Flow-Edge Guided Unsupervised Video Object Segmentation

Yifeng Zhou, Xing Xu, Fumin Shen, Xiaofeng Zhu, and Heng Tao Shen

***Abstract*—Recently, deep learning techniques have achieved significant improvements in unsupervised video object segmentation (UVOS). However, many of existing approach cannot accurately identify the foreground objects and the background as they commonly use the coarse temporal features (*e.g.* , optical flow and multi-frames attention). In this paper, we present a novel model termed Flow Edge-based Motion-Attentive Network (FEM-Net), to address the unsupervised video object segmentation problem. Firstly, a motion-attentive encoder is used to jointly learn the spatial and temporal features. Then, a Flow Edge Connect (FEC) module is designed to hallucinate edges of the ambiguous or missing region in the optical flow. During the segmentation stage, the complementary temporal feature composed by the motion-attentive feature and flow edge is fed into a decoder to infer the salient foreground objects. Experimental results on two challenging public benchmarks (*i.e.* DAVIS-16 and FBMS) demonstrate that the proposed FEM-Net compares favorably against the state-of-the-art methods.**

***Index Terms*—Video Segmentation, Optical Flow, Attention Mechanisms, Deep Learning.**

I. Introduction

UNSUPERVISED video object segmentation (UVOS) is the task of separating the primary foreground objects from the background automatically. It has many applications, including autonomous driving, background exchanging, and video editing. Different from the supervised video object segmentation task (SVOS), which provides the label of target objects in the first frame, UVOS is more challenging because of the lack of prior knowledge. Due to the primary objects in the video are diversified, it is hard to separate them by only using the texture feature. In addition to that, the state of the object is changeable, *e.g.* in the case of the street, a parked bicycle is viewed as the background while the bicycle which is ridden by a rider is oppositely viewed as the foreground object. Unlikely with the SVOS, without the user interaction or the first label, the UVOS suffers from many difficult problems. However the UVOS is more applicable, not only can the UVOS handle the object that appears halfway, but it also alleviates the preparation for the prior knowledge.

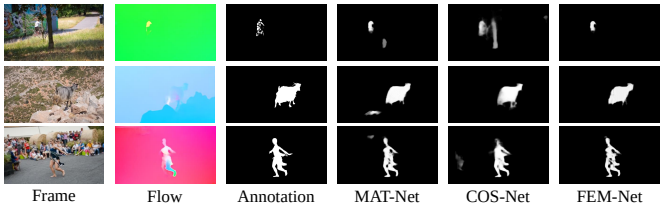


Fig. 1. With the guidance of connected flow-edge, FEM-Net can separate the moving objects accurately. The columns from left to right are the input frame, the optical flow, the segmentation ground-truth, the results of MATNet [1], COSNet [2], and FEM-Net, respectively.

A competent UVOS model should have three properties: 1) learning a powerful object representation from the frames sequences. 2) leveraging the temporal feature effectively to distinguish the primary foreground objects and the background, instead of segmenting frames independently. 3) being aware of the edge boundary and synthesize the boundary and mask of the objects jointly. Early traditional segmentation methods make use of the handcrafted features, *e.g.* turning point trajectories into dense regions [3]. Also, many approaches [4], [5] use color, texture and other image cues to guide the model learn a generic feature of foreground objects. With the increasing popularity of deep neural networks (DNNs), many learning-based approaches have achieved impressive results on image segmentation task [6], [7], [8]. However, generally applying these image-based segmentation algorithms independently to each video frame is time-consuming and unstable.

Using the ConvLSTM is a good way to bridge the image-based segmentation method to video level [9]. Many researchers also proposed to use multi-scale to handle various difficult scenarios such as illumination changes, background, or camera motion [10], [1]. Xu *et al.* [11] proposed a decision network to adaptively assign different frame regions to different networks, thus maximizing the usage of video redundancy. Lu *et al.* [2] proposed a co-attention module to embed the correlation between video frames and use the embedded feature to discover the foreground objects. Zhen *et al.* [12] consider conducting the semantic segmentation and boundary detection jointly in a pyramid way, therefore, these two features can benefit each other. These RGB-based segmentation methods are incompetent to extract temporal features unless they use more frames simultaneously.

In the UVOS, the motion feature (*i.e.* optical flow) is general used in video segmentation tasks to alleviate the compute cost

This work was supported in part by the National Key Research and Development Program of China (No. 2018AAA0102200); the National Natural Science Foundation of China (61976049, 61632007 and U20B2063); the Fundamental Research Funds for the Central Universities under Project (ZYGX2019Z015) and the Sichuan Science and Technology Program, China (2018GZDZX0032, 2019YFG0003, 2019ZDZX0008, 2019YFG0533 and 2020YFS0057). (Corresponding author: Xing Xu).

Y. Zhou, X. Xu, F. Shen, X. Zhu and H. T. Shen are with the Center for Future Multimedia and School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (Email: joeyf.z.y.wen@gmail.com; xing.xu@uestc.edu.cn; fumin.shen@gmail.com; seanzhuxf@gmail.com; shenhengtao@hotmail.com)

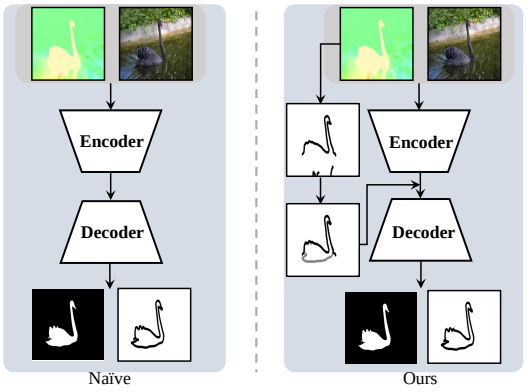


Fig. 2. The previous flow-based methods all consider directly using the optical flow. In this paper we proposed to refine the flow feature by connecting the flow edge.

and leverage the temporal motion information between frames [13]. Tokmakov *et al.* [14] proposed a model to map the optical flow to the segmentation mask. However, this method ignores the texture feature and results in under-segmentation problems. Cheng *et al.* [15] jointly predict the segmentation mask and the optical flow, thus, these two features can compensate for each other. Jain *et al.* [16] treat the RGB and flow feature equally, by fusing these two features to generate the segmentation mask. By viewing the flow as cues, Xiao *et al.* [17] proposed to deeply exploit motion cues for boosting video object segmentation and embed the FlowNet to their model. Li *et al.* [18] extract the appearance feature to estimate the segmentation mask and build the embedding graph based on the optical flow. Zhou *et al.* [1] proposed a motion attentive network to alleviate ambiguity in object appearance.

These flow-based methods all consider directly using the optical flow, however, due to the accuracy of optical flow, the noise may mislead models to separate the foreground objects incorrectly as shown in Fig. 1 Due to the motion of the camera, changing light intensity, and the occlusion, the boundary of optical flow is always ambiguous. Using this edge-blur optical flow may cause the blur segmentation mask as well as the false detection of segmentation boundaries. The previous methods are confronted with the problems of invalidation when objects appear in the middle of the video; hard implement when applied to online segmentation; ignoring the texture feature; the blur segmentation results; time-consuming; lacking the essential temporal and global feature.

Inspired by [19], [20], [21], the ambiguous flow boundaries are typically incomplete and do not align correctly with real boundaries of primary objects. On the contrary, a accurate flow boundaries can provide the outline feature of the moving objects (*i.e.* primary foreground objects) which are the key features of segmentation. It is important to refine the incomplete flow boundaries. During the refining procedure, the goal of in-painting the flow boundaries which is supervised by the real boundaries label, can guide the extraction of the real pure motion features. In this paper, we proposed a novel Flow Edge based Motion-Attentive Network (FEM-Net) to fully overcome the shortcomings of existing approaches. Specifically, as shown in Fig. 2, to effectively leverage the optical flow, the FEM-Net contains two stages: 1) The first is the flow edge connect stage, which contains the Flow Edge Connect module (FEC). In the FEC module, motion and appearance features are utilized to connect the weak flow edge, which was detected by the optical flow. 2) The second is the edge-based (*i.e.* , boundary aware) object segmentation mask synthesize module. In this stage, the synthesis strong connected flow edge and the motion-attentive appearance features are jointly used to generate the segmentation mask of foreground objects.

The mainly contributions of this paper are summarized as:

- We proposed a novel *Flow Edge-based Motion-Attentive Network* (FEM-Net) which contains two stages: the flow edge connect stage and the edge-based (*i.e.* , boundary aware) object segmentation mask synthesize module.
- We firstly use the appearance feature to further complete edges of flow fields, then the completing motion features are utilized to guide the model focus on the moving foreground primary objects, instead of using the raw motion and appearance features.
- Extensive experiments on two benchmark datasets demonstrate that the proposed FEM-Net compares favorably against the state-of-the-art methods.

II. Related Work

A. Video Object Segmentation

The video object segmentation can be divided into two types, the first is Unsupervised Video Object Segmentation (UVOS) which separates the foreground objects without any annotation. Another is the Supervised Video Object Segmentation (SVOS) in which the annotation of the first frame is given.

1) Supervised Video Object Segmentation: In SVOS task, the annotation of the first frame is given. Numerous algorithms view the SVOS as one-shot video object segmentation problem [22], [23]. They first train their models on the training dataset, then, during the test phase, according to the first given frame, they fine-tune their model. Some data augmentation methods are important in these approaches. The fine-tuning operation for these methods is time-consuming, therefore these approaches are not practical. Many studies proposed the mask propagation methods [24], [25] to predict mask based on the previous mask. In these methods, the previous mask and current frame are fed into the network. Since these approaches assume that no drastic changes between consecutive frames, the previous mask can be transformed to the current mask in a learnable way. Therefore, these methods are invalid when some objects appear in the middle of the video. Other methods aim to refine or modify the previous mask [26]. In addition, objects tracking methods are utilized to propagate the annotation of the first frame [27], [28], [29]. However, these methods are limited by the occlusion, motion of the camera, changing light intensity. Recently many researchers use the temporal cycle consistency [30], [25], [31] to predict the segmentation mask. In the case of online segmentation, these methods which require both the previous and future frame is hard to implement.

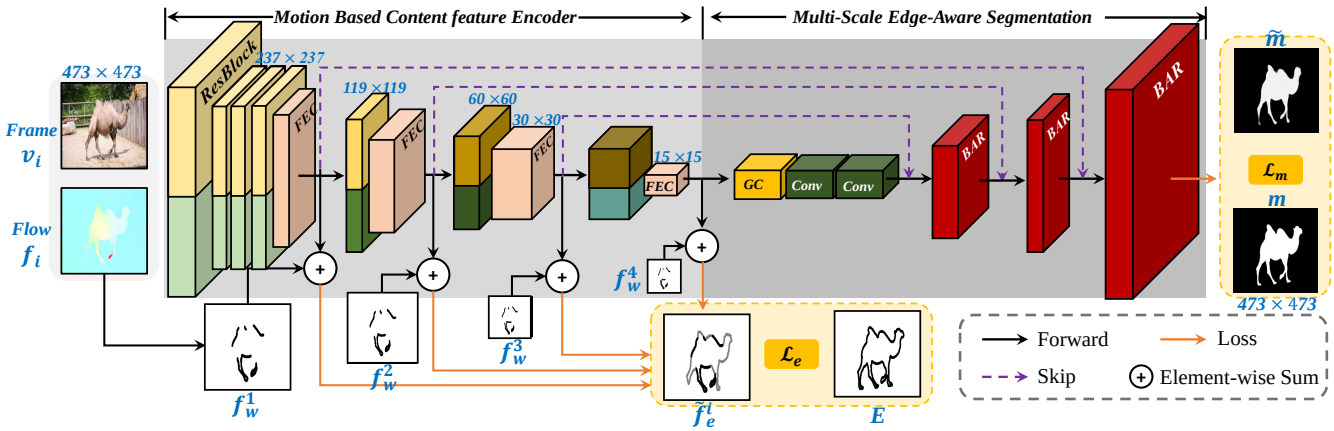


Fig. 3. The overview of FEM-Net. Given the input frame v_i, its correspond optical flow f_i and weak flow edge f'_w which computed by the candy algorithm. FEM-Net extracts the motion-attentive feature in the motion-based content feature encoder where four FEC modules are embedded. Then, the multi-scale edge-aware segmentation decoder computes the segmentation mask $\tilde{m}$ by 3 BAR modules.

2) Unsupervised Video Object Segmentation: The early UVOS methods take advantage of the handcrafted motion features to detect the primary objects [3], [32], [33], or utilize the appearance, shape to generate the region proposals [4], [34], [5]. With the development of deep learning, many researchers use deep neural network to solve this task in an end-to-end way [6], [35]. The ConvLSTM [9], [36] is also used to enhance the temporal information and produce the accurate segmentation mask. Recently, more researchers design the graph neural network to represent the relation between arbitrary frame pairs [37]. Detection-based methods [38], [7] first detect the bounding box of the object then, fed the target region and the global feature into the segmentation network. However, these methods are time-consuming and lack the essential temporal feature. It is hard to distinguish the primary objects within a single frame, even for the human. Lu *et al.* [2] proposed to determine the primary objects by referring more frames. Using the co-attention module to encode the correlation of video frames, the model can discover the foreground objects. This method needs an elaborate train and test pipeline, therefore, it's hard to apply to practical applications. These RGB-based models are always confronted with the problems of lacking global and temporal features, while the flow-based model can easily extract motion features from the optical flow.

B. Temporal Feature Complement Based on Optical Flow

Optical flow is a fundamental problem in computer vision. Since Horn *et al.* [39] first presented a method for finding the optical flow pattern, several researchers try to solve this problem by directly using the end-to-end fully convolutional network [40], [41]. The optical flow is wildly used to represent the motion and temporal features of consecutive frames [30], [42]. The inherent correlation information between frames, which is important to detect the primary objects, can be represented by the optical flow. Therefore, exploiting the flow information then mapping the flow to the segmentation mask is utilized in many flow based approaches [14], [15], [16], [17]. In order to produce the temporal consistent segmentation results, Liu *et al.* [43] leveraged the temporal consistent loss. Recently, Zhou *et al.* [1] proposed a motion-attentive transition module to jointly use the appearance and motion feature. However, because of the inaccuracy of optical flow, directly using the optical flow is unable to produce the sharp segmentation edges and thus have difficulties separating the objects from the background. The boundary of flow is always ambiguous, thus, some background regions may be segmented out incorrectly. To address these problems above, we designed the Flow Edge Connect module (FEC) to complement the information of optical flow. Instead of only using the optical flow, the connected flow edges are also fed into the segmentation network to guide our model comprehensively capture the moving details of primary objects.

C. Attention Mechanisms in Neural Networks

The attention mechanisms are widely used in deep neural network [44], [45], [46], [47], [48]. With attention, models can ignore unnecessary parts and only focus on the important regions, thus saving the computing cost and using the data more effectively. Wang *et al.* [49] exploited the attention based on multi-frame to guide the model focus on the moving objects. Zhou *et al.* [1] proposed the motion-attentive module to produce motion attention based on the optical flow. Lu *et al.* [2] proposed the co-attention to capture the global feature and comprehensively use the inherent correlation between frames. Our method builds upon the motion-attentive mechanisms. Different from the previous approaches, we firstly use the appearance feature to further complete edges of flow fields, then the completing flow fields are utilized to guide the model focus on the moving foreground primary objects, instead of directly using the raw motion and appearance feature.

III. Proposed Method

A. Preliminary

1) Problem Formulation: We formulate the UVOS as a flow based encoder-decoder problem. Given a RGB video sequence $\boldsymbol{v} \equiv \{v_1, v_2, \ldots, v_T\} \in \mathbb{R}^{T\times W\times H\times 3}$, we firstly estimate its corresponded forward optical flow $\tilde{\boldsymbol{f}} \equiv$

$\{\tilde{f}_1, \tilde{f}_2, \ldots, \tilde{f}_{T-1}\} \in \mathbb{R}^{T\times W\times H\times 2}$ and transfer it into RGB domain $\boldsymbol{f} \equiv \{f_1, f_2, \ldots, f_{T-1}\} \in \mathbb{R}^{T\times W\times H\times 3}$. The first five convolutional blocks of ResNet [50] are utilized to extract the feature of $\boldsymbol{v}$ and $\boldsymbol{f}$. The attention-based encoder $E(\cdot)$ is designed here to enhance the extracted feature complementarily. Then, using the edge detection algorithm to compute the edge of $\boldsymbol{f}$. After we get the weak flow edge $\boldsymbol{f_w} \in \mathbb{R}^{T\times W\times H\times 1}$, the FEC module is designed to complete the boundary of $\boldsymbol{f_w}$. Based on the strong flow edge $\boldsymbol{f_s}$, we learn a decoder $D(\cdot)$ to generate the segmentation mask $\tilde{m} \in \mathbb{R}^{T\times W\times H\times 1}$. This procession could be mathematically expressed as: $\tilde{m} = D(E(\boldsymbol{v}, \boldsymbol{f}), \boldsymbol{f_s})$.

2) Network Architecture: The framework of the proposed FEM-Net is illustrated in Fig. 3. For any frame $v_i \in \mathbb{R}^{W\times H\times 3}$ and its correspond optical flow $f_i \in \mathbb{R}^{W\times H\times 3}$, our network aims to separate the primary objects in v_i based on f_i. Our proposed FEM-Net can be divided into two parts: the motion-based content feature encoder; and the multi-scale edge-aware segmentation decoder. The v_i and f_i are fed into two independent Residual Block based layers to extract the appearance $\mathcal{F}_a$ and motion feature $\mathcal{F}_m$, respectively. In consideration of the time cost and the accuracy, the candy algorithm is used to process the f_i to get the flow edge f_w. The multi-scale structure is designed to extract $\mathcal{F}_a$ and $\mathcal{F}_m$ more effectively. Each down-sample convolutional layer is followed by a FEC module, each FEC module has four FEC blocks. The FEC module takes the $\mathcal{F}_a$ and $\mathcal{F}_m$ as input to enhance the down-sampled features and connect weak edge in f_w simultaneously. After computing and completing features of each scale, we fed these features to the decoder which includes global convolutional layers and 3 Boundary-Aware Refinement (BAR) modules to obtain the final mask output.

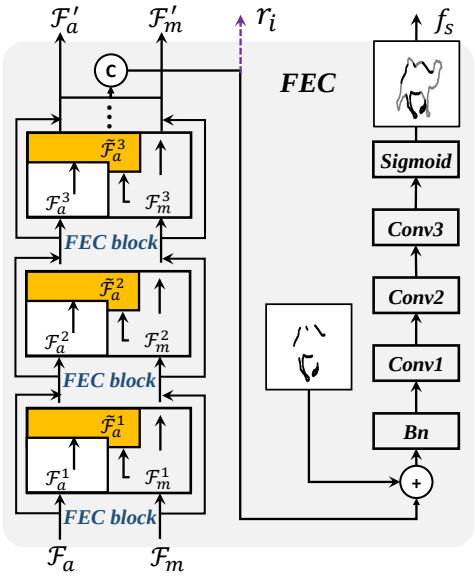


Fig. 4. The computational graph of the Flow Edge Connect Module with FEC blocks. In practice, the FEC module is composed by 5 FEC blocks, each blocks extract and refine the motion and appearance feature. The final concatenated feature is used to connect the weak flow edge.

B. Motion-Based Content Feature Encoder in FEM-Net

Since previous approaches directly using the encoder feature to produce the segmentation mask, these methods do not have a global view or rich comprehensive information about the frames. It's hard to distinguish the moving foreground objects within a single frame. Therefore we propose to use the motion feature extracted from the optical flow, to guide our model focus on the moving objects (*i.e.* , primary objects). In the FEM-Net, the motion feature can boost our model to focus on the important regions, meanwhile, the appearance feature can be utilized to refine the motion feature. This is win-win cooperation to use these two kinds of features in a complementary way.

As illustrated in the Fig. 3, FEM-Net aims to use the appearance feature to refine the flow feature and meanwhile use the flow feature to guide the encoder to extract more features of the moving primary objects. The motion-based content feature encoder contains three steps, the first is feature extract, in this paper we use ResNet101 [50] as backbone; the second is the motion-attentive transition; the last is flow edge generation.

1) Flow Edge Connect Module: After extracting the feature of frame $\mathcal{F}_a \in \mathbb{R}^{W\times H\times C}$ and its correspond RGB optical flow $\mathcal{F}_m \in \mathbb{R}^{W\times H\times C}$, based on [1], we further advance the motion-attentive transition module by introducing the flow edge connection module. As shown in Fig. 4 left part, taking $\mathcal{F}_a$ and $\mathcal{F}_m$ as input, the first step of each FEC block is to compute the attention of appearance and motion features respectively. According to [44], [51], [1], we take the same strategy to mine the intra-class correlations within $\mathcal{F}_a$ and $\mathcal{F}_m$. Specially, we first project the input features to low dimension and use softmax to compute the attention:

$$S_a^{(l)} = softmax(P(\mathcal{F}_a^{(l-1)})), \tag{1}$$

where $S_a^{(l)} \in \mathbb{R}^{W\times H\times 1}$ denotes the attention of $\mathcal{F}_a^{(l-1)}$; $P(\cdot)$ is the project function. The superscript l denotes the l^{th} FEC block. Specifically, the $\mathcal{F}_a$ is equal $\mathcal{F}_a^{(0)}$. The attention of $\mathcal{F}_m^{(l-1)}$ can be computed in the same way. Then the attentive feature can be computed by:

$$\mathcal{F}_a^i = \mathcal{F}_a^{(l-1)} \odot S_a^{(l)}, \mathcal{F}_m^i = \mathcal{F}_m^{(l-1)} \odot S_m^{(l)}, \tag{2}$$

where $\mathcal{F}_a^i, \mathcal{F}_m^i$ is the inner attentive feature product of i^{th} FEC block. The $\odot$ denotes element-wise Hadamard product, during the computation phase $S_a^{(l)}$ will broadcast to the same channel number as $\mathcal{F}_a^{(l-1)}$.

Only using attentive features can not effectively boost the model to focus on the important regions since these two features are from different domains and work independently. Inspired by [2], the co-attention module is designed here to transfer the flow feature to the motion attention, and use the motion attention to refine the appearance feature.

To mine the correlation between v_i and f_i in their own feature embedding domain space, we compute the affinity matrix $\boldsymbol{A} \in \mathbb{R}^{WH\times WH}$ between $\mathcal{F}_a^i$ and $\mathcal{F}_m^i$.

$$\boldsymbol{A} = (\mathcal{F}_m^i)^T W \mathcal{F}_a^i, \tag{3}$$

where the $W \in \mathbb{R}^{C\times C}$ is the weight matrix, $\mathcal{F}_m^i$ and $\mathcal{F}_a^i$ are transfer to matrix shape $\mathbb{R}^{C\times WH}$. Each row $c \in \{1,\ldots,C\}$ represent the feature value of channel c, each column p,q represent the feature value at spatial position $p,q \in \{1,\ldots,WH\}$. Finally, each value of $\boldsymbol{A}_{pq}$ represents the correlation between $p^{th} \in \{1,\ldots,WH\}$ feature in $\mathcal{F}_m^i$ and $q^{th} \in \{1,\ldots,WH\}$ feature in $\mathcal{F}_a^i$. Since the W contains a huge number of parameters, the computational cost is unsatisfied and the risk of over-fitting is increasing. If we use formula 3, we need to flatten the $\mathcal{F}_m^i$ and $\mathcal{F}_a^i$ to $\mathbb{R}^{C\times HW}$ then calculate the W. The cost is equal to $HW \times HW \times C \times C$. Therefore, we use the same strategy as Zhou *et al.* [1]: the M is factorized into two reductive matrices $P \in \mathbb{R}^{C\times \frac{C}{d}}$ and $Q \in \mathbb{R}^{C\times \frac{C}{d}}$. Then, the affinity matrix $\boldsymbol{A}$ can be computed by:

$$\boldsymbol{A} = (\mathcal{F}_m^i)^T P Q^T \mathcal{F}_a^i = (P^T(\mathcal{F}_m^i))^T (Q^T \mathcal{F}_a^i). \quad (4)$$

The reduction operation can be viewed as the channel-wise feature refinement. After obtaining $\boldsymbol{A}$ we normalize the $\boldsymbol{A}$ row-wise with a softmax function to get $\boldsymbol{A^r}$. In practice, the weights of 1×1 convolution are viewed as P and Q, respectively. The $\boldsymbol{A}$ can be computed based on the $\mathcal{F}_m^i$ and $\mathcal{F}_a^i$. The cost is equal to $\frac{2HWC^2}{d} + \frac{H^2W^2C^2}{d^2}$. We can find that the later factorized method dose save a lot of computing costs. Then, the motion guided appearance feature $\tilde{\mathcal{F}}_a^i \in \mathbb{R}^{C\times W\times H}$ is given by:

$$\tilde{\mathcal{F}}_a^i = \mathcal{F}_a^i \boldsymbol{A^r}. \quad (5)$$

In order to maintain the original information and avoid the gradient vanishing problem, we skip connect the input and output of each FEC block. Fives FEC blocks are cascaded serially to mine the deep correlation between the input. The number of FEC blocks is chosen according to the MATNet [1] where 5 MAT layers are demonstrated to be the optimal number for these motion and appearance feature attention-based fusion module.

This procession could be mathematically expressed as:

$$\mathcal{F}_a^{(l)} = \tilde{\mathcal{F}}_a^i + \mathcal{F}_a^{(l-1)}, \mathcal{F}_m^{(l)} = \mathcal{F}_m^i + \mathcal{F}_m^{(l-1)}, \quad (6)$$

where $\mathcal{F}_a$ and $\mathcal{F}_m$ are the input features of first FEC block, $\mathcal{F}_a^{'}$ and $\mathcal{F}_m^{'}$ are the final output of last FEC block, $\mathcal{F}_a^{(l)}$ and $\mathcal{F}_m^{(l)}$ denote the output of l^{th} FEC block. Specifically, $l \in \{1,2,3,4,5\}$.

As shown in Fig. 4 right part, $\mathcal{F}_a^{'}$ and $\mathcal{F}_m^{'}$ are concatenated to obtain the motion-attentive feature $\mathcal{F}_{ma}$. Then, $\mathcal{F}_{ma}$ is used to connect the weak edge in f_w which was detected by the candy algorithm. Three convolutional layers are designed to extract the edge feature based on the enhanced motion and appearance feature. Then the sigmoid function is used to generate the strong edge map f_s.

2) Multi-Scale Feature Extraction: Since various difficult scenarios such as illumination changes, background or camera motion can affect our model to learn the inherent features of the primary objects and background, we extract the features in a multi-scale manner, which is shown in Fig. 3. Each down-sample convolutional layer is followed by a FEC module, after the FEC module enhancing the features which are output by the previous ResBlock, the outputs $\mathcal{F}_a^{'}$ and $\mathcal{F}_m^{'}$, respectively, are recursively fed into the next ResBlock as input. This procession could be mathematically expressed as:

$$\mathcal{F}_a^{\{L\}}, \mathcal{F}_m^{\{L\}} = \Phi_{FEC}^{\{L\}}(D_{sa}^{\{L\}}(\mathcal{F}_a^{\{L-1\}}), D_{sm}^{\{L\}}(\mathcal{F}_m^{\{L-1\}})), \quad (7)$$

where $D_{sa}(\cdot)$ ($D_{sm}(\cdot)$) denotes the down-sampling and extracting ResBlock of appearance (motion) feature, $\Phi_{FEC}(\cdot)$ denotes the FEC computation, L denotes L^{th} down-sample or FEC operation. Specifically, shown in Fig. 4, $D_s^{\{L\}}(\mathcal{F}_a^{\{L-1\}})$ is equal to the input of L^{th} FEC module (*i.e.* , $\mathcal{F}_a$), $\mathcal{F}_a^{\{L\}}$ is equal to the output of L^{th} FEC module (*i.e.* , $\mathcal{F}_a^{'}$). The number of L is equal to the down-sample times which is 4, therefore, $L \in \{1,2,3,4\}$.

C. Multi-Scale Edge-Aware Segmentation in FEM-Net

The decoder takes full use of the multi-scale feature and edge information to separate the primary foreground objects and background. The four multi-scale motion-attentive features extracted by the FEC modules are utilized to carry out segmentation. Although the previous FEC modules have considered extracting the global feature, a global convolution layer is still designed here to captures inherent information. Since the features from low-level can be used to restore the details of the object (shape and boundary), the skip connection is also embedded in the decoder.

After getting the motion-attentive feature $\mathcal{F}_{ma}$, we concatenated it with the strong edge flow map f_s to get the edge-aware motion-attentive feature $\mathcal{F}_{ema}$. The $\mathcal{F}_{ema}$ not only focuses on the primary moving objects which guided by the affinity matrix $\boldsymbol{A}$, but also includes rich detail information about the objects provided by the f_s. In the decoder network, each deconvolutional layer takes $\mathcal{F}_{ema}$ and the output of the previous layer as input to conduct the segmentation. Inspired by [1], the Boundary-Aware Refinement (BAR) module is embedded in FEM-Net because it can benefit the decoder to enlarge the receptive filed and capture more details for segmentation. This recursive module is shown in Fig. 5. The

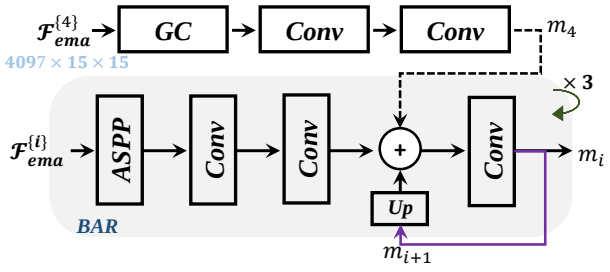


Fig. 5. The architecture of Boundary-Aware Refinement (BAR) module, GC denotes the global convolutional layer, Conv denotes convolutional layer, ASPP is Atrous Spatial Pyramid Pooling module [52]. This BAR will iterate 3 times, the initial mask is m_4. The output m_i of the previous BAR will be up-sampled and fed into the next BAR (Purple line).

first row is the basic step of decoder network, this global process is applied to the last output $\mathcal{F}_{ema}^{\{4\}} \in \mathbb{R}^{4097\times 15\times 15}$ of encoder network. The output $m_4 \in \mathbb{R}^{1\times 15\times 15}$ is unable to capture the rich details in the original frame, therefore, a multi-scale strategy is used here to carry out the segmentation iteratively.

The initial raw mask m_4 is viewed as the input of the first BAR module. We treat the edge-aware motion-attentive feature $\mathcal{F}_{ema}$ as the complementary information to refine our segmentation mask. We also use Atrous Spatial Pyramid Pooling (ASPP) [52] to capture the feature context at multiple scales. After propagating through the ASPP, the raw mask m_{i+1} from the previous stage is added to it linearly and followed by several convolutional layers to refine the mixture features. Then, the output mask m_i could be viewed as the input of the next BAR module. This iteration will not stop until the m_i is up-sampled to the target scale. Finally, the segmentation mask can be computed by the extra 1×1 convolution layer and Sigmoid function based on the last output $m_1 \in \mathbb{R}^{1\times 119\times 119}$.

D. Objective Function

The FEM-Net is trained on the whole training dataset frame by frame. The loss function can be divided into two aspects, the first is mask loss; the second is boundary loss. The PWC-Net is used here to estimate the optical flow between two consecutive frames due to its accuracy and speed. Then, the boundary map is calculated by the ground-truth mask label. We simply assume the boundary pixel when its four direction pixel values are XOR. Similar to [1], we also introduce the heuristic method to automatically detect hard negative pixels. Specifically, the off-the-shelf HED model [53] is used to predict the $P \in \mathbb{R}^{1\times 473\times 473}$ probability of a pixel P_e whether belongs to an edge pixel. If the P_e is higher than the set threshold, this pixel is regarded as the hard negative pixel. Then in the loss function, its weight is set as $w_i = 1 + P_e$, otherwise, $w_i = 1$. Both the mask and boundary loss are binary cross entropy loss. The overall objective function of our FEM-Net is to minimize the integration of the above these two loss terms by:

$$\mathcal{L} = \mathcal{L}_{mask}(\tilde{m}, m) + \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{bdry}(\tilde{f}_e^i, E), \tag{8}$$

where the $\tilde{m} \in \{0,1\}^{W\times H}$ is the output segmentation mask, the N is equal to 4 which is the number of FEC layer, the $\tilde{f}_e^i \in \{0,1\}^{W\times H}$ is the edge map of i^th scale, the m and E is the ground-truth of mask and boundary, respectively. The $\mathcal{L}_{mask}$ and $\mathcal{L}_{bdry}(\tilde{f}_e^i, E)$ is the binary cross entropy loss:

$$\begin{aligned}\mathcal{L}(O, Y) = -\sum_{i}^{M} w_i(Y_i \log(O_i) \\ +(1 - Y_i)\log(1 - O_i)),\end{aligned} \tag{9}$$

where the O represents the output of our FEM-Net, the Y denotes the ground-truth of mask or boundary, respectively, the w_i is the weight value, the M equal to the number of pixel in each frame. The multi-scale training manner is used here to extract comprehensive structure features from coarse to fine. Each scaled edge map is output by a 1×1 convolution then activated by the sigmoid function. Finally, all the outputs (N edge maps) are bilinear interpolated to 473×473 to compute the loss. The SGD optimizer is adopted to train our network with the learning rate is initially set to 1e-4 for the encoder and 1e-3 for the decoder.

IV. Experiment

A. Experiment Setup

We compare FEM-Net with the state-of-the-art methods on the DAVIS-16 and FBMS datasets.

DAVIS-16 is a densely annotated video segmentation dataset. It includes 50 high quality, full HD video sequences (30 videos for training and 20 for testing), each video is densely annotated frame-by-frame. For a fair comparison, the standard evaluation protocol from [54] is adopted here, *i.e.* , region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$, and the time stability $\mathcal{T}$.

FBMS [55], [56] is a sparsely annotated video segmentation dataset which contains 59 video sequences. Different from DAVIS, only some of the frames are annotated (about 720 frames). We evaluate our proposed method on the testing split which is comprised of 30 sequences. The region similarity $\mathcal{J}$ is chosen for the quantitative evaluation.

B. Implementation Details

The FEM-Net is trained in an end-to-end manner. The backbone of FEM-Net is ResNet101 [50]. We embed four FEC modules into FEM-Net, for each training frame of size $473 \times 473 \times 3$, the frame is down-sample to the size of $\{119, 60, 30, 15\}$. The output size of our FEC-Net is $119 \times 119 \times 3$, for the segmentation task, we use the bilinear interpolate algorithm to restore the original size of the input. The SGD optimizer is adopted to train our network with the learning rate is initially set to 1e-4 for the encoder and 1e-3 for the decoder. The batch size is set to 2, the momentum is set to 0.9 and the weight decay is 1e-5. Some common data augmentation methods are used during training our network *e.g.* horizontal flip and rotations. We train the FEM-Net on the training video in DAVIS-16 [54], which includes about 2K frames. The optical flow is estimated by the PWC-Net [61] due to the accuracy and time cost. All the experiments are conducted on a single Nvidia RTX 2080Ti GPU card which is on par with the previous work [2], [1] under the same experimental setting.

C. Evaluation Metrics

The metrics used in this paper are the same with [54], which are composed of the region similarity, contour accuracy, temporal in-stability.

Region Similarity $\mathcal{J}$ is defined as the intersection-over-union of the predict segmentation and the ground-truth mask. This metric measures the region-based segmentation similarity by counting the number of the mislabeled pixels. Given an output segmentation M and the ground-truth mask G, the Region Similarity can be computed by $\mathcal{J} = \frac{|M\cap G|}{|M\cup G|}$.

Contour Accuracy $\mathcal{F}$. In order to measure the accuracy of the counters, we compute the boundary-based precision P_b and recall R_b between the boundary points of $b(M)$ and $b(G)$ via a bipartite graph matching. The F-measure $\mathcal{F}$ is used here to quantify the trade-off between the two, which is defined as $\mathcal{F} = \frac{2P_bR_b}{P_b+R_b}$. The morphology operators are used to approximate the bipartite matching.

Fig. 6. Qualitative results on DAVIS and FBMS. From top to bottom: *camel*, *dance-twirl*, *motocross-jump* from DAVIS-16, *marple7* and *rabbits02* from FBMS. We can find that FEM-Net can tackle the similar background foreground segmentation, motion blur, large occlusion, small objects.

TABLE I
QUANTITATIVE RESULTS ON THE TEST SET OF DAVIS-16 [54], USING THE REGION SIMILARITY $\mathcal{J}$, BOUNDARY ACCURACY $\mathcal{F}$ AND TIME STABILITY $\mathcal{T}$. WE ALSO REPORT THE RECALL AND THE DECAY PERFORMANCE OVER TIME FOR BOTH $\mathcal{J}$ AND $\mathcal{F}$. THE BEST SCORES ARE MARKED IN **BOLD**.

	Method	KEY [34]	MSG [3]	NLC [57]	CUT [15]	FST [42]	SFL [58]	LMP [14]	FSEG [16]	LVO [59]	ARP [60]	PDB [36]	MATNet [1]	COSNet [2]	**FEM-Net**
$\mathcal{J}$	Mean ↑	49.8	53.3	55.1	55.2	55.8	67.4	70.0	70.7	75.9	76.2	77.2	78.0	**80.2**	79.9
	Recall ↑	59.1	61.6	55.8	57.5	64.9	81.4	85.0	83.0	89.1	91.1	90.1	92.1	91.9	**93.9**
	Decay ↓	14.1	2.4	12.6	2.2	**0.0**	6.2	1.3	1.5	**0.0**	7.0	0.9	4.3	**3.6**	4.4
$\mathcal{F}$	Mean ↑	42.7	50.8	52.3	55.2	51.1	66.7	65.9	65.3	72.1	70.6	74.5	74.8	**77.4**	76.9
	Recall ↑	37.5	60.0	61.0	51.9	51.6	77.1	79.2	73.8	83.4	83.5	84.4	87.5	87.4	**88.3**
	Decay ↓	10.6	5.1	11.4	3.4	2.9	5.1	2.5	1.8	1.3	7.9	**-0.2**	2.9	4.3	2.4
$\mathcal{T}$	Mean ↓	26.9	30.2	42.5	27.7	36.6	28.2	57.2	32.8	26.5	39.3	29.1	14.7	13.4	13.7

Temporal In-Stability $\mathcal{T}$. Different from the image segmentation, the temporal coherence is a relevant aspect in video object segmentation. The mask M_t of frame t is transformed into polygons representing its contours $P(M_t)$. Each point $p_t^i \in P(M_t)$ is described by using the Shape Context Descriptor (SCD) [62]. Similar to [54], the mean SCD matching cost between p_t^i and p_{t+1}^i of per matched point is used as the measure of temporal stability $\mathcal{T}$.

Given a metric $\mathcal{M} \in \{\mathcal{J}, \mathcal{F}, \mathcal{T}\}$ and $\boldsymbol{V}$ the dataset of video sequences V_i, let $\overline{\mathcal{M}}(V_i)$ be the error measure average on V_i, we consider three different statistics: (1) the mean of $\mathcal{M}_m = \frac{1}{N}\sum_{V_i \in \boldsymbol{V}} \overline{\mathcal{M}}(V_i)$ denotes the average performance of the method, where N equal to the number of video in test dataset $N = |\boldsymbol{V}|$; (2) the decay value quantifies the performance loss over time, the output should be stable throughout the video. The decay value is defined as $\mathcal{M}_d = \frac{1}{N}\sum_{Q_i \in \boldsymbol{V}} \overline{\mathcal{M}}(Q_i^1) - \overline{\mathcal{M}}(Q_i^4)$ where the $Q_i = \{Q_i^1 \dots Q_i^4\}$ is a partition of V_i in quartiles; (3) the recall value denotes the number of metric value which higher than a threshold. This is defined as $\mathcal{M}_r = \frac{1}{N}\sum_{V_i \in \boldsymbol{V}} |\overline{\mathcal{M}}(V_i) > \tau|$, with $\tau = 0.5$ in our experiments.

TABLE II
ABLATION STUDY AND FURTHER ANALYSIS OF FEM-NET ON DAVIS-16 WITH SYMMETRIC FEC BLOCK ARCHITECTURE, NO FLOW EDGE COMPLEMENT, FUSION STRATEGIES.

Network Variant	Mean $\mathcal{J}$ ↑	$\Delta\mathcal{J}$	Mean $\mathcal{F}$ ↑	$\Delta\mathcal{F}$
FEM-Net (Symmetric)	76.1	-3.8	73.2	-3.7
FEM-Net (No Flow Edge)	77.5	-2.4	75.0	-1.9
FEM-Net (Fusion Strategy)	73.6	-6.3	71.1	-5.8
FEM-Net	79.9	-	76.9	-

D. Ablation Study

In this section, we analyze the impacts of different components in FEM-Net.

1) Asymmetric Structure Based Feature Extraction: In each FEC block, intuitively, a symmetric design is plausible to achieve more effective performance. In other words, the appearance features may able to guide FEM-Net to refine the motion features. Specifically, instead of calculating the $F_m^{(l)}$ by Eq 6, the appearance attention is used to refine the motion feature cyclically. However during the experiments, as shown in Table II, we found that this symmetric design may lead our network to discard some important motion features and

results in poor performance. The Mean $\mathcal{J}$ drops about **3.8%**, and Mean $\mathcal{F}$ drops about **3.7%**, due to the symmetric FEC block making it more complex to extract the motion feature.

2) Effect of Flow-Edge: Different from previous approaches, the boundary information is guided by the weak flow edge computed by the optical flow. In this section, we study the effects of our flow edge connect (FEC) block by comparing the full model to that of the same architecture without flow-edge complement. As shown in Table II, the flow-edge complement improves the performance of FEM-Net, in addition, this complement also accelerates the converge time and enhances the training stability of our network.

TABLE III
REGION SIMILARITY COMPARISONS WITH DIFFERENT NUMBERS OF FEC BLOCKS CASCADED IN EACH FEC MODULE ON DAVIS-16.

FEC Block	N = 6	N = 5	N = 4	N = 3
Mean $\mathcal{J}$	79.36	**79.95**	78.80	78.30

3) FEC Blocks: For the number of FEC blocks, we not only evaluate the performance of FEM-Net with the different number of FEC blocks shown in Table III, but also visualized the output of each FEC where 5 FEC blocks are cascaded, shown in Fig. 7 row 2. As the number of FEC blocks increases, the performance of FEC-Net improves. When the number of FEC blocks exceeds 5, the performance of the model starts to degrade. Based on this observation, we regard $N = 5$ as the optimal number of FEC blocks in the FEC module.

In order to demonstrate the robustness of FEM-Net, we test our model by two kinds of optical flow: 1) the optical flow estimated by the FlowNet2 [41]. 2) the optical flow estimated by the PWC-Net [61]. According to Sun *et al.* , the optical product by PWC-Net is better than that of FlowNet2. We use these two kinds of flow to carry out the segmentation and found the FEM-Net can against the optical flow inaccuracy. As shown in Table IV, we compare FEM-Net with MATNet which directly uses the raw optical flow. The mean $\mathcal{J}_{pwc}$ denotes the mean region similarity for using the optical flow which is estimated by PWC-Net, while mean $\mathcal{J}_{fn2}$ denotes for FlowNet2. The performance gap between $\mathcal{J}_{pwc}$ and $\mathcal{J}_{fn2}$ is bigger than that of any FEM-Net with different FEC blocks.

Noted that in the FEM-Net with 3 FEC blocks, the $\mathcal{J}_{fn2}$ is even higher than $\mathcal{J}_{pwc}$. From Table IV we can find that, the $\mathcal{J}_{fn2}$ of N FEC block based FEM-Net is better than $\mathcal{J}_{pwc}$ of $N-1$ FEC block based. This demonstrated the improvement of each FEC block.

TABLE IV
ROBUST ANALYSIS OF FEM-NET ON DAVIS-16 WITH OPTICAL FLOW ESTIMATED BY PWC-NET AND FLOWNET2, RESPECTIVELY.

Network Variant	Mean $\mathcal{J}_{pwc}\uparrow$	Mean $\mathcal{J}_{fn2}\uparrow$	$\Delta\mathcal{J}$
MATNet	78.09	77.48	0.61
FEM-Net (5 FEC Blocks)	**79.95**	**79.41**	0.54
FEM-Net (4 FEC Blocks)	78.80	78.60	0.20
FEM-Net (3 FEC Blocks)	78.30	78.31	-0.01

4) Effect of BARs: In order to show the improving degree of each BAR, we visualize the output of each BAR. In practice, we use the 1×1 convolution and the Sigmoid function to get each mask and boundary. BARs take in different scales input features to carry out the segmentation, all the results are interpolated into original size: 473×473. With the number of BAR increasing, the segmentation becomes more accurate. In Fig. 7 from left to right are the outputs of the first to fourth BAR. In addition, we also visualized the FECs synthesis process to prove the effectiveness of each BAR. Specifically, we can find that the early BAR concentrates on the object's shape and position, while the later BARs focus on the boundaries and details. The lower BARs are only aware of the low-level features (the position, shape area), in these stages BARs are unable to predict the boundary. While the higher BARs can synthesis the details based on the high-level features. From the outputs of FECs, we can find that lower FECs can not synthesis the clear boundary map because these FECs do not extract the high-level information. The later FECs focus on the high-level features, thus the boundary maps become clear.

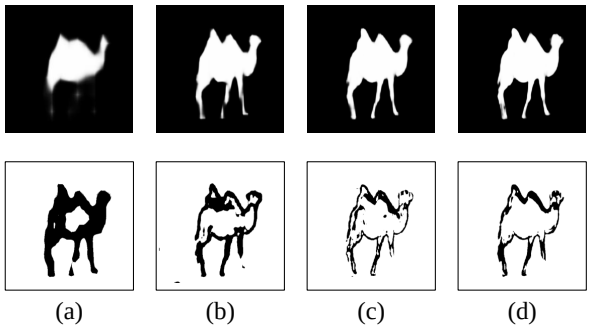


Fig. 7. The visualization for the output of each BAR (first row). From left to right, (b)-(d) is the first to last BAR's output; (a) is synthesized directly by the feature. In order to make it more comprehensive, we also visualized FECs synthesis process (second row).

E. Compare with Existing Methods

We compare our proposed FEM-Net with the state-of-the-arts methods in the public benchmarks of DAVIS-16 and FBMS. We measure the results provided by the respective authors under fair and identical conditions to obtain their evaluation results.

1) Evaluation on DAVIS-16: The previous methods can be divided into two classes: (1) the RGB-based methods segment the objects through the RGB frame. Many researchers [34], [57] use the salient object detection to extract the spatial saliency features. The representative method NLC [57] has achieved **55.1%** on Mean $\mathcal{J}$. These methods lack temporal information and thus obtain poor results. Since the LSTM is competent to extract the temporal feature, based on the spatial saliency feature, some researchers [36], [9] proposed to use the ConvLSTM to jointly extract the spatial and temporal feature. The PDB has made a great improvement and achieved **77.2%** on Mean $\mathcal{J}$. With the co-attention of different refer frames,

the COSNet [2] learns to capture the global correlations and scene context thus, outperforms all the RGB-based methods across most metrics and reaches **80.2%** **77.4%** of Mean $\mathcal{J}$ and $\mathcal{F}$ of, respectively. (2) the Flow-based methods use the optical flow as a cue to segment objects. The early flow-based methods [15], [14] aim to learn a mapping function which maps the optical flow to the segmentation mask. The representative method LMP has achieved **70.0%** over Mean $\mathcal{J}$. However these methods ignore the importance of appearance feature, therefore, other researchers proposed to leverage the motion and appearance feature jointly [16], [1]. Recently the proposed MATNet has achieved favorable performance against other approaches, which gets **78.0%** and **74.8%** on Mean $\mathcal{J}$ and Mean $\mathcal{F}$.

As shown in Table I, our model outperforms all the flow-based methods and achieve the results on DAVIS with **79.9%** over Mean $\mathcal{J}$, however, slightly lower than COSNet the RGB-based method. The RGB-based methods are incompetent to extract temporal features unless they use more frames simultaneously. Although the RGB-based COSNet has achieved the best result, due to the co-attention is important in these RGB-based methods, the training procedure is more complex than that of flow-based methods, and could not be applied to the online segmentation task. In addition, these methods depend on the reference frames, therefore, the extend-ability to the sparse dataset is defective. The accuracy will drop sharply when the reference frames are insufficient. We can find that FEM-Net has achieved the best results **93.9%** and **88.3%** in terms of recall value of $\mathcal{J}$ and $\mathcal{F}$ respectively. Many researchers trained their models with extra data [1], [49] and fine-tune on the DAVIS dataset. For a fair comparison, we trained their models on the DAVIS-16 by using the officially published code. The FEM-Net is trained on the DAVIS-16 without any other extra data.

In Table I, many learning-based state-of-the-art UVOS methods [1], [14], [15], [57] leverage the optical flow to improve the accuracy of segmentation. Different from these methods, the edge details in the optical is fully used in our FEM-Net. The connected flow-edge map can provide considerable information for the network to perceive the edge of moving objects, and then focus to extract the features of the primary objects.

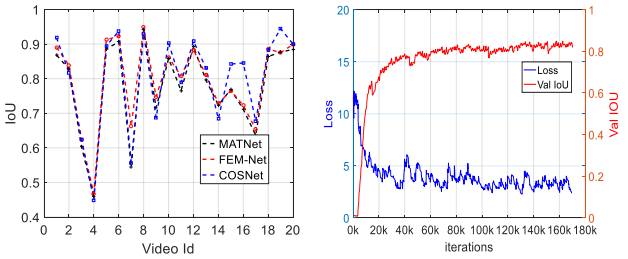
TABLE V
QUANTITATIVE PERFORMANCE ON THE TEST SEQUENCES OF FBMS [55] USING REGION SIMILARITY (MEAN $\mathcal{J}$).

Method	COSNet [2]	MATNet [1]	FSEG [16]	MSTP [63]	ARP [60]
Mean $\mathcal{J}$	75.6	76.1	68.4	60.8	59.8
Method	IET [64]	OBN [18]	PDB [36]	SFL [58]	**FEM-Net**
Mean $\mathcal{J}$	71.9	73.9	74.0	56.0	**78.5**

2) Evaluation on FBMS: We also conduct experiments on the FBMS datasets for completeness. Table V reports the detailed results on FBMS. The FEM-Net achieves the best results on FBMS with **78.5%** over Mean $\mathcal{J}$ which outperforms the second-best results, *i.e.* , MATNet [1], by **2.4%**. Because the correlation of the neighbor frame is low, these networks are unable to capture the motion or co-attentive feature. Some of the RGB-based methods [60], [2], [36] drop sharply when they apply to the sparse dataset. Specifically, for ARP drops from **76.2%** to **59.8%**; for COSNet drops from **80.2%** to **75.6%**; for PDB drops from **77.2%** to **74.0%**. From this aspect, we can find that the adaptability of FEM-Net outperforms other methods.

3) Qualitative Results: Fig. 6 shows some segmentation results for representative frames from two datasets. Even for the frames with the high similarity between the background and the foreground *e.g.* the *camel* sequence from DAVIS-16. The FEM-Net can accurately separate the primary objects form the background. The *dance-twirl* sequence also has many challenges for video object segmentation, specifically, due to the fast motion of her body, this video contains the occlusion, object deformation and motion blur problems. We can find that our proposed method can robustly tackle these problems with ease. The remaining rows are samples of FBMS dataset which show the segmentation results for the case of occlusion, fast-moving, and small objects, respectively. The flow-edge motion-attentive mechanism helps FEM-Net effectively focus on the primary objects and capture the background feature, then segment the objects from the cluttered background.

F. Further Analysis



(a) Validation IoU (b) Loss tendency

Fig. 8. Experimental results: (a) The comparison of mean IoU of each video sequence (from test dataset of DAVIS) with the state-of-the-art methods. The results of MATNet, FEM-Net, COSNet are shown in black, red, blue dotted line, respectively; (b) The training loss tendency (red line) and IoU (blue line) on the validation dataset of FEM-Net.

1) Flow-Edge Fusion Phase: As shown in Fig. 4, we fuse the flow edge with the motion-attentive feature extracted by 4 cascaded FEC blocks. In addition, we also conduct the experiments that fuse the flow edge information in each FEC block. The flow edge is used to refine the motion feature in each scale stage, specifically, we concatenate the weak flow edge with the motion feature $\mathcal{F}_m^{(l)}$ in each FEC block, and use the convolutional layer to transform it to the same channel number. As shown in Table II, the edge and the flow feature come from different domains, simply concatenate them will decrease the performance of our network. This observation shows the superiority of fusion in the motion-attentive feature embedding space.

2) IoU Various: As shown in Fig. 8 (a), compared with the state-of-the-art RGB-based segmentation method COSNet and the Flow-based MATNet. The flow-based methods fail to achieve the same performance as the RGB-based methods. Specifically, in *scooter-black* sequence shown in Fig. 9, the

white car which appears in the middle of the video are separated as the primary objects. Due to the motion of the camera, the white car looks like moving backward. Therefore, the flow-based network is misled. The RGB-based methods do not have such trouble, these methods only focus on the early objects. However, only focus on the early objects will ignore the real foreground primary objects which appear in the middle of the video, which is also shown in Fig. 9. The person who shows up in the middle of the video can be accurately segmented out by our FEM-Net, while it is difficult for the COSNet, because the co-attention of COSNet ignores the objects that appear mid-way.

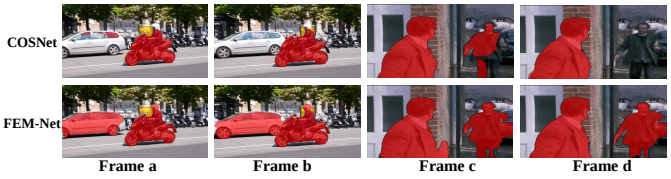


Fig. 9. The representative sample results of RGB-based and Flow-based methods. The first row is the results from COSNet, the second row is the results of FEM-Net. The white car is stationary but due to the motion of camera it looks like moving backward. The person near the car appears midway in the video.

3) Tendency of Training Loss: As shown in Fig. 8 (b), FEC block helps FEM-Net to converge and enhance the stability of training. The BCE loss is converged about 100k iterations and the IoU on the validation dataset fluctuates around 0.8. In addition, it takes about 25 hours to train the FEM-Net.

4) Runtime Evaluation: We evaluated the time consumption of the inference period. Data loading time is not counted, while the optical flow estimating time is only dependent on the flow net, here we use PWC-Net, this estimating time is also not counted. In addition, the edge detection cost is also excluded. Since all the pre-processing are done (optical flow, edge, HED), we only focus on the segmentation time of the FEM-Net. Specifically, in practice, in order to produce more satisfied results, we not only feed the original frame but also its flipped version into FEM-Net and regard the mean of the above two as the final output segmentation. We test the runtime of FEM-Net on 20 videos of DAVIS test dataset. We found that, for the original frame segmentation, the average speed of 20 test videos is about **16** fps; for flipped segmentation, the average speed of 20 test videos is about **7** fps (including the flipping time and average calculation time).

V. Conclusion

In this paper, we proposed the Flow Edge based Motion-Attentive Network (FEM-Net) to solve the unsupervised video object segmentation problem. Different from previous flow-based UVOS methods, a novel Flow Edge Connect (FEC) module is designed to refine flow feature by hallucinating edges of the ambiguous or missing region in the optical flow. In addition, the motion-attentive encoder is used to jointly learn the spatial and temporal features. Through the complete flow-edge and the motion-attentive feature, FEM-Net learns to focus on the primary objects and segment the objects from the cluttered background. The proposed method has achieved superior results on two public benchmarks. Extensive experimental results demonstrate that FEM-Net can effectively handle the occlusion, object deformation, and motion blur problems. The FEM-Net is a novel flow-based network which can be extended to many flow-based tasks such as action recognition and optical flow estimation.

References

[1] T. Zhou, J. Li, S. Wang, R. Tao, and J. Shen, "Matnet: Motion-attentive transition network for zero-shot video object segmentation," *IEEE Transactions on Image Processing*, vol. 29, pp. 8326–8338, 2020.

[2] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, "See more, know more: Unsupervised video object segmentation with co-attention siamese networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3623–3632.

[3] P. Ochs and T. Brox, "Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions," in *International Conference on Computer Vision*, 2011, pp. 1583–1590.

[4] I. Endres and D. Hoiem, "Category independent object proposals," in *European Conference on Computer Vision*, 2010, pp. 575–588.

[5] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 628–635.

[6] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *International Conference on Computer Vision*, 2017, pp. 2961–2969.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[9] C. Ventura, M. Bellver, A. Girbau, A. Salvador, F. Marques, and X. Giro-i Nieto, "Rvos: End-to-end recurrent network for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5277–5286.

[10] L. A. Lim and H. Y. Keles, "Foreground segmentation using a triplet convolutional neural network for multiscale feature encoding," 2018.

[11] Y.-S. Xu, T.-J. Fu, H.-K. Yang, and C.-Y. Lee, "Dynamic video segmentation network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6556–6565.

[12] M. Zhen, J. Wang, L. Zhou, S. Li, T. Shen, J. Shang, T. Fang, and L. Quan, "Joint semantic segmentation and boundary detection using iterative pyramid contexts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 666–13 675.

[13] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2701–2710.

[14] P. Tokmakov, K. Alahari, and C. Schmid, "Learning motion patterns in videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 531–539.

[15] J. Cheng, Y. Tsai, S. Wang, and M. Yang, "Segflow: Joint learning for video object segmentation and optical flow," in *International Conference on Computer Vision*, 2017, pp. 686–695.

[16] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2126.

[17] H. Xiao, J. Feng, G. Lin, Y. Liu, and M. Zhang, "Monet: Deep motion exploitation for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1140–1148.

[18] S. Li, B. Seybold, A. Vorobyov, X. Lei, and C.-C. Jay Kuo, "Unsupervised video object segmentation with motion-based bilateral networks," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 207–223.

[19] C. Gao, A. Saraf, J.-B. Huang, and J. Kopf, "Flow-edge guided video completion," in *European Conference on Computer Vision (ECCV)*, 2020, pp. 713–729.

[20] J. Song, H. Zhang, X. Li, L. Gao, M. Wang, and R. Hong, "Self-supervised video hashing with hierarchical binary auto-encoder," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3210–3221, 2018.

[21] F. Shen, C. Shen, Q. Shi, A. van den Hengel, Z. Tang, and H. T. Shen, "Hashing on nonlinear manifolds," *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1839–1851, 2015.

[22] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool, "One-shot video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 221–230.

[23] P. Voigtlaender and B. Leibe, "Online adaptation of convolutional neural networks for video object segmentation," 2017.

[24] V. Jampani, R. Gadde, and P. V. Gehler, "Video propagation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 451–461.

[25] X. Li, S. Liu, S. De Mello, X. Wang, J. Kautz, and M.-H. Yang, "Joint-task self-supervised learning for temporal correspondence," in *Advances in Neural Information Processing Systems*, 2019, pp. 318–328.

[26] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung, "Learning video object segmentation from static images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2663–2672.

[27] A. Khoreva, R. Benenson, E. Ilg, T. Brox, and B. Schiele, "Lucid data dreaming for object tracking," in *The DAVIS Challenge on Video Object Segmentation*, 2017.

[28] J. Cheng, Y.-H. Tsai, W.-C. Hung, S. Wang, and M.-H. Yang, "Fast and accurate online video object segmentation via tracking parts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7415–7424.

[29] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M.-H. Yang, "Deep regression tracking with shrinkage loss," in *Proceedings of the European conference on computer vision*, 2018, pp. 353–369.

[30] X. Wang, A. Jabri, and A. A. Efros, "Learning correspondence from the cycle-consistency of time," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2566–2576.

[31] X. Xu, K. Lin, Y. Yang, A. Hanjalic, and H. T. Shen, "Joint feature synthesis and embedding: Adversarial cross-modal retrieval revisited," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

[32] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *European Conference on Computer Vision*, 2010, pp. 282–295.

[33] K. Fragkiadaki, G. Zhang, and J. Shi, "Video segmentation by tracing discontinuities in a trajectory embedding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1846–1853.

[34] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *International Conference on Computer Vision*, 2011, pp. 1995–2002.

[35] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3899–3908.

[36] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Proceedings of the European conference on computer vision*, 2018, pp. 715–731.

[37] W. Wang, X. Lu, J. Shen, D. J. Crandall, and L. Shao, "Zero-shot video object segmentation via attentive graph neural networks," in *International Conference on Computer Vision*, 2019, pp. 9236–9245.

[38] X. Chen, Z. Li, Y. Yuan, G. Yu, J. Shen, and D. Qi, "State-aware tracker for real-time video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9384–9393.

[39] B. K. Horn and B. G. Schunck, "Determining optical flow," vol. 281, pp. 319–331, 1981.

[40] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *International Conference on Computer Vision*, 2015, pp. 2758–2766.

[41] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2462–2470.

[42] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1777–1784.

[43] Y. Liu, C. Shen, C. Yu, and J. Wang, "Efficient semantic video segmentation with per-frame inference," 2020.

[44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[45] S. Jetley, N. A. Lord, N. Lee, and P. H. S. Torr, "Learn to pay attention," 2018.

[46] L. Peng, Y. Yang, Z. Wang, Z. Huang, and H. T. Shen, "Mra-net: Improving vqa via multi-modal relation attention network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 10.1109/TPAMI.2020.3004830, 2020.

[47] X. Xu, K. Lin, L. Gao, H. Lu, H. T. Shen, and X. Li, "Cross-modal common representations by private-shared subspaces separation," *IEEE Transactions on Cybernetics*, pp. 1–14, 2020.

[48] X. Xu, T. Wang, Y. Yang, L. Zuo, F. Shen, and H. T. Shen, "Cross-modal attention with semantic consistence for image-text matching," *IEEE Transactions on Neural Networks and Learning Systems*, p. 10.1109/TNNLS.2020.2967597, 2020.

[49] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S. C. Hoi, and H. Ling, "Learning unsupervised video object segmentation through visual attention," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3064–3074.

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[51] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[52] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[53] S. Xie and Z. Tu, "Holistically-nested edge detection," in *International Conference on Computer Vision*, 2015, pp. 1395–1403.

[54] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 724–732.

[55] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *Proceedings of the European Conference on Computer Vision*. Springer, 2010, pp. 282–295.

[56] P. Ochs, J. Malik, and T. Brox, "Segmentation of moving objects by long term video analysis," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 6, pp. 1187–1200, 2013.

[57] A. Faktor and M. Irani, "Video segmentation by non-local consensus voting." in *BMVC*, vol. 2, no. 7, 2014, p. 8.

[58] M. Keuper, B. Andres, and T. Brox, "Motion trajectory segmentation via minimum cost multicuts," in *International Conference on Computer Vision*, 2015, pp. 3271–3279.

[59] P. Tokmakov, K. Alahari, and C. Schmid, "Learning video object segmentation with visual memory," in *International Conference on Computer Vision*, 2017, pp. 4481–4490.

[60] Y. Jun Koh and C.-S. Kim, "Primary object segmentation in videos based on region augmentation and reduction," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3442–3450.

[61] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8934–8943.

[62] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, pp. 509–522, 2002.

[63] Y.-T. Hu, J.-B. Huang, and A. G. Schwing, "Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation," in *Proceedings of the European conference on computer vision*, 2018, pp. 786–802.

[64] S. Li, B. Seybold, A. Vorobyov, A. Fathi, Q. Huang, and C.-C. Jay Kuo, "Instance embedding transfer to unsupervised video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6526–6535.

Yifeng Zhou received the B.S degree from Nanjing University of Posts and Telecommunications, China, in 2019. He is currently pursuing his master's degree in Center for Future Media and School of Computer Science and Engineering, University of Electronic Science and Technology of China, China. His research interests include video analysis, computer vision, and machine learning.

Xing Xu (M'15) received the B.E. and M.E. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2009 and 2012, respectively, and the Ph.D. degree from Kyushu University, Fukuoka, Japan, in 2015. He is currently with the School of Computer Science and Engineering, University of Electronic of Science and Technology of China, China. He is the recipient of six academic awards, including the IEEE Multimedia Prize Paper 2020, Best Paper Award from ACM Multimedia 2017, and the World's FIRST 10K Best Paper Award-Platinum Award from IEEE ICME 2017. His current research interests mainly focus on multimedia information retrieval and computer vision.

Fumin Shen received the bachelor's degree from Shandong University in 2007 and the Ph.D. degree from the Nanjing University of Science and Technology, China, in 2014. He is currently a professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His major research interests include computer vision and machine learning. He was a recipient of the Best Paper Award Honorable Mention from ACM SIGIR 2016 and ACM SIGIR 2017 and the World's FIRST 10K Best Paper Award-Platinum Award from the IEEE ICME 2017.

Xiaofeng Zhu received his PhD in computer science from The University of Queensland, Australia. He is currently a professor with the University of Electronic Science and Technology of China, Chengdu, China. He has published more than 120 peer-reviewed research papers such as TIP, TNNLS, TKDE, ACM Multimedia, CVPR, AAAI, IJCAI, and MICCAI. His current research interests include large-scale multimedia retrieval, feature selection, sparse learning, data preprocess, and medical image analysis.

Heng Tao Shen (SM'10) is Professor and Dean of School of Computer Science and Engineering, Executive Dean of AI Research Institute, and Director of Center for Future Media at the University of Electronic Science and Technology of China. He obtained his BSc with 1st class Honours and PhD from Department of Computer Science, National University of Singapore in 2000 and 2004 respectively. He then joined the University of Queensland and became a Professor in late 2011. His research interests mainly include Multimedia Search, Computer Vision, and Artificial Intelligence. He has published 300+ peer-reviewed papers, including 100+ IEEE/ACM Transactions, and received 8 Best Paper Awards from international conferences, including the Best Paper Award from ACM Multimedia 2017 and Best Paper Award - Honourable Mention from ACM SIGIR 2017. is/was an Associate Editor of ACM Transactions of Data Science, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, and IEEE Transactions on Knowledge and Data Engineering. He is an ACM Fellow and an OSA Fellow.