applied sciences

MDPI

Article

Complementary Segmentation of Primary Video Objects with Reversible Flows

Junjie Wu [1], Jia Li [1,*] and Long Xu [2]

[1] State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100190, China; wujunjie@buaa.edu.cn
[2] State Key Laboratory of Space Weather, National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China; lxu@nao.cas.cn
* Correspondence: jiali@buaa.edu.cn

Abstract: Segmenting primary objects in a video is an important yet challenging problem in intelligent video surveillance, as it exhibits various levels of foreground/background ambiguities. To reduce such ambiguities, we propose a novel formulation via exploiting foreground and background context as well as their complementary constraint. Under this formulation, a unified objective function is further defined to encode each cue. For implementation, we design a complementary segmentation network (CSNet) with two separate branches, which can simultaneously encode the foreground and background information along with joint spatial constraints. The CSNet is trained on massive images with manually annotated salient objects in an end-to-end manner. By applying CSNet on each video frame, the spatial foreground and background maps can be initialized. To enforce temporal consistency effectively and efficiently, we divide each frame into superpixels and construct a neighborhood reversible flow that reflects the most reliable temporal correspondences between superpixels in far-away frames. With such a flow, the initialized foregroundness and backgroundness can be propagated along the temporal dimension so that primary video objects gradually pop out and distractors are well suppressed. Extensive experimental results on three video datasets show that the proposed approach achieves impressive performance in comparisons with 22 state-of-the-art models.

Keywords: primary object segmentation; video; objective function; complementary CNNs; neighborhood reversibility

check for updates

Citation: Wu, J.; Li, J.; Xu, L. Complementary Segmentation of Primary Video Objects with Reversible Flows. *Appl. Sci.* **2022**, *12*, 7781. https://doi.org/10.3390/app12157781

Academic Editor: Andrea Prati

Received: 30 June 2022
Accepted: 30 July 2022
Published: 2 August 2022

1. Introduction

Segmenting primary objects aims to delineate the physical boundaries of the most perceptually salient objects in an image or video. Perceptual saliency means that the objects should be visually salient in image space while present in most of the video frames. This is a useful assumption that works under various unconstrained settings, thus benefiting many computer vision applications such as action recognition, object class learning [1], video summarization, video editing, content-based video retrieval and video surveillance.

Despite impressive performance in recent years [2–13], primary object segmentation remains a challenging task since in real-world images there exist various levels of ambiguities in determining whether a pixel belongs to the foreground or background. These ambiguities are more serious in video frames due to some video attributes representing specific situations, such as fast motion, occlusion, appearance changes and cluttered background [14]. Specially, these attributes are not exclusive; thus a sequence can be annotated with multiple attributes. As shown in Figure 1, due to the camera and/or object motion, the primary objects may suffer motion blur (e.g., the last dog frame), occlusion (e.g., the second dog frame) and even be out-of-view (e.g., the last two turtle frames). Moreover, the primary objects may co-occur with various distractors in different frames (e.g., the turtle video frames), making them difficult to consistently pop-out throughout the whole video.

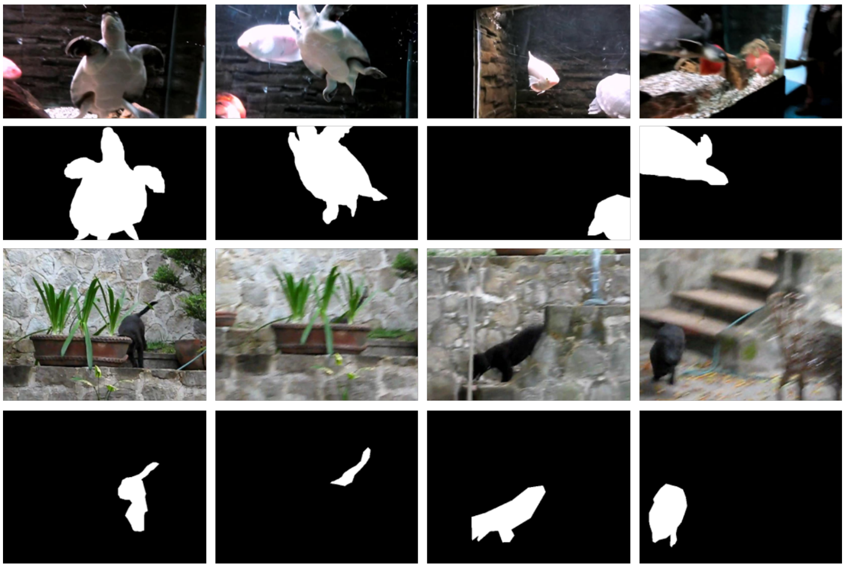

Figure 1. Primary objects may co-occur with or be occluded by various distractors. They may not always be the most salient ones in each separate frame but can consistently pop out in most video frames (the two groups of frames and masks are taken from the dataset **VOS** [15] and dataset **Youtube-Objects** [1], respectively).

To address these issues, there exist three major types of models, which can be roughly categorized into interactive, weakly supervised and fully automatic models. Of these models, interactive models require manually annotated primary objects in the first frame or several selected frames before starting automatic segmentation [16–18], while weakly supervised models often assume that the semantic tags of primary video objects are known before segmentation so that external cues such as object detection can be used [19,20]. However, the requirement of interaction or semantic tags prevents their usage in processing large-scale video data [21].

Beyond these two kinds of models, fully automatic models aim to directly segment primary objects in a single video [21–26] or co-segment the primary objects shared by a collection of videos [27–29] without any prior information about the objects. Although recently, CNNs have achieved impressive progress in object segmentation, insufficient video data with pixel-level annotations may prevent the end-to-end training of a sophiscated spatio-temporal model. In view of the remarkable performance in image-based primary object sementation, an easy method is to extend the image-based models to videos by considering spatial attributes and the additional temporal cues of primary video objects [22,30,31].

Such spatiotemporal attributes such as attractive appearance, better objectness, distinctive motion from its surroundings and frequent occurrence in the whole video mainly focus on foreground features and have attracted much attention from most models [2,32,33]. However, the background is actually symbiotic with the foreground and contains much connotative information. Thus, some models pay more attention to background cues, such as boundary connectivity [32,34] and surroundings [35], even including complex dynamic background modeling [36]. Naturally, this leads to several models [22,37] that consider both foreground and background cues to assist foreground segmentation. However, there exist two issues. On one hand, sometimes the complexity of primary objects renders these attributes insufficient (e.g., distractors share common visual attributes with targets), then these models may fail on certain videos in which the assumptions may not hold. On the other hand, