), AND SONGNAN LIN
School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China
Corresponding author: Zhen Leng (hhylx1950@gmail.com)

ABSTRACT Most existing deep learning-based motion segmentation methods treat motion segmentation as a binary segmentation problem, which is generally not the real case in dynamic scenes. In addition, the object and camera motion are often mixed, making the motion segmentation problem difficult. This paper proposes a joint learning method which fuses semantic features and motion clues using CNNs with deformable convolution and a motion embedding module, to address multi-object motion segmentation problem. The deformable convolution module serves to fusion color and motion information. And the motion embedding module learns to distinguish objects' motion status with inspiration from geometric modeling methods. We perform extensive quantitative and qualitative experiments on benchmark datasets. Especially, we label over 9000 images of KITTI visual odometry dataset to help training the deformable module. Our method achieves superior performance in comparison to the current state-of-the-art in terms of speed and accuracy.

INDEX TERMS Supervised learning, motion segmentation, video object segmentation.

I. INTRODUCTION

Motion segmentation, a key challenge in computer vision, aims at partitioning an image into regions of homogenous motion on the pixel level in moving camera videos. Motion segmentation has been shown to benefit a variety of applications such as autonomous driving [1], augmented reality [2], and human-computer interaction [3]. Different from semantic segmentation that classifies pixels with appearance cues only, motion segmentation primarily uses motion information to detect the object to be segmented. For stationary cameras, the moving pixels can be estimated accurately [4]. While the camera is moving, segmenting moving objects becomes more challenging since most image pixels start to move due to camera motion.

To solve the problem, most state-of-the-art algorithms [5]–[9] geometrically model the motion of cameras, scenes, and objects and then group the pixels according to the geometric motion model. Reference [10] use Dual-mode single Gaussian model(SGM) to prevent the background model from being contaminated by foreground pixels. Reference [11] cast the motion segmentation problem as point trajectories clustering and solve the problem with the multicut optimization. Another set of methods [12], [13] analyzes optical how between a pair of frames to group pixels with consistent flows into different motion regions. However, motion estimation based on optical flow remains a long-standing challenge. On the one hand, the 2D projection of the motion of faraway objects and the objects moving along the camera is insignificant and hard to model. On the other hand, pixel-wise motion segmentation based on optical flow often is built with sophisticated probabilistic models, which are sensitive to handpicked parameters and need complicated object inference schemes [4].

Recently, convolutional neural networks (CNNs) have been developed for motion segmentation and shown promising results. Early deep learning-based methods [14] take predicted image plane motion (optical flow) as a prior and utilize the high capacity of deep learning models to map the optical flow into corresponding segmentation masks. The performance of these methods, however, may suffer difficulties when different parts of an object exhibit nonhomogeneous motion patterns. Recently, [13] proposes a way to the fusion of the networks for deep semantic and deep optical flow. The motion features and semantic features are integrated with multiplication operations to improve the performance of both tasks. Whereas these methods are operational, we believe that it still needed to mine the relationship between these two features. Motion features for partly moving objects are usually significant but need to be complemented by semantic features to detect the whole object. Also, most deep learning-based motion segmentation methods focus on the

The associate editor coordinating the review of this manuscript and approving it for publication was Claudio Cusano.

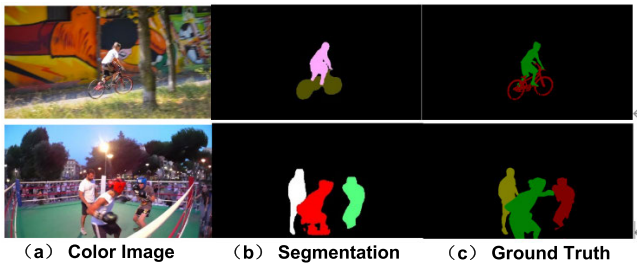

FIGURE 1. Segmentation results in two motion sequences. Our motion segmentation results are in (b) And ground truths are given in (c).

binary segmentation mask, where pixels are classified as either moving or part of the background, lacking the ability to distinguish moving objects with different moving status.

To distinguish objects with different motion, we propose a method which represents object motions with embeddings whose distance denotes the relative similarity between motions. Compared with the traditional model selection and motion clustering process, we model motions with neural networks that simplify the process. Motion property derives from dense optical flow. After that, we combine the instance segmentation with our motion embedding network to detect multiple moving objects in a single image.

Moreover, instead of fusing motion features from optical flows and semantic features in a naïve way, we propose a custom fusion module that fuses deep motion and semantic features with a deformable convolution operator. Unlike previous methods, our method can adaptively focus on moving parts of the image by enhancing the convolution kernel with dynamic offsets computed from both color and optical flow information. Given different prior information, the module will compute the corresponding convolution kernel to sample the image part with significant motion.

Additionally, to further train and evaluate our model on the real urban dataset, which is very important for autonomous driving and augmented reality applications, we construct an annotated dataset based on the KITTI dataset that is labeled with instance segmentation and minimal human labor.

Fig.1 shows the results of our motion segmentation algorithm on the Davis dataset. It is can be seen our method can find difference between different objects and can segment them regardless of their moving patterns.

The main contributions of our method include:

- We propose an enhanced deformable convolution module that incorporates the appearance and motion features for motion segmentation with adaptively computed convolution kernel.
- We propose a motion embedding method which fuses model selection and motion representation into a single neural network, in which optical flow patches are encoded into fixed-length vectors to distinguish different moving objects.
- Compared with the state-of-the-art methods, the quantitative and qualitative evaluation on multiple challenging datasets show that our method achieves multi-object motion segmentation and perform favorably against the state-of-the-art.

II. RELATED WORK

In this section, we give a brief overview of recent works in motion segmentation. Video object segmentation is a related topic sharing many similar techniques with motion segmentation. Therefore, we will also provide an overview here.

Motion Segmentation: Traditionally, motion segmentation is solved with motion information from the sparse trajectory which is the trajectory of sparse feature point movement over video frames [15]. The trajectories can be clustered into different subspaces with various clustering methods. Moving objects can be recognized with these different motion groups. For example, Haque *et al.* [16] use linear subspace to cluster feature point trajectories into different objects. Shen *et al.* [7] solve the clustering problem with the multi-cut algorithm. Bideau and Learned-Miller [17] segment an image according to motion boundaries calculated with optical flow. Zhang *et al.* [4] combine multiple geometric models which are suitable under different conditions and employs the ORK kernel to cluster point trajectories. However, these methods cannot realize dense image segmentation only with sparse trajectories. Besides, only rigid moving objects can be completely segmented, which limits its applications.

Another set of geometric methods model the motion status of every pixel as a probabilistic model with motion information from the optical flow. Narayana *et al.* introduce a probabilistic model that labels independent motions using optical flow orientations [20]. Similarly, Bideau and Learned-Miller adopt another probabilistic model to segment moving objects with the optical flow in an image sequence [10]. Their method is later extended to integrate semantic segmentation information extracted from neural networks [19] which greatly improves the segmentation results. These methods achieve dense segmentation and are not limited to the number of objects. Meanwhile, the object information can be integrated to segment non-rigid objects. However, these methods are sensitive to handpicked parameters and need complicated object inference schemes.

Recently, with the fast application of deep learning, a set of methods considers motion segmentation as a pixel-wise classification problem and employs convolution networks to deal with the task. Such methods usually use encoder-decoder style networks mapping optical flow to segmentation mask. Some methods [20], [21] combine the motion segmentation with other tasks such as visual odometry or depth prediction. Such a combination provides large datasets for training and is often used in autonomous driving scenes. Although unsupervised learning alleviates the need for a large amount of training data with ground-truth annotation, the precision suffers because the photometric consistency assumption often cannot hold in practical applications.

Other methods combine appearance cues to improve precision and deal with nonrigid moving objects [13], [22], [33].

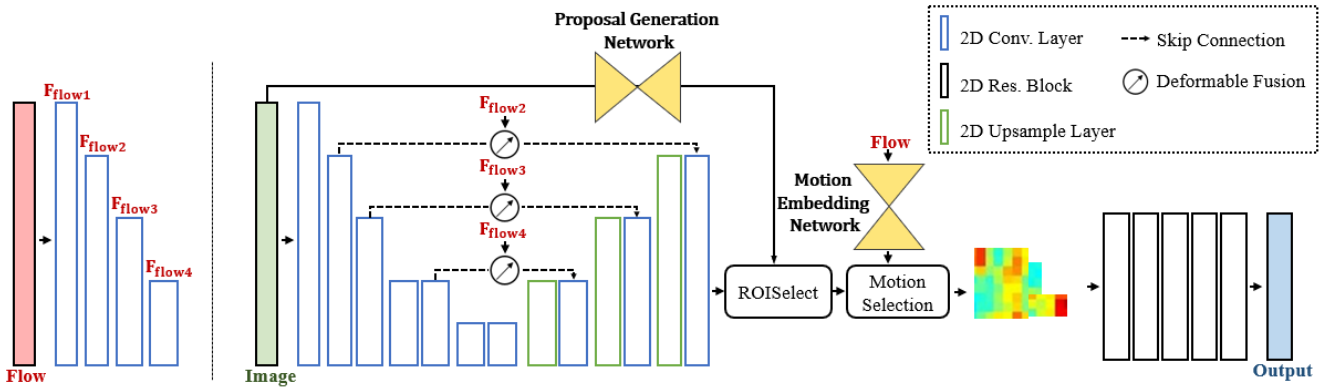

FIGURE 2. The overall architecture of our multi-object motion segmentation pipeline. Features extracted from the image and the optical flow are fused with the deformable fusion. ROISelect module from the mask RCNN serves to generate object proposals. The motion selection module further selects objects with significant motion according to the motion embedding. Motion masks are computed for moving objects.

Reference [13] combine semantic features and motion features and generate the segmentation mask with a decoder. Some methods additionally integrate RNN for multi frame information. Among them, Lim and Keles train a triplet framework to learn a mapping from image features to a foreground segmentation probability mask [22]. They evaluate the method in the static background. Tokmakov *et al.* employ CNNs to infer motion status from optical flow [11]. Later they propose to integrate motion information and semantic information in a recurrent network to achieve video object segmentation in video sequences [13]. Haque *et al.* propose an approach to fuse semantic features and motion cues to address the problem of monocular semantic motion segmentation [13]. However, the integration of motion and semantic features is still not fully exploited. Besides, the number of output layers is fixed for end-to-end networks which makes the segmentation of multiple objects difficult.

Video object segmentation: Similar to motion segmentation, video object segmentation takes continuous frames as input and generates segmentation masks for objects. However, video segmentation aims to segment the prominent foreground object. Unsupervised video segmentation methods which need to find the object of interest is closely related to motion segmentation in that many unsupervised method employ motion cues to find the target [23], [24]. These methods are often built upon the assumption that motion is different for foreground objects. Recent unsupervised video object segmentation [25] employs RNN to achieve multi-frame information integration and attention mechanism is often used to locate the object of interest. Some methods [18] jointly solve motion segmentation and video object segmentation tasks. Whereas related, this line of work is different since objects segmented are consistent during the video regardless of their motion status.

III. MOTION SEGMENTATION

Our multi-object motion segmentation method consists of four main steps, that is, feature enhancement, object proposal generation, motion detection, and fine segmentation. Fig 2. shows an illustration of our neural network. To achieve reliable multi-object motion segmentation, we first fuse color and motion features to detect possibly moving objects with the feature enhancement and proposal generation module. Then, the moving status of the motion patches is identified with our motion embedding module. Finally, the object masks are segmented in the final segmentation step from the image patches. Compared with Mask-RCNN, we integrate motion information in both feature extraction and proposal filtering stage. The detail of our method is given in Fig.2.

A. MOTION EMBEDDING

To make motion segmentation adapt to various geometric models, we propose a motion embedding module which encodes optical flow patches into fixed-length vectors with a data-driven approach. The idea is that different relative motion between camera and object can be mapped to different embedding representation according to the respected geometric model. For instance, when the underlying motion is general, a fundamental matrix is used to model the epipolar geometry, and when scene-motion is degenerate like a planar scene or a pure rotation, homography is preferred [26]. For a general 3D scene, the relationship between 3D motion and 2D optical flow can be described as follows:

$$u(x,y) = -f_c\left(\frac{T_X}{Z} + R_Y\right) + \frac{xY_z}{Z} + yR_z - \frac{x^2R_Y}{f_c} + \frac{xyR_x}{f_c}$$
$$v(x,y) = -f_c\left(\frac{T_Y}{Z} + R_X\right) + \frac{yY_z}{Z} - xR_z + \frac{y^2R_X}{f_c} - \frac{xyR_Y}{f_c}$$

where f_c is the focal length, Z is the depth of the 3D point, and u(x, y), v(x, y) denotes the image plane motion at coordinate(x, y). However, the real world scene-motions are not so conveniently divided. They are more typified by near-degenerate scenarios such as a scene that is almost but not quite planar, or a motion that is rotation-dominant but with a non-vanishing translation. In such cases, imposing a false dichotomy in deciding an appropriate model would pose difficulty for subsequent separation.

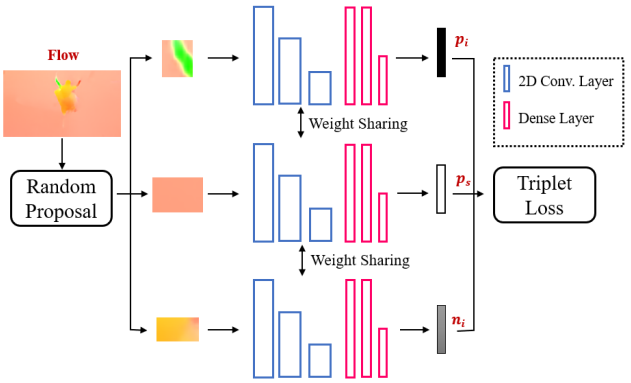

FIGURE 3. The training framework for our proposed motion embedding network. Optical flow patches are encoded by our motion embedding module. Supervision signal induced by the triplet loss pushes the network toward generating embeddings with motion information.

Inspired by the geometry studies in traditional motion segmentation methods, we propose a novel network module that performs model selection and motion representation in a single step, in which the motion representations are encoded as fixed-length embeddings containing geometric motion information from the optical flows. As illustrated in Fig.3, our motion embedding module is a multi-layer perceptron prepended with CNN layers. Given small image patches as input, the output vector will be a D-dimensional vector $V_D = Net_\theta(Img_{HxW})$. We further normalize the output vector to ease the comparison between different motions.

To deal with multiple moving components within patches, we use max-pooling to highlight the most prominent motion in the image patch. The network module is trained end-to-end with the triplet loss to maximize the ability of the module to encode different motions. The training framework is shown in Fig.3. The input training image is selected that only one moving object exists. Then we randomly crop three patches from the image and make sure the three patches contain both the moving object and static environment. The cropped image patches are classified as moving if the motion mask covers most of the patch. We denote the moving patches as the positive sample group $p_{\{1,2,...w\}}$ and the static patches as the negative sample group $n_{\{1,2,...q\}}$. The training loss is designed so that encodings from samples of the same group present smaller distance than encodings from different groups. The training loss is defined as:

$$loss = \max\left(||f(p_i) - f(p_s)||^2 - ||f(p_i) - f(n_s)||^2 + \alpha, 0\right) + \max(||f(n_i) - f(n_s)||^2 - ||f(n_i) - f(p_s)||^2 + \alpha, 0)$$

p_i and p_s are samples from the positive group whereas n_i and n_s are samples from the negative group. α is a bias parameter to avoid zero losses.

During the evaluation, the input motion patch comes from the previous proposal step instead of randomly cropped patches in the training dataset. We compute the motion embedding for both the proposed object patch and the whole image. The distance between the motion embeddings is used to select the moving objects with the ordered residual kernel (ORK). Moving object masks are segmented form the selected moving patches using the Mask-RCNN pipeline.

B. FEATURE ENHANCEMENT MODULE

Given an optical flow frame and the corresponding color image, the proposed feature enhancement module will generate corresponding motion aware features for further segmentation. It mainly consists of two networks: the motion feature extraction and the deformable fusion.

Motion Feature Extraction: This module extracts motion features via a feature extraction network. The network consists of convolutional layers with ReLUs as the activation function. In our implementation, we adopted a modified residual structure from [27]. The extracted features will be utilized for feature-wise temporal alignment. To remove the interference of the scene depth, the input optical flow is preprocessed into separate motion angle and magnitude which would help motion feature extraction as mentioned in [12].

Deformable Fusion: The goal of the deformable fusion module is to take the semantic feature F_{img} and optical flow feature F_{flow} to predict sampling parameters θ for the fused feature F_{sf}:

$$\theta = f_\theta(F_{img}, F_{flow})$$

Here, $\theta = \{\Delta p_n | n \in R$ refers to the offsets of the convolution kernels, where $R = \{(-1,-1), (-1,0), \ldots, (0,1), (1,1)\}$ donates a regular grid of a 3×3 convolution kernel. With θ and F_{img}, the fused feature F_{sf} can be computed by the deformable convolution:

$$F_{sf} = f_{deform}(F_{img}, \theta)$$

More specifically, for each position p_0 on the fused feature map F_{sf}, we have:

$$F_{sf}(p_0) = \sum_{p_n \in R} w(p_n) F_{img}(p_0 + p_n + \Delta p_n)$$

The convolution will operate on the irregular position $p_n + \Delta p_n$ which is fractional. To address the issue, we implement the operation with bilinear interpolation, which is the same as [7]. Here, the deformable alignment module consists of several regular and deformable convolutional layers. The visualization is given in Fig.4. The original deformable convolution utilizes semantic features for both offset computation and deformable convolution input. Oppositely our method concatenates both semantic and motion features in the offset computation. The sampling parameter generation function f_θ concatenates F_{img} and F_{flow} and uses a 3×3 bottleneck layer to reduce the channel number of the concatenated feature map. Then, the sampling parameters are predicted by a convolutional layer with the kernel size as the output channel number. Finally, the fused feature F_{sf} obtained from θ and F_{img} based on deformable convolution operation. In practice, we use three deformable convolutional layers in the network to enhance the fusion. The learned features will

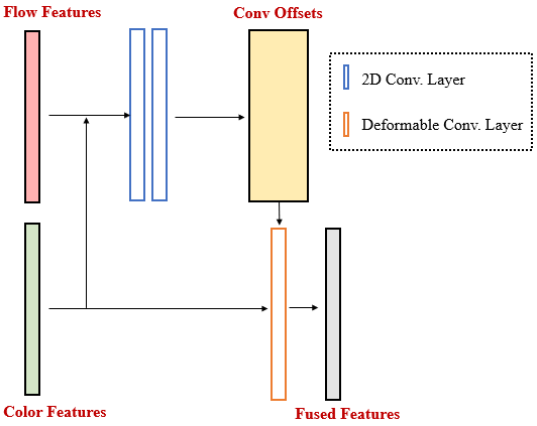

FIGURE 4. Our deformable fusion module.

implicitly capture motion cues and also explore static regions within the same image structure.

C. DATASET ANNOTATION

To train the motion segmentation model, we need a large number of images with motion annotation. However, training datasets for motion segmentation in real scenes are scarce. Some methods use artificial images for training [11] which cannot integrate semantic information. Reference [18] annotate their own datasets. But the annotated datasets are either too small for training or biased toward a certain type of object.

We aim to generate a large number of coarse motion annotations for the KITTI dataset without much human labor. For each picture in the dataset, we use the state-of-the-art instance segmentation method PANet [28] to obtain masks for individual objects in the scene. Then human annotators identify the moving objects and select all corresponding masks to finish the annotation. Compared to fully manual annotations, this method can save a lot of time, whereas not lose much accuracy. We label more than 9000 samples with this method. We denote the dataset as KITTI-Moseg. The only downside is the annotation accuracy is subject to the instance segmentation method used. But in our experiment, the coarse annotation has proved useful in training.

IV. EXPERIMENTS AND EVALUATION

A. IMPLEMENTATION DETAILS

We implement the network with the PyTorch framework. Network training is done with the SGD algorithm with a batch size of 8 and a learning rate starting at 0.003. The learning rate gradually decreases to 3e-5 during training. the batchsize, momentum, and weight decay are set to 2,0.9, and 1e-5, respectively. The initial weights for the motion embedding and feature fusion module are set with the normal distribution. The data are augmented online with the horizontal and vertical flip, random resizing and rotations covering a range of degrees$(-10, 10)$. The network is implemented with PyTorch. Optical flow is calculated with PWC-Net [29] and unsampled to the same resolution with the color image. The embedding network is trained with patches from the flyingthings dataset [27] and annotations are provided by [30]. Other network components are pretrained with standard training procedures from Mask-RCNN [34]. Our network contains 63.06M parameters and the FLOPs is 104.7G. The training iteration costs 482ms and evaluation costs 183ms.

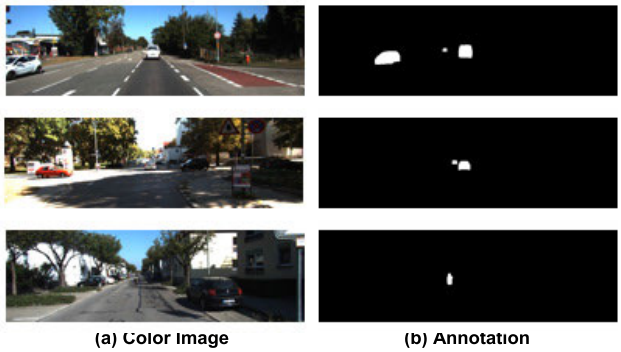

FIGURE 5. Our annotated KITTI-Moseg dataset.

The evaluation metrics we use are F-score and intersection over union (IOU) which are common in motion segmentation. We compare our results with both binary and multi-object motion segmentation methods. The datasets we use are given below.

B. DATASETS

KITTI-Moseg: The KITTI dataset contains urban and highway scenes collected by an autonomous driving platform. The sequences contain moving objects commonly seen on the road such as cars, pedestrians, and trucks. The motion pattern in the dataset is mostly forward, which is quite different from other motion segmentation dataset but quite common in driving scenes. The video odometry dataset contains 20 sequences in which we annotate the first 9 ones for training and the following two sequences for testing. Approximately ten thousand images are annotated.

FBMS: The Freiburg-Berkeley motion segmentation dataset [28] is an extension of the BMS dataset with 33 additional video sequences. A total of 720 frames is annotated. FBMS-59 comes with a split into a training set and a test set. Whereas some annotations do not fit the motion segmentation described above, we adopt the annotation provided by Keuper *et al.* [10] for evaluation. It contains complex object movements and deformations.

Davis: The Davis dataset [32] is a video segmentation dataset that includes two variants. In the 2016 variant, a single instance is annotated for each video. The 2017 variant contains multiple instances annotated. We use the dataset for evaluation and comparison on both binary and multi-object motion segmentation evaluation. We also exclude some sequences in evaluation since these sequences contain moving objects in the background.

C. EXPERIMENTS

We conduct experiments on FBMS, Davis and Complex [20] datasets to showcase the effect of our methods. We compare our binary motion segmentation results with LSMO [19]

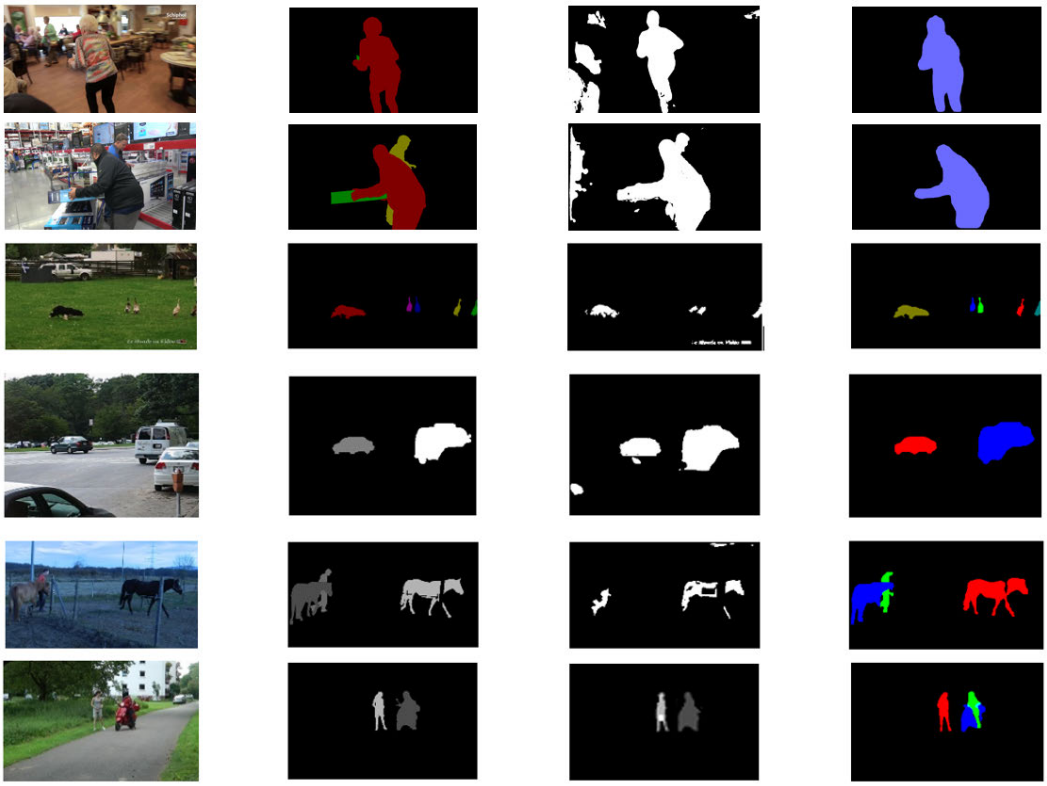

FIGURE 6. Results on multi-object motion segmentation. First two rows contain samples from the davis dataset, others are samples from the FBMS and complex dataset.

TABLE 1. Multi object motion segmentation.

Method	FBMS		Davis-16		Complex	
	IOU	F-score	IOU	F-score	IOU	F-score
CUT[9]	0.681	0.636	0.458	0.427	0.675	0.602
CCG[14]	0.611	0.563	0.469	0.437	0.536	0.458
BOB[21]	0.671	0.649	0.514	0.471	**0.649**	**0.656**
SROWN[42]	-	-	**0.619**	-	-	-
Ours	**0.683**	**0.672**	0.492	0.462	0.638	0.629

which is the state-of-the-art deep learning based motion segmentation methods. For multi-object motion segmentation, we compare with [7] which is the state-of-the-art. For a fair comparison, we don't use any post-processing techniques, such as the conditional random field(CRF) [32]. Standard metrics including boundary F-score and IOU are used to evaluate the performances.

1) RESULTS ON MULTI-OBJECT MOTION SEGMENTATION

Here we compare the results on multi-object segmentation of our methods with [7] and [14] in Table 1.

We can see that our method achieves a 3% increase in IOU and a 1% increase in F-score compared with the state-of-the-art methods in the FBMS dataset. The increase in precision can be attributed to our motion fusion and motion embedding module. We report the ablation study in section D.

Reference [4] is a feature clustering method that only give sparse results. Reference [21] is close-source and we take the results directly from the paper. Visualization is only presented for [16] which is the original version of [21] and our method. We can see from the results that our module can correctly segment different moving objects with similar appearance whereas ignore stationary objects. Compare with previous geometric motion segmentation methods, our methods show less cluster in segmentation results and achieve better boundary accuracy with the semantic information. On the FBMS dataset, our method performs better due to relatively large motion in the dataset. Also, we achieve higher segmentation precision with our feature fusion module and the RCNN-like architecture. However, motions of different objects are often more mixed in the Davis dataset, which poses a challenge on our ability to distinguish moving objects.

