Higher-Order Conditional Random Field established with CNNs for Video Object Segmentation

Chuanyan Hao[1], Yuqi Wang[1], Bo Jiang[1], Sijiang Liu[1] and Zhi-Xin Yang[2*]
[1] School of Education Science and Technology, Nanjing University of Posts and Telecommunications
Nanjing 210023, China
[e-mail: hcy@njupt.edu.cn, 1020162914@njupt.edu.cn, jiangbo@njupt.edu.cn, liusj@njupt.edu.cn]
[2] State Key Laboratory of Internet of Things for Smart City, Department of Electromechanical Engineering
University of Macau, Macau 999078, China
[e-mail: zxyang@um.edu.mo]
*Corresponding author: Zhixin Yang

Received July 22, 2021; revised August 12, 2021; accepted August 21, 2021;
published September 30, 2021

Abstract

We perform the task of video object segmentation by incorporating a conditional random field (CRF) and convolutional neural networks (CNNs). Most methods employ a CRF to refine a coarse output from fully convolutional networks. Others treat the inference process of the CRF as a recurrent neural network and then combine CNNs and the CRF into an end-to-end model for video object segmentation. In contrast to these methods, we propose a novel higher-order CRF model to solve the problem of video object segmentation. Specifically, we use CNNs to establish a higher-order dependence among pixels, and this dependence can provide critical global information for a segmentation model to enhance the global consistency of segmentation. In general, the optimization of the higher-order energy is extremely difficult. To make the problem tractable, we decompose the higher-order energy into two parts by utilizing auxiliary variables and then solve it by using an iterative process. We conduct quantitative and qualitative analyses on multiple datasets, and the proposed method achieves competitive results.

Keywords: Video object segmentation, Conditional random field, Convolution Neural Networks, Higher-order potential,

This work was partially supported by the National Natural Science Foundation of China (Grant Nos. 61802197, 61907025, 61807020) and is also funded in part by the Science and Technology Development Fund, Macau SAR (File Nos. SKL-IOTSC-2018-2020, 0018/2019/AKP ,0 0 08/2019/AGJ, and FDCT/194/2017/A3), in part by the University of Macau under Grant MYRG2018-00248-FST and MYRG2019-0137-FST.

1. Introduction

Video object segmentation (VOS) refers to the separation of a foreground object from the background of a video sequence. This task can be roughly categorized as unsupervised and supervised. Unsupervised methods do not require any annotated data, whereas supervised methods require the annotation of the foreground object in the video sequence. For more accurate segmentation of specific objects, we consider supervised methods in this study. The combination of traditional and new algorithms, such as conditional random fields (CRFs) and convolutional neural networks (CNNs), significantly promotes the development of existing research on video object segmentation. Although considerable progress has been made in this regard, the segmentation results are still unsatisfactory when the video scene is extremely complicated, for example, the disappearance and reappearance of an object and object occlusion. VOS is a basic task in the field of computer vision and has important applications in video classification and video editing.

Recently, CNN-based methods have achieved excellent results in many vision tasks, such as object detection [1], image editing [2], prediction task [3-4], and classification task [5] and so on. As for VOS, among the recently proposed advanced algorithms, a method based on the combination of a probabilistic graph model and CNN has obtained significant results. The combination of a probabilistic graph model and deep learning algorithm has also been employed. The latest research in this field shows that the combination of a probabilistic graph model and deep learning algorithm can significantly enhance the accuracy of the model. Specifically, the CNN-based method [6] has strong object representation ability and high-order dependency representation ability, whereas the probabilistic-graph-model-based method has limited expression ability owing to its own reasons; thus, it cannot model complex scenes or complex dependencies. If one model can exhibit the advantages of both models, the accuracy of the new model will significantly increase. This requires a new algorithm that can integrate the benefits of both to make the model more sensitive to the appearance information of the object and make better use of the temporal information. In addition, considering the high complexity of optical flow calculation, we propose a new filtering mechanism to improve the efficiency of optical flow calculation to obtain the temporal information of video sequences. The experimental results show that the proposed model is competitive in terms of both accuracy and efficiency.

We propose a higher-order CRF model for treating the task of VOS as a problem of finding the best labeling node in the graph model, illustrated in **Fig. 1**. Our model attempts to embed the computational process of a CNN into the iterative updating process of CRFs. Specifically, the temporal potential in our model is produced by a color histogram, and the spatial potential in our model is produced by a color histogram and optical flow orientation histogram. Existing methods have limited presentation capabilities for the object, making it impossible to effectively model a complicated segmentation scene. Some higher-order energies [7] based on global feature limitations have been suggested to solve these problems. The higher-order energy model based on CNNs is superior to that based on traditional features.

In this study, CNNs encode the unary potential and higher-order potential. We train a CNN to refine the coarse mask of the input in an entire video by using a reference frame and mask. We assume that the mask can be refined effectively and efficiently by employing trained CNNs, and we can define a function to evaluate a given mask as a whole. The higher-order potential can then be established by using a CNN-based function over the pixels within a frame. Thus, when complicated scenarios appear, our model can deal with the segmentation of objects. Finally, the higher-order energy dependent on CNNs is integrated into the Markov random

field (MRF) inference. However, the optimization of the energy equation is a difficult problem because of the existence of higher-order energy terms. Thus, we apply very efficient method to solve this problem. By introducing auxiliary variables, we first decouple the optimization equation into two parts and then use iterative algorithms. In this process, the higher-order energy term based on the CNN does not need to be calculated specifically. Our approach achieves competitive performance on the DAVIS 2016 dataset.

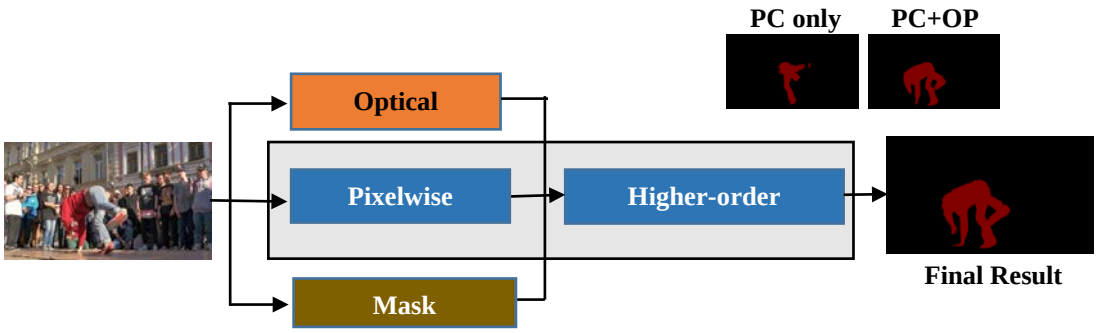

Fig. 1. Structure of our model. We construct a high-order conditional random field for video object segmentation. In this method, the segmentation problem is transformed into a node-labeling problem in the graph model, and the final segmentation result is obtained by modeling the high-order energy equation. Unary potential and high-order potential are constructed by using a convolutional neural network, and pairewise potential is constructed by applying color feature and motion feature.

2. Related Work

2.1 Unsupervised VOS

Unsupervised approaches do not require labeled data to be entered and automatically remove the object of interest from the video. The point trajectory segments video objects by analyzing the long-term motion information for pixels. In general, pixels belonging to the same object have a similar direction of motion and speed. Thus, motion information plays an important role in correctly separating objects from a video. In [8], the long-term motion information point trajectory was used to segment video objects, and promising results were obtained. Specifically, these approaches produce point trajectories and group them together. These clustered points are used to prioritize the segmentation of video objects. Over-segmentation methods [9] cluster pixels according to traditional features (color and texture) and then establish a spatial temporal MRF model. These methods generate oversegmentation regions. The oversegmentation algorithm is very important in the traditional object segmentation algorithm, which is between standard VOS algorithms and pixel matching. This method significantly reduces the computational cost because it is based on block matching rather than pixel matching. However, this algorithm cannot handle complicated segmentation scenarios.

In [10], the results of salience detection were used as prior video object segmentation information. In [11], certain region selection techniques were used to select a number of candidate objects on each frame, sort on the basis of the score, and select the most likely candidate block among the candidates. This method was improved by [12] by fully utilizing the repeatability of the object in a video sequence. Finally, the detection of saliency and the proposal for objects were treated as preprocessing for VOS. However, these techniques always produced inaccurate outputs, resulting in unsatisfactory segmentation outcomes. Moreover, these techniques are costly in terms of their calculations.

Although these methods of video segmentation have benefits, they are only sufficient for minimal scenes, where the object to be segmented varies considerably from the context. In addition, because of the large number of invalid calculations, the model incurs high computational costs. As a result, the supervised method is widely used to reduce the computational complexity of the model and improve its segmentation accuracy.

2.2 Supervised VOS

The supervised approach can be solved by inserting label data to provide guidance information for the model to solve the problem when the unsupervised segmentation method cannot define particular segmentation artifacts. Work in [9] determines how to propagate the labeled data to the video as a whole. Chen et al. [8] use the motion information of pixels to propagate through the optical flow, and the main objective of this technique is to locate the corresponding point on the frames for each pixel. Literature [13] proposes a method of pixel block matching to obtain a pixel trajectory in the video and then propagate the labeled data to the entire video sequence by trajectory. However, this method cannot solve the problems of occlusion and rotation in the video. A kind of optical flow based techniques are suggested in [14-15], that incorporates trajectory and object segmentation. First, the optical flow was used to obtain the motion trajectory of the pixels, and then, the pairwise differences were calculated for all trajectories. The results were recorded in a two-dimensional matrix known as the adjacency matrix. The spectral clustering algorithm was used to divide all trajectories into objects and backgrounds under supervision of the labeled information, and then the segmentation was completed. The accuracy of the results produced by these methods depends on the accuracy of the motion estimation.

The other methods [16] are completely different from the previous method. They usually construct a probability model for the foreground (object) and background using labeled data and then predict the label probability of pixels in the frames. Finally, the pixel can be labeled according to the probability that the pixel belongs to the foreground and background. Work in [8] set a Gaussian mixture model on the labeled object and background data and then used the model to predict the next frame to update the probability model repeatedly with the segmentation results. Literature [7] added an energy constraint of a higher order to ensure the consistency of superpixel segmentation. A long-term strategy is proposed in [17] to improve the global consistency. In particular, the problem of VOS is transformed into the problem of spatiotemporal propagation of labels.

2.3 CNN-based Methods

In recent years, many methods have been applied to VOS owing to the excellent performance of CNNs in static image segmentation. These techniques can be roughly categorized into two classes: one based on motion and one based on detection. The distinction between the two techniques is whether the motion data are considered. Temporal information is a significant indicator of the segmentation of video objects.

In general, motion-based approaches make the most use of the temporal consistency of the moving object; in particular, pixels in the same object have similar motion vectors in each frame. A combination of optical flow and deep networks was proposed in [18]. The optical flow is very important for the use of the model's temporal information. Some methods [19] take advantage of the optical flow to maintain a consistency of motion between frames and improve the accuracy of model segmentation. A CNN-based spatial-temporal MRF model was proposed by [20-21]. In [22], the optical flow was used to enhance label propagation. A combination of a CNN and recurrent neural network for VOS was proposed in [23].

Some methods use the learned object appearance to perform pixel-level detection of objects in each frame of the video. These approaches depend on fine-tuning a trained CNN using reference frame annotation. In [6], a model that combines offline training with online fine-tuning was proposed. This model fine-tunes a CNN in the video reference frame. Subsequently, in [24], an online adaptive network was proposed for VOS, where the first frame of a given video sequence was used by the network to fine-tune the changes in the appearance of the object.

CNN-based approaches can be divided into two categories: one is based on motion data and the other is based on pixel detection. The classification is based on whether the motion information between frames is used as a cue for the segmentation of video artifacts. The segmentation accuracy of the model may be improved by using reliable motion information between frames. When the object's appearance and position change smoothly, the model can easily solve the complex deformation and displacement problems of the object. However, these models are easily affected by occlusion and rapid motion. Furthermore, because the model makes full use of motion information between frames, it is robust in case of occlusion and fast motion. However, when the foreground object and background are similar in appearance, it is difficult for the model to segment the object precisely.

3. Proposed Method

Given a video sequence, $V = \{f_1, f_2, \ldots, f_n\}$, the objective is to segment the foreground object from V. The discrete random field, X, is defined over all pixels in V and $l(X) \in \{0,1\}$ to denote the labeling of all pixels. The proposed method is used to inference and minimize $E(X)$,

$$l^*(X) = \text{argmin}_{l(X)} E(X) \tag{1}$$

where $E(X)$ is defined as

$$E(X) = E_u(X) + \alpha \cdot E_p(X) + \beta \cdot E_h(X) \tag{2}$$

$E_u(X)$, $E_p(X)$, and $E_h(X)$ denote the unary potential, pairwise potential, and high-order potential, respectively. α and β are the weights used to balance this term with other terms. These terms are described in detail in the following sections.

3.1 Unary Potential

The deep visual word model has been shown to be effective in the segmentation of video objects, and we used it to generate the unary potential of each pixel as **Fig. 2** shown. In detail, a fixed number of cluster centroids is used to represent an object in an embedding space, and the range of the metric learning method is interpolated. Each centered cluster in the embedding space represents a portion of the foreground object in the current frame. We used a deep visual word model to represent each object in the frame.

The use of a deep visual word model helps matching to be more robust. Some parts of an object may remain consistent even though the object as a whole may be occluded, distorted, or vanish in the remaining frames of the same video sequence.

First, in the first frame, f_1, we input all pixels into a CNN, f_θ, to calculate the embedding for each pixel, x_i, which forms the support set, S. Then, for all pixels, we compute the visual words. Let the set of background pixels be S_b, and the set of foreground pixels be S_f, where $S = S_b \cup S_f$. The k-means algorithm was used to partition each set into K clusters, $S_b^1, \ldots, S_b^K$ and $S_f^1, \ldots, S_f^K$. φ_b^k and φ_f^k denote the respective cluster centroids, where

$$\begin{cases} S_b^1, \dots, S_b^K = \text{argmin}_{S_b^1,\dots,S_b^K} \sum_{k=0}^{K} \sum_{x_i \in S_b^k} \left\| f_\theta(x_i) - \varphi_b^k \right\|_2^2 \\ S_f^1, \dots, S_f^K = \text{argmin}_{S_f^1,\dots,S_f^K} \sum_{k=0}^{K} \sum_{x_i \in S_f^k} \left\| f_\theta(x_i) - \varphi_f^k \right\|_2^2 \end{cases} \tag{3}$$

Here, φ_b^k and φ_f^k are defined as follows:

$$\begin{cases} \varphi_b^k = \frac{1}{S_b^K} \sum_{x_i \in S_b^k} f_\theta(x_i) \\ \varphi_f^k = \frac{1}{S_f^K} \sum_{x_i \in S_f^k} f_\theta(x_i) \end{cases} \tag{4}$$

Finally, a deep visual word model was used to represent the pixel label. In other words, the matching probability of pixels and visual words can be defined as follows:

$$\begin{cases} p\left(l(x_j) = 1 \middle| x_j\right) = \frac{1}{\sigma} \sum_{k=1}^{K} \left\| f_\theta(x_i) - \varphi_f^k \right\|_2^2 \\ p\left(l(x_j) = 0 \middle| x_j\right) = \frac{1}{\sigma} \sum_{k=1}^{K} \left\| f_\theta(x_i) - \varphi_b^k \right\|_2^2 \end{cases} \tag{5}$$

Here, σ is defined as follows:

$$\sigma = \sum_{k=1}^{K} \left[\left\| f_\theta(x_i) - \varphi_f^k \right\|_2^2 + \left\| f_\theta(x_i) - \varphi_b^k \right\|_2^2 \right] \tag{6}$$

The unary potential is represented by the negative log likelihood of the labeling for each single random variable as follows:

$$E_u(X) = -\log p\left(y_i = 1 \middle| x_j\right)[y_i = 1] - \log p\left(y_i = 0 \middle| x_j\right)[y_i = 0] \tag{7}$$

Here, $[*] = 1$, when $*$ is true; otherwise, $[*] = 0$, y_i is the label assigned by our proposed method.

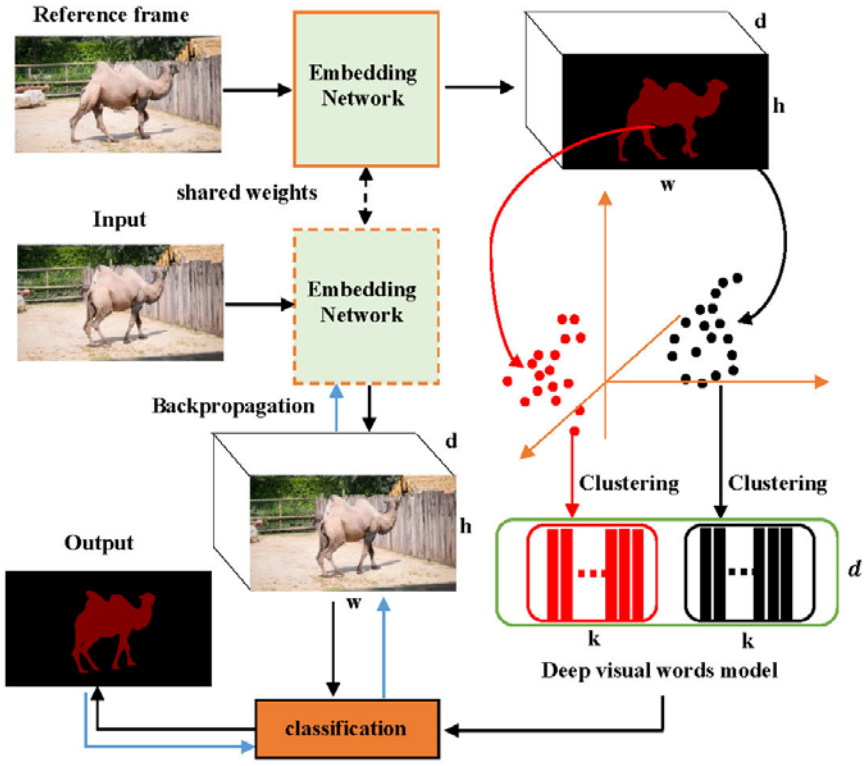

Fig. 2. Overview of constructing a visual word model. The embedding network is a deep CNN, and its input is the reference frame (generally the first frame) and its mask. It outputs a high-dimensional vector with dimension d and performs clustering for each class with subclasses of k. The centroid of each cluster is selected from among them as a guide for establishing the visual word model.

3.2 Pairwise Potential

Pairwise potential is often used to decrease the discontinuity of pixels having the same label such that, owing to some interfering factors, the neighboring pixels of the same label do not break. Similar to other methods, the pairwise potential is primarily used for spatial-temporal smoothness in the proposed approach. Two pixels are spatially connected if they share one edge, and two superpixels are temporally connected if they have in-between pixels linked by optical flow. In particular, we used the color and optical flow orientation of the histogram to calculate the local similarity. ε_t denotes the time pair set, and ε_s denotes the spatial pair set. Thus, E_p can be defined as follows:

$$E_p(X) = \theta_s \cdot \sum_{(s,s')\epsilon\varepsilon_s} E_p^s(s,s') + \theta_t \cdot \sum_{(t,t')\epsilon\varepsilon_t} E_p^t(t,t') \tag{8}$$

Here, $E_p^s(s,s')$and $E_p^t(t,t')$ are the energies linked to the spatial and temporal dependencies, respectively. θ_s and θ_t are the two weight parameters for a linear combination. Overall, the spatial and temporal pairwise potentials are defined as follows:

$$\begin{cases} E_p^s = [l(s) \neq l(s')] \cdot \exp\left(-\sigma_h^{-1}\|h(s) - h(s')\|^2\right) \cdot \exp\left(-\sigma_c^{-1}\|c(s) - c(s')\|^2\right) \\ E_p^t = [l(t) \neq l(t')] \cdot \exp\left(-\sigma_c^{-1}\|c(t) - h(t')\|^2\right) \end{cases} \tag{9}$$

Here, $h()$ is a normalized histogram of optical flow discretized with respect to the angle, and $c(\cdot)$ is the color histogram. σ_h and σ_c are the two weight parameters for the balance.

3.3 High-order Potential

In general, when the shape of the object is irregular and the speed of motion between frames is too high, the quadratic energy equation is not sufficient to handle complex segmentation scenarios. Local information constraints cannot be addressed effectively. The higher-order energy term, based on global constraints, is, therefore, considered to enhance the ability of the model to deal with complex scenarios.

We represent all pixels in a frame as a clique for high-order dependencies, where the labeling for each pixel depends on all other pixels in the same frame. We formulate an energy function, $f_\tau(\cdot)$, to evaluate a given mask, Y_{mask}. To build high-order dependencies with all pixels in the current frame, we define $f_\tau(\cdot)$ as

$$f_\tau(\cdot) = \|Y_{mask} - Y_{mask}^*\|_2^2 \tag{10}$$

If the current mask is more similar to the mask of the ground truth, then it always has a very low energy penalty while Y_{mask}^* is unsolved. Here, we approximate Y_{mask}^* by means of a CNN and define $f_\tau(\cdot)$ as

$$f_\tau(\cdot) = \left\|Y_{mask} - \mathrm{RGMP}(f_1, Y_{mask}^{f_1}, f_i, Y_{mask})\right\|_2^2 \tag{11}$$

where RGMP() is a CNN-refined mask that accepts f_1 for the reference frame given in it, $Y_{mask}^{f_1}$ for the reference frame, current frame f_i for the previous frame, and Y_{mask} for the previous frame, and the refined mask is the output. It can be observed visually that this definition apportions a lower energy to a mask when the refined mask is more similar to itself. The $f_\tau(\cdot)$ function can allocate better masks to lower energies, and RGMP() can refine a coarse mask to a better one and hold a decent mask unchanged. Thus, it is possible to define the high-order potential in the proposed method as $E_h(X) = f_\tau(\cdot)$.

3.4 Inference

The aforementioned higher-order energy definition is more expressive than the traditional higher-order energy definition, but the problem of optimization in the MRF is intractable. In this paper, we propose an approximate method to solve the inference problem.

We decouple the pairwise potential, E_p, and higher-order potential, E_h, to solve the problem by adding an auxiliary variable, Y, and (12) can be approximated as follows:

$$E(X,Y) = E_u(X) + \alpha \cdot E_p(X) + \beta \cdot E_h(Y) + \Lambda \cdot \|X - Y\|_2^2 \quad (12)$$

Specifically, Y is a near approximation of X. This function can be solved by iteratively updating either X or Y. Here, we use a classical iterative method called iterated conditional modes (ICM) for efficiency considerations. In particular, we update $X_i \epsilon X$ to minimize $E(X)$, whereas the rest of the variable, X, is fixed. The ICM method constantly updates variables until convergence, or achieves the number of iterations we set. Generally, the number is set to K. The specific optimization process is shown in the above algorithm.

Algorithm 1 Optimization algorithm

Input: The outer loop K, the inner loop L, total number of pixels P, and the number of frames F
Output: the segmentation masks $y^{(K)}$
Initialization: $x^{(0)} = y^{(0)}$
for k from 1 to K do
  $x^{(k,0)} \leftarrow x^{(k-1)}$
  for l from 1 to L do
    for i from 1 to P do
      $x_i^{(k,l)} \leftarrow argmin_{x_i} \left\{ \Lambda(x_i - y_i^{(k-1)})^2 + E_u(x_i) + \sum_{(i,j)\in\eta_\tau} E_t\,(x_i, x_j^{(k,l-1)}) \right\}$
  $x(k) \leftarrow x^{(k,L)}$
  for c from 1 to F do
    $y_c^{(k)} \leftarrow rgmp(x_c^{(k)})$
return $y^{(K)}$

4. Implementation and Training Details

4.1 Implementation Details

Details of $f(\theta)$. We used the Deeplabv2 [25] model trained on the COCO dataset [26] as the $f(\theta)$ encoder. The $f(\theta)$ encoder converts the input frame pixels into higher-dimensional vectors and uses a bilinear interpolation algorithm to process and restore the image to its original image size. Finally, a clustering algorithm was used to cluster these high-dimensional vectors to form a visual word model. Under normal circumstances, we classified the foreground and background clusters into $K = 50$ clustering centers.

Details of $f_\tau(\cdot)$. We propose a novel structure of the encoder and decoder, the input of which is the reference frame and the mask, the mask of the previous frame, and the current frame, and the final output of which is the precise frame mask. The network, shown in **Fig. 3**, is composed of two encoder-sharing parameters: a decoder and a convolution block. A guidance stream and target stream are included in the encoder. The reference stream input includes the reference image (first frame) and the ground truth mask. A guide mask and target image corresponding to the previous frame are provided for the target stream. The encoder was built using ResNet50. We modified it to accept the four-channel vector input by inputting an extra single-channel filter in the first convolution layer. Except that newly added filters are randomly initialized, all weights of our model are initialized with a pretrained ImageNet network. The outputs from the two encoder streams are merged and then input into the global convolution block. The module performs global feature matching and obtains the contours of the

foreground object. We use global convolution to overcome the locality of the convolution operation, effectively expanding the receptive field. Our decoder accepts the output of the global convolution block via a skip connection and generates a mask output in the target encoder stream. We used the refinement module to effectively merge the features of different scales. Based on the original structure, we improved and replaced the convolutional layer with the remaining blocks. To produce object masks, three refinement blocks, a softmax layer, and a final convolutional layer were included in our decoder.