We also include additional visualization on some FBMS and Complex sequences to compare with [21] in Fig. 7 and visualizations with SROWN [39] in Fig. 8. Neither methods provide source code for evaluation. Thus we compare with the reported results in the paper. We outperform [21] on both FBMS and Complex datasets. Compared with [39], although our method lacks supervision, we still achieve comparable results.

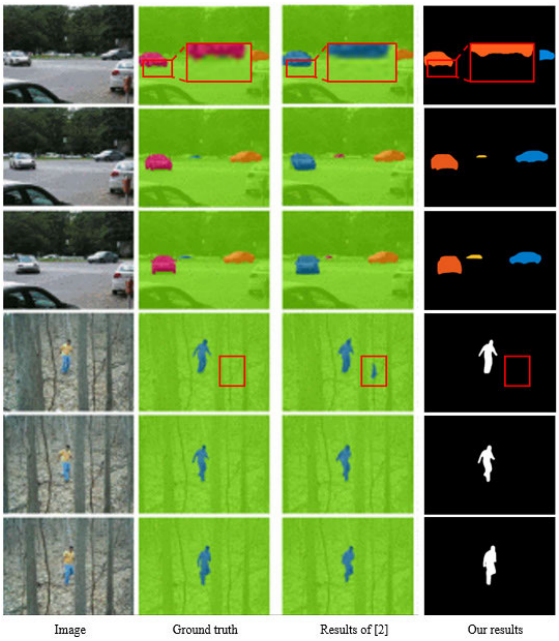

FIGURE 7. Additional results on FBMS and complex sequences to compare with [19].

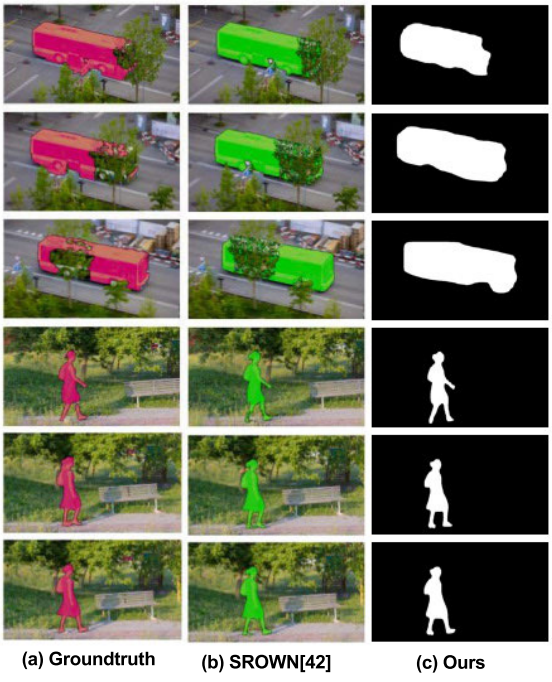

FIGURE 8. Comparison with SROWN [39] on the davis dataset.

2) RESULTS ON SINGLE-OBJECT MOTION SEGMENTATION

We repeat the test for single-object motion segmentation to compare with some other deep learning-based methods. We take the union of all moving objects as the foreground for our method. For the Davis dataset, we take the 2016 version which only contains one moving object per scene.

As seen, our method outputs promising results on the Davis-16 dataset, compared with existing top performing motion segmentation algorithms.

TABLE 2. Results on single-object motion segmentation.

Method	IOU	F-score
LSMO[29]	0.782	0.759
SMSNet[24]	0.772	0.743
LVO[19]	0.759	0.721
Ours	0.804	0.781

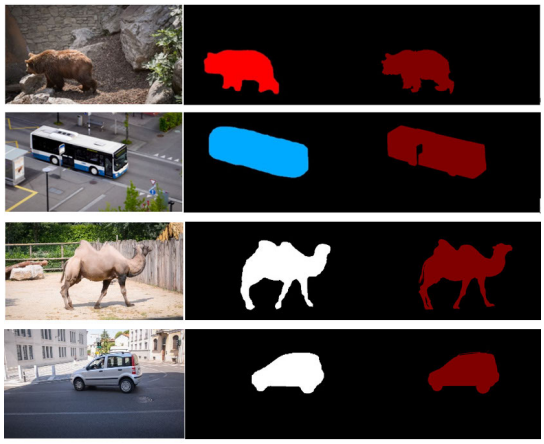

FIGURE 9. Results on single object motion segmentation. Our motion segmentation results are in (b) And ground truths are given in (c).

TABLE 3. Segmentation performance of our deformable module under different network variants.

Name	F-score	IOU
Concat-Vgg	0.89	0.54
Multiply-Vgg	0.89	0.56
Deformable-Vgg	0.91	0.58
Multiply-Resnet50	0.95	0.59
Deformable-Resnet50	0.96	0.64

Concat-Vgg, Multiply-Vgg and Deformable-Vgg come with Vgg [35] backbone whereas Multiply-Resnet50 and Deformable-Resnet50 are networks with Resnet50 [36] backbones.

D. ABLATION STUDY

In order to survey the effectiveness of our proposed feature enhancement module and motion embedding module, we conduct ablation experiments on Davis-17 by substituting the component with an ordinary convolution module.

1) MOTION FUSION MODULE

The baseline for comparison with our motion fusion module is a concatenation and convolution module. We apply both modules in an end-to-end framework [31] and compare their performances with the same training procedures in our KITTI-Moseg dataset. We also evaluate our module for two commonly seen network architectures to showcase the generality of our network.

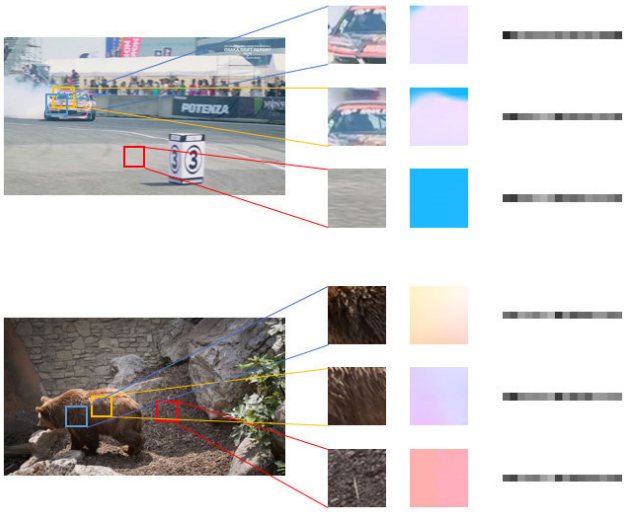

FIGURE 10. Image patches followed by corresponding optical flows and ground truth motion masks from different moving regions in an image. Moving status of the image patch on the second column is the same as that of the first whereas the third is different. Our motion can detect the moving status though their optical flows suggest otherwise.

Using both F-score and IOU indicators, we can see that the deformable convolution increases the segmentation performance compared with traditional integration schemes between the semantic and optical flow features for both Vgg and Resent backbones. The IOU decreases 6% without deformable convolution. The improvement can be attributed to the better fusion ability of our motion enhancement module.

2) MOTION EMBEDDING MODULE

We investigate the performance of our motion embedding module by measuring the ability to correctly discriminate objects with different motions despite the similar optical flow. We use the average of the optical flow as our baseline. The validation set is image patches from the Davis dataset with motion labels. We train our network with the flyingthings dataset.

We give a visualization of the training optical flow patches in fig 6. Different colors represent different direction and intensity of the containing optical flow. The training optical flow patches are grouped according to their prominent motion. Each training sample contains two patches with the same motion and the other is different. We give two groups in which the last one is the outliner. We also make experiments on some samples in the highly challenging synthetic flyingthings dataset. The results suggest that our method can adopt to various motions and present higher differentiation ability with precise optical flows.

The precision and recall are shown to showcase the ability of our method to differentiate between different motions. The results are in Table 4.

To showcase the effect of our motion embedding module, we remove the motion embedding module from our network and the relevant indicator decrease by 9∼15%.

TABLE 4. Distance comparison between Patches with different motion.

Image	DisSim[a]	DisDiff[b]
Image A	0.426	0.849
Image B	0.501	0.617
Synthetic A	0.125	0.541
Synthetic B	0.157	0.672

aDisSim refers to the distance between image patches with similar motion whilst bDisDiff refers to the distance between patches with different motion. There motion status is determined with ground truth motion masks.

TABLE 5. The ability to distinguish moving status of patches with our motion embedding module.

Method	Recall	Precision
OEM [a]	0.84	0.93
AOV [b]	0.75	0.78

[a] OEM represents results with our Motion Embedding Module.
[b] AOV represents results with Average of Optical Flow.

TABLE 6. Comparison with cross entropy loss.

	Module Precision	System mIOU
Cross Entropy	0.51	0.438
Triplet	**0.97**	**0.672**

TABLE 7. Comparison with different pretraining datasets.

	IOU	F-score
flyingthings	0.679	0.668
KITTI	**0.683**	**0.672**

Besides, the ability to recognize motion regardless of their appearance can be proofed by the evaluation results on the realistic Davis dataset despite training with the synthetic flyingthings dataset.''

We additionally include learning results in TABLE 5 with the cross entropy loss as follows: The cross entropy loss network is trained with the ground truth motion mask as the supervision signal. We use similar network structure for the cross entropy network. We can see our network performs much better than the cross entropy network which tries to learn the motion directly from the optical flow.

3) THE PROPOSED KITTI DATASET

Our proposed KITTI dataset annotation only contains binary segmentations instead of instance level segmentations. Thus we only pretrain optical flow encoder network with the dataset. In the experiments, we add experiments on the comparison between pretraining on the flyingthings and our annotated KITTI dataset. The results are compared by evaluating our the FBMS dataset.

Pretraining on our KITTI datasets shows superior results. This can be attributed to the motion pattern of our annotated dataset resembles the motion in real life.

V. CONCLUSION

In this paper, we have proposed a motion segmentation model that can differentiate between different moving objects. We introduced two mechanisms that enable the network to find moving objects. The feature enhancement module fuses motion and semantic information. The motion embedding module achieves motion clustering by mapping the motion of different objects to different encoding vectors. The experiments show that our proposed framework outperforms previous motion segmentation models on multiple challenging datasets.

REFERENCES

[1] H. Rashed, A. El Sallab, S. Yogamani, and M. ElHelw, "Motion and depth augmented semantic segmentation for autonomous navigation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jun. 2019, pp. 364–370.

[2] M. R. U. Saputra, A. Markham, and N. Trigoni, "Visual SLAM and structure from motion in dynamic environments: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, pp. 1–36, Jun. 2018.

[3] S. Drab and N. M. Artner, "Motion detection as interaction technique for games & applications on mobile devices," in *Proc. Pervasive Mobile Interact. Devices*, 2005, pp. 48–51. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.84.4764

[4] C. Zhang, Z. Liu, C. Bi, and S. Chang, "Dependent motion segmentation in moving camera videos: A survey," *IEEE Access*, vol. 6, pp. 55963–55975, 2018.

[5] X. Xu, L. F. Cheong, and Z. Li, "Motion segmentation by exploiting complementary geometric models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2859–2867.

[6] T. Zhuo, Z. Cheng, P. Zhang, Y. Wong, and M. Kankanhalli, "Unsupervised online video object segmentation with motion property understanding," 2018, *arXiv:1810.03783*. [Online]. Available: http://arxiv.org/abs/1810.03783

[7] J. Shen, J. Peng, and L. Shao, "Submodular trajectories for better motion segmentation in videos," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2688–2700, Jun. 2018.

[8] Y. Sugaya and K. Kanatani, *Automatic Camera Model Selection for Multibody Motion Segmentation*. Glen Burnie, MD, USA: MVA, 2002.

[9] Y. Sugaya and K. Kanatani, "Geometric structure of degeneracy for multibody motion segmentation," in *Proc. ECCV Workshop SMVP*, 2004, pp. 13–25.

[10] M. Keuper, B. Andres, and T. Brox, "Motion trajectory segmentation via minimum cost multicuts," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3271–3279.

[11] C. Zhang, J. Zheng, Y. Zhang, M. Han, and B. Li, "Moving object detection algorithm based on pixel spatial sample difference consensus," *Multimedia Tools Appl.*, vol. 76, no. 21, pp. 22077–22093, Nov. 2017.

[12] Y. Yang, Q. Zhang, P. Wang, X. Hu, and N. Wu, "Moving object detection for dynamic background scenes based on spatiotemporal model," *Adv. Multimedia*, vol. 2017, no. 28, pp. 1–9, 2017.

[13] P. Bideau, A. Roy Chowdhury, R. R. Menon, and E. Learned-Miller, "The best of both worlds: Combining CNNs and geometric constraints for hierarchical motion segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 508–517.

[14] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik, "Learning to segment moving objects in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4083–4090.

[15] S. D. Jain, B. Xiong, and K. Grauman, "Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2126.

[16] N. Haque, N. D. Reddy, and K. M. Krishna, "Joint semantic and motion segmentation for dynamic scenes using deep convolutional networks," 2017, *arXiv:1704.08331*. [Online]. Available: http://arxiv.org/abs/1704.08331

[17] P. Bideau and E. Learned-Miller, "It's moving! A probabilistic model for causal motion segmentation in moving camera videos," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 433–449.

[18] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 282–295.

[19] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2463–2472.

[20] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1777–1784.

[21] M. Narayana, A. Hanson, and E. Learned-Miller, "Coherent motion segmentation in moving camera videos using optical flow orientations," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1577–1584.

[22] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1851–1858.

[23] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, "SfM-Net: Learning of structure and motion from video," 2017, *arXiv:1704.07804*. [Online]. Available: http://arxiv.org/abs/1704.07804

[24] P. Tokmakov, K. Alahari, and C. Schmid, "Learning video object segmentation with visual memory," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4481–4490.

[25] L. A. Lim and H. Y. Keles, "Foreground segmentation using a triplet convolutional neural network for multiscale feature encoding," 2018, *arXiv:1801.02225*. [Online]. Available: http://arxiv.org/abs/1801.02225

[26] K. Fragkiadaki, G. Zhang, and J. Shi, "Video segmentation by tracing discontinuities in a trajectory embedding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1846–1853.

[27] Z. Cao, A. Kar, C. Haene, and J. Malik, "Learning independent object motion from unlabelled stereoscopic videos," 2019, *arXiv:1901.01971*. [Online]. Available: http://arxiv.org/abs/1901.01971

[28] J. Vertens, A. Valada, and W. Burgard, "SMSnet: Semantic motion segmentation using deep convolutional neural networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 582–589.

[29] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2004. [Online]. Available: https://www.cambridge.org/core/books/multiple-view-geometry-in-computer-vision/0B6F289C78B2B23F596CAA76D3D43F7A, doi: 10.1017/CBO9780511811685.

[30] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4040–4048.

[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.

[32] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8934–8943.

[33] P. Tokmakov, K. Alahari, and C. Schmid, "Learning motion patterns in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3386–3394.

[34] P. Ochs, J. Malik, and T. Brox, "Segmentation of moving objects by long term video analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1187–1200, Jun. 2014.

[35] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 724–732.

[36] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[37] X. Y. Stella and J. Shi, "Multiclass spectral clustering," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 313–319.

[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[40] L. Yang, J. Han, D. Zhang, N. Liu, and D. Zhang, "Segmentation in weakly labeled videos via a semantic ranking and optical warping network," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4025–4037, Aug. 2018, doi: 10.1109/TIP.2018.2834221.
[41] S. Nikitidis, S. Zafeiriou, and I. Pitas, "Camera motion estimation using a novel online vector field model in particle filters," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 8, pp. 1028–1039, Aug. 2008, doi: 10.1109/TCSVT.2008.927107.

ZHEN LENG was born in Henan, China, in 1994. He received the B.S. degree in optics engineering from the Beijing Institute of Technology, Beijing, in 2015, where he is currently pursuing the Ph.D. degree in optics engineering. His research interests include computer vision, visual odometry, and deep learning.

JING CHEN (Member, IEEE) was a Postdoctoral Research Fellow with the Graz University of Technology, Austria, in 2003. She is currently a Doctoral Supervisor and an Assistant Professor with the School of Optics and Photonics, Beijing Institute of Technology, Beijing. Her main research interests include augmented reality, human–computer interaction, visual SLAM, and deep learning.

SONGNAN LIN was born in Hebei, China, in 1993. She received the B.S. degree from the Department of Precision Instrument, Tsinghua University, Beijing, in 2015. She is currently pursuing the Ph.D. degree in optics engineering with the Beijing Institute of Technology, Beijing.

Her research interests include intersection of image processing, computer vision, and deep learning.