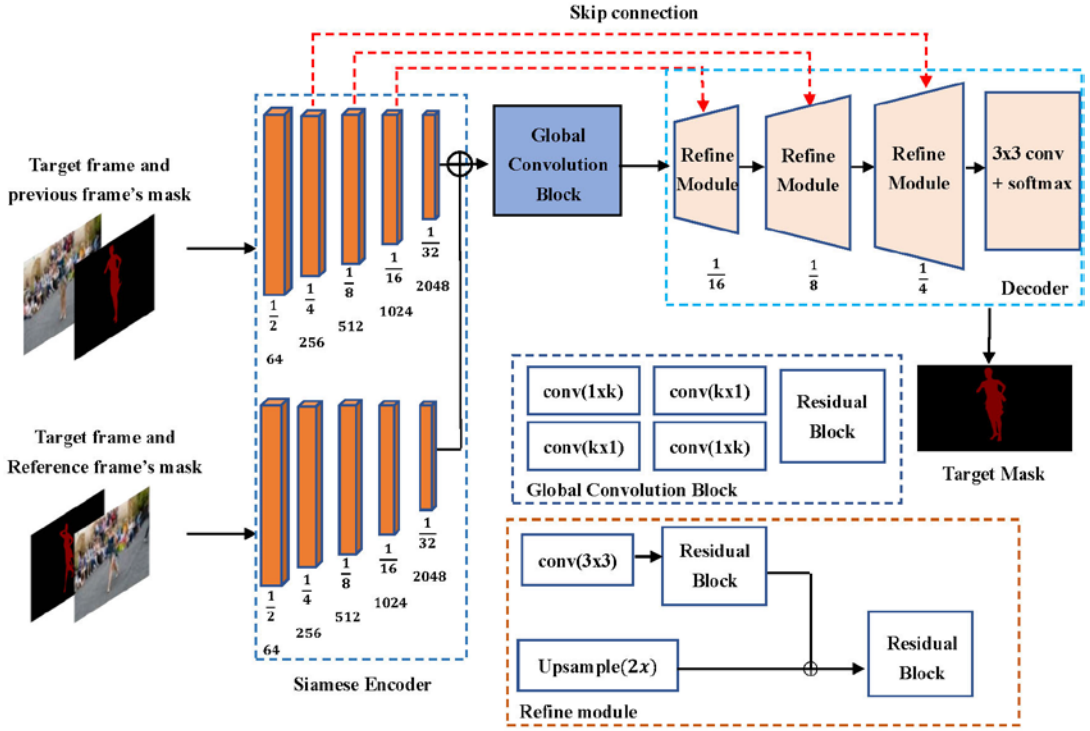

Fig. 3. Overview of the proposed network. The network is made up of two encoder-sharing parameters, a decoder, and a global convolution block. The specific composition of each structural block is shown below.

4.1 Details of Training

Training of $f(\theta)$. We assume that there are internal changes in the object in the video, and the standard loss function is insufficient for VOS. In other words, if the identity of the sample is obvious, then a triple loss function is designed. This is not the case for VOS because an object may have many portions, and each component may be different. Therefore, it is an additional constraint to pull these samples very close together and can be detrimental to learning a robust metric. We changed the typical triplet loss to conform to the task of video target segmentation. We officially call an anchor sample, x^a. $x^p \in p$ is a positive sample in a positive sample pool of p. Likewise, x^nrepresents a negative sample pool, and γ represents a negative sample pool. The standard triplet loss makes a correct probability large enough to increase the advantage and avoid ambiguity. Therefore, we only push the smallest negative point away from the smallest positive point by modifying the loss function. The loss function is defined as:

$$\sum_{x^a \in A} \min_{x^p \in p} \|f(x^a) - f(x^p)\|_2^2 - \min_{x^n \in \gamma} \|f(x^a) - f(x^n)\|_2^2 + \alpha \tag{13}$$

Among them, α is a balance variable that controls the distance between negative and positive samples, and the anchor is defined as A. We have two pools for each anchor sample, x^a: one is the positive sample pool, P, the label of which is the same as the anchor sample; the other is the negative sample pool, γ. We selected the sample closest to the anchor point of each pool and compared the negative and positive distances. Intuitively, the loss function brings the nearest positive factors closer and pushes the nearest negative factor farther away.

The anchor points are sampled from one frame, and the pixels in the other two frames are linked together. The positive pool, P, forms pixels with the same mark as the anchor point, and the remainder forms the negative pool, γ. Note that to have temporal variation, the pool is sampled from two different frames; to prevent bad samples, we do not select pixels from anchor frames in pools.

One frame was used as an anchor point in each iteration, and forward passing was carried out on three randomly selected frames. To sample 256 anchor samples, we then used the anchor frame, and the positive and negative pools were both foreground and background pixels in the other two frames. According to (13), we calculated the loss, and the network was trained end-to-end.

Training of $f_\tau(\cdot)$. We used patches of 256×256 and 256×512 to perform pretraining and then go on to fine-tuning. In the fine-tuning period, the number of repetitions was set to five. We used a random affine transformation to expand all of the training samples. We used the Adam optimizer for all of our experiments with a fixed learning rate of e^{-5}. With a single NVIDIA GeForce 1060 GPU, fine-tuning required approximately three days, and pretraining took approximately five days.

Our network was first trained on the static image datasets and then fine-tuned on the VOS datasets. First, we used an image dataset with instance object masks (PascalVOC) to simulate training samples. Specifically, we used the method of random affine transformation to further transform the mask of the target frame. We randomly processed a training sample from each generated image that contained at least 50% of the object. After pretraining, we performed fine-tuning training on the VOS dataset. Through fine-tuning training in real VOS scenarios, our model learned how to segment accurately after adapting to changes in the appearance and motion of the object.

5. Experimental Results

5.1 Ablation Study

We performed ablation studies using the DAVIS2016 dataset. We evaluated the accuracy of the model using the contour accuracy ($\mathcal{F}$) and regional similarity ($IoU(\mathcal{J})$). **Table 1** lists the results of the proposed model for various configurations. It is clear that the model can produce better output results and is more robust when it contains higher-order energy.

Table 1. The ablation study on DAVIS 2016. UP represents there is the unary potential in the model only. UP&PP represents there is not the high order potential in the model. UP&PP&HOP represents there is high order potential in the model.

Approach	$IoU(\mathcal{J})$			$\mathcal{F}$		
	Mean	**Recall**	**Delay**	**Mean**	**Recall**	**Delay**
UP	81.5	-	5.0	82.7	-	-
UP&PP	83.6	-	7.8	85.3	-	-
UP&PP&HOP	**84.3**	95.3	13.7	**86.2**	93.2	15.6

In our experiments, when only performing the unary potential (UP), we set $\alpha = \beta = 0$. When only performing pairwise and unary potentials (PP&UP), we set $\beta = 0$. When performing pairwise, unary, and high-order potentials (UP&PP&HOP), we set $\alpha = \beta = 1$.

It can be seen from the results that the higher-order potential proposed in this study significantly improves the accuracy of the results. Specifically, without the higher-order potential based on the global result, the segmentation results of a single frame are often disturbed by local information. In particular, when the color difference between the foreground and background is small, the foreground object and background are often confused, resulting in false segmentation. However, when global higher-order potential constraints are added, the algorithm can optimize the segmentation results of each frame by conducting feature statistics on the entire video. Because this method uses global information to optimize local information, the proposed model has strong robustness against random noise, irregular motion, and small differences in foregrounds and backgrounds, as shown in **Fig. 4**.

Fig. 4. Results of the ablation study conducted at DAVIS 2016. The result is improved by adding pairwise potential energy (PP) to unitary potential energy (UP). In addition to the first two potential energies (UP and PP), the higher-order potential energy (HOP) based on global consistency constraint are added to significantly improve experimental results.

5.2 State-of-the-art Comparison

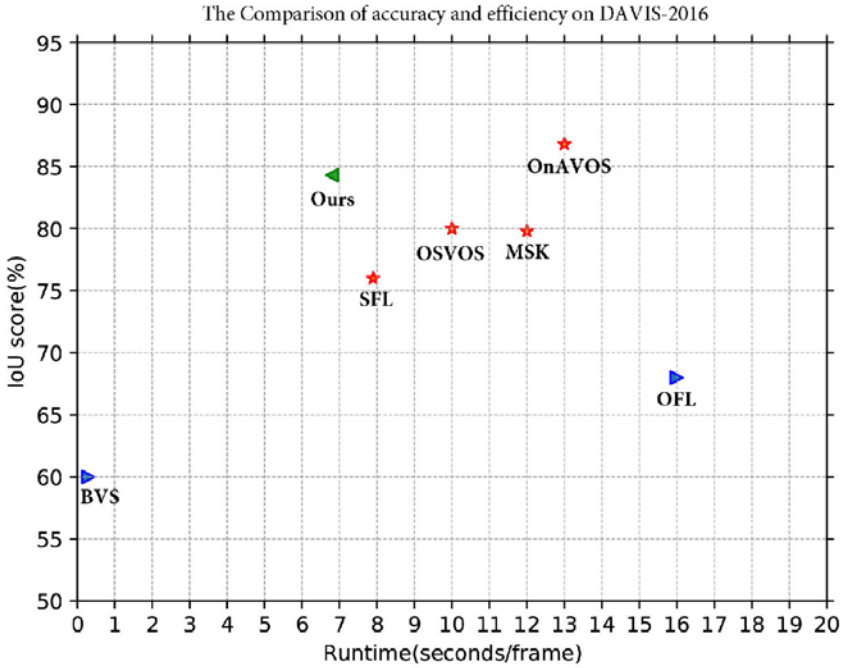

Fig. 5. Comparison of accuracy and efficiency on DAVIS-2016.

We conducted a comparative experiment on the DAVIS2016 dataset to compare the proposed model with other models. It contains a total of 50 high-resolution (480P) video sequences, and contains many challenging segmentation scenarios (appearance changes, fast motion, and occlusion). What's more, the proposed method also achieves a competitive performance in terms of accuracy and efficiency as shown in **Fig. 5**.

Table 2. The results of proposed model on the dataset of DAVIS 2016 and compared with the benchmark method published on DAVIS 2016.

Approach	Global Mean	$IoU(\mathcal{J})$			$\mathcal{F}$		
		Mean	Recall	Delay	Mean	Recall	Delay
FCP [17]	53.8	58.4	71.5	-2.0	49.2	49.5	-1.1
BVS [16]	59.4	60.0	66.9	28.9	58.8	67.9	21.3
OFL [14]	65.7	68.0	75.6	26.4	63.4	70.4	27.2
PML [27]	66.35	70.2	86.3	11.2	62.5	73.2	14.7
SiamMask [13]	69.75	71.7	86.8	3.0	67.8	79.8	2.1
CTN [28]	71.4	73.5	87.4	15.6	69.3	79.6	12.9
SFL [29]	76.05	76.1	90.6	12.1	76.0	85.5	10.4
PLM [30]	77.4	75.5	89.6	8.5	79.3	93.4	7.8
MSK [22]	77.55	79.7	93.1	8.9	75.4	87.1	9.0
OSVOS [6]	80.2	79.8	93.6	14.9	80.6	92.6	15.0
RGMP [31]	81.75	81.5	91.7	10.9	82.0	90.8	10.1
CIM [20]	84.2	83.4	94.9	12.3	85.0	92.1	14.7
STM [32]	**89.4**	**88.7**	97.4	5.0	**90.1**	95.2	4.2
Ours	85.25	84.3	95.3	13.7	86.2	93.2	15.6

Table 3. The results of proposed model on the dataset of DAVIS 2016 and compared with the benchmark method published on DAVIS 2016.

Approach	DAVIS 2016 (MIOU%)	YouTube (MIOU%)
CIM [20]	83.4	78.4
OSVOS [6]	79.8	78.3
OFL [14]	68.0	77.6
MSK [22]	79.7	72.6
STV [33]	73.6	-
VPN [18]	70.2	-
OnAVOS [24]	**86.16**	77.4
Ours	84.3	**79.2**

To make our approach more convincing, we compared our model with other models on the YouTube dataset, and the results show that our model obtained state-of-the-art results among similar algorithms. Because there are few cases of object occlusions and object appearance changes in the YouTube dataset, the algorithm based on temporal information propagation can often obtain satisfactory results easily. Although the DAVIS2016 dataset contains very challenging segmentation scenarios (occlusions and complex deformations), most foreground objects can be correctly identified and segmented by using the CNN-based approach. Thanks

to the proposed higher-order constraint, the pixels that have similar semantic features are specified as soft preferences for assignment with the same label (foreground or background). This global clique is the key to our approach and it ensures long-term appearance consistency during segmentation. By taking advantage of both the probabilistic graphical model and a CNN, our model achieves competitive results on these datasets, as shown in **Table 2** and **Table 3**.

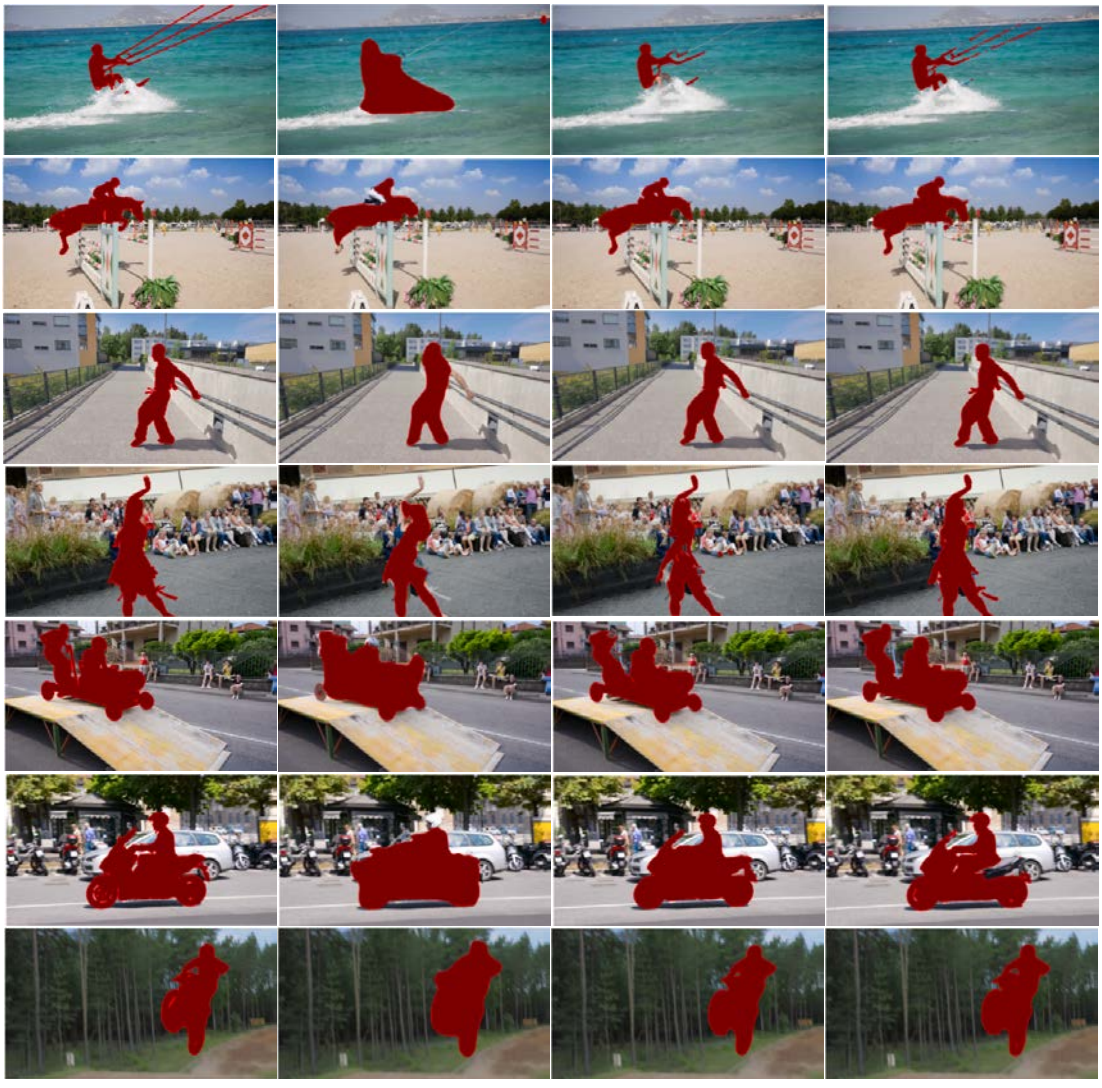

Fig. 6. Comparison of results. By comparing the results of the proposed algorithm with those of other algorithms, it can be seen from the experimental results that the proposed algorithm achieves competitive performance and satisfactory results in some very challenging segmentation scenarios.

5.3 Qualitative Evaluation

Fig. 6 shows several qualitative examples of our segmentation results and a comparison with some excellent algorithms on the DAVIS 2016 dataset. The above experimental results show that our model can produce satisfactory segmentation results in challenging scenes with object

occlusions, object appearance changes, fast motion, and small differences between the background and the foreground object. Even though the full object appearance is not revealed in the first frame, our model successfully captures the target information. Inevitably, however, the method cannot fully capture some detailed parts such as the human leg, or the foot of the rider and back seat of motorbike. This is most likely because the missed information is not detected on the object in first frame but is highly similar to that of other distractor objects. Fig. 7 shows some failure cases. This may be because the instances have similar appearances and are close to each other, resulting in excessively proximate embeddings. Moreover, the object has a blurry appearance owing to its transparent appearance and fast motion.

5.4 Limitations

Fig. 7 shows typical examples of incorrect segmentation: the small difference between the background and the foreground object causes pixels to be incorrectly labeled. The difficulty of this method lies in the optimal solution of high-order energy equations, and it is difficult to use a unified framework to deal with such problems. In general, the methods for solving high-order energy equations can be summarized as follows: The first method is to equate higher-order functions to lower-order functions (usually quadratic energy functions) through equivalent transformations, and then use the standard graph cut algorithm to solve. Another method is to approximate the high-order energy equation to the second-order energy equation, and then use the standard graph cut algorithm to solve. Our method is the latter. In future research, we hope to explore a more efficient algorithm to solve higher-order energy equations without adding any variables. At the same time, a bi-layered parallel training architecture [34] could be considered for acceleration.

Fig. 7. Typical failed experiment results. The top image is the ground truth, and the bottom image is the result of our model.

6. Conclusion

In this study, we proposed an efficient and effective higher-order CRF model for VOS. A higher-order energy equation was established to model the task of VOS. The unary potential energy and higher-order potential energy of the model were modeled by using a CNN. To solve the problem of optimization in the MRF, we decomposed a higher-order energy equation into two parts and optimized it by using a traditional iterative method. Finally, a standard graph cut algorithm was used to complete the segmentation. We performed quantitative and qualitative evaluations on multiple datasets, and the proposed model achieved competitive

results. However, the accuracy and speed of the proposed approach cannot reach the real-time requirement for some application scenarios. To solve this problem, the next step is to add all the previous frame information to predict the current frame, as well as the computing cost and memory usage issues caused by it.

References

[1] J. Cheng, Y. Liu, X. Tang, V. S. Sheng, and M. Li et al., "DDOS attack detection via multi-scale convolutional neural network," *Comput. Mater. Contin.*, vol. 62, no. 3, pp. 1317–1333, 2020. Article (CrossRef Link).

[2] T. Yang, S. Jia and H. Ma, "Research on the application of super resolution reconstruction algorithm for underwater image," *Comput. Mater. Contin.*, vol. 62, no. 3, pp. 1249–1258, 2020. Article (CrossRef Link).

[3] M. Duan, K. Li, A. Ouyang, K. N. Win, K. Li, and Q. Tian, "EGroupNet: A Feature-enhanced Network for Age Estimation with Novel Age Group Schemes," *ACM Trans. Multim. Comput. Commun.*, vol. 16, no. 2, pp. 42:1-42:23, Jun. 2020. Article (CrossRef Link).

[4] M. Duan, K. Li, K. Li, and Q. Tian, "A Novel Multi-task Tensor Correlation Neural Network for Facial Attribute Prediction," *ACM Trans. Intell. Syst. Technol.*, vol. 12, no. 1, pp. 3:1-3:22, Feb. 2021. Article (CrossRef Link).

[5] L. Pan, C. Li, S. Pouyanfar, R. Chen and Y. Zhou, "A novel combinational convolutional neural network for automatic food-ingredient classification," *Comput. Mater. Contin.*, vol. 62, no. 2, pp. 731–746, 2020. Article (CrossRef Link).

[6] S. Caelles, K. K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. V. Gool, "One-shot video object segmentation," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Honolulu, USA, pp. 5320-5329, 2017. Article (CrossRef Link).

[7] Y. Chen, C. Hao, A. X. Liu, and E. Wu, "Appearance-consistent video object segmentation based on a multinomial event model," *ACM Trans. Multimed. Comput. Com.*, vol. 15, no. 2, pp. 40:1–40:15, 2019. Article (CrossRef Link).

[8] Y. Chen, C. Hao, W. Wen, and E. Wu, "Efficient frame-sequential label propagation for video object segmentation," *Multimed. Tools Appl.*, vol. 77, no. 5, pp. 6117–6133, 2018. Article (CrossRef Link).

[9] Y. Chen, C. Hao, A. X. Liu, and E. Wu, "Multi-level model for video object segmentation based on supervision optimization," *IEEE Trans. Multimed.*, vol. 21, no. 8, pp. 1934-1945, 2019. Article (CrossRef Link).

[10] Y.-T. Hu, J.-B. Huang, and A. G. Schwing, "Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation," in *Proc. of Eur. Conf. Comput. Vis.*, Munich, Germany, pp. 813-830, 2018. Article (CrossRef Link).

[11] J. K. Yeong, and C.-S. Kim, "Cdts: Collaborative detection, tracking, and segmentation for online multiple object segmentation in videos," in *Proc. of IEEE Int. Conf. Comput. Vis.*, Venice, Italy, pp. 3621-3629, 2017. Article (CrossRef Link).

[12] J. K. Yeong, and C.-S. Kim, "Primary object segmentation in videos based on region augmentation and reduction," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Honolulu, USA, pp. 7417-7425, 2017. Article (CrossRef Link).

[13] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Long Beach, USA, pp. 1328-1338, 2019. Article (CrossRef Link).

[14] H. Y. Tsai, H. M. Yang, and J. M. Black, "Video segmentation via object flow," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Las Vegas, USA, pp. 3899-3908, 2016. Article (CrossRef Link).

[15] N. S. Rani, M. Chandrajith, B. R. Pushpa and B. R. Pushpa, "A deep convolutional architectural framework for radiograph image processing at bit plane level for gender & age assessment," *Comput. Mater. Contin.*, vol. 62, no. 2, pp. 679–694, 2020. Article (CrossRef Link).

[16] N. Märki, F. Perazzi, O. Wang, and A. Sorkine-Homung, "Bilateral space video segmentation," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Las Vegas, USA, pp. 743-751, 2016. Article (CrossRef Link).
[17] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation," in *Proc. of IEEE Int. Conf. Comput. Vis.*, Santiago, USA, pp. 3227-3234, 2015. Article (CrossRef Link).
[18] V. Jampani, R. Gadde, and P. V. Gehler, "Video propagation networks," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Honolulu, USA, pp. 3154-3164, 2017. Article (CrossRef Link).
[19] H. Xiao, J. Feng, G. Lin, Y. Liu, and M. Zhang, "Monet: Deep motion exploitation for video object segmentation," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Salt Lake City, USA, pp. 1140-1148, 2018. Article (CrossRef Link).
[20] L. Bao, B. Wu, and W. Liu, "Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Salt Lake City, USA, pp. 5977-5986, 2018. Article (CrossRef Link).
[21] C. Chen, K. Li, S. G. Teo, X. Zou, K. Li, and Z. Zeng, "Citywide Traffic Flow Prediction Based on Multiple Gated Spatio-temporal Convolutional Neural Networks," *ACM Trans. Knowl. Discov. Data*, vol. 14, no. 4, pp. 42:1-42:23, July. 2020. Article (CrossRef Link).
[22] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung, "Learning video object segmentation from static images," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Honolulu, USA, pp. 3491-3500, 2017. Article (CrossRef Link).
[23] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. Price, S. Cochen, and T. Huang, "Youtube-vos: Sequence-to-sequence video object segmentation," in *Proc. of Eur. Conf. Comput. Vis.*, Munich, Germany, pp. 603-619, 2018. Article (CrossRef Link).
[24] P. Voigtlaender, and B. Leibe, "Online adaptation of convolutional nenural networks for video object segmentation," in *Proc. of the 2017 British Mach. Vis. Conf.*, June 2017. Article (CrossRef Link).
[25] 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, 2018. Article (CrossRef Link).
[26] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollr, "Microsoft coco: Common objects in context," in *Proc. of Eur. Conf. Comput. Vis.*, Zurich, Switzerland, pp. 740-755, 2014. Article (CrossRef Link).
[27] Y. Chen, J. Pont-Tuset, A. Montes, and L. V. Gool, "Blazingly fast video object segmentation with pixel-wise metric learning," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Salt Lake City, USA, pp. 1189-1198, 2018. Article (CrossRef Link).
[28] W. D. Jang, and C. S. Kim, "Online video object segmentation vias convolutional trident network," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Honolulu, USA, pp. 7474-7483, 2017. Article (CrossRef Link).
[29] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos, "Efficient video object segmentation via network modulation," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Salt Lake City, USA, pp. 6499-6507, 2018. Article (CrossRef Link).
[30] J. S. Yoon, F. Rameau, J. Kim, S. Lee, S. Shin, and I. S. Kweon, "Pixel-level matching for video object segmentation using convolutional neural networks," in *Proc. of IEEE Int. Conf. Comput. Vis.*, Venice, Italy, pp. 2186-2195, 2017. Article (CrossRef Link).
[31] S. W. Oh, J. Lee, K. Sunkavalli, and S. J. Kim, "Fast video object segmentation by reference-guided mask propagation," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recog.*, Salt Lake City, USA, pp. 7376-7385, 2018. Article (CrossRef Link).
[32] S. W. Oh, J. Lee, N. Xu, and S. J. Kim, "Video object segmentation using space-time memory networks," in *Proc. of IEEE Int. Conf. Comput. Vis.*, Seoul, Korea, pp. 9225-9234, 2019. Article (CrossRef Link).
[33] W. Wang, S. Bing, J. Xie, and F. Porikli, "Super-trajectory for video segmentation," in *Proc. of IEEE Int. Conf. Comput. Vis.*, Venice, Italy, pp. 1680-1688, 2017. Article (CrossRef Link).

[34] J. Chen, K. Li, K. Bilal, X. Zhou, K. Li, and P. S. Yu, "A bi-layered parallel training architecture for large-scale convolutional neural networks," *IEEE Trans. Parallel Distributed Syst.*, vol. 30, no. 5, pp. 965-976, May 2019. Article (CrossRef Link).

Chuanyan Hao received the Ph.D. in Soft Engineering from University of Macau in 2015. She is currently an assistant professor in Nanjing University of Posts and Telecommunications. Her main research interests include texture synthesis and analysis, image and video processing and editing, image-based modeling and animation, data-driven approaches and so on.

Yuqi Wang is a graduate student at Nanjing University of Posts and Telecommunications. Her main research interests include educational informationization and educational evaluation.

Bo Jiang received his Ph.D. in Computer Science from State Key Lab of CAD&CG, Zhejiang University in 2014. He is currently an Assistant Professor in Nanjing University of Posts and Telecommunications. His research interests include computer vision, machine learning and their applications in education.

Sijiang Liu received the Ph.D. in Computer Applied Technology from Institute of Automation, Chinese Academy of Sciences in 2013. He is currently a lecturer in Nanjing University of Posts and Telecommunications. His main research interests include real-time rendering, virtual reality, augmented reality, computer simulation and so on.

ZhixinYang obtained his PhD in Industrial Engineering and Engineering Management from the Hong Kong University of Science and Technology. He is currently an Associate Professor in the State Key Laboratory of Internet of Things for Smart City, Faculty of Science and Technology, and the Director of Research Service and Knowledge Transfer Office both at the University of Macau. His current research interests include fault diagnosis and prognosis, machine learning, and computer vision-based robotics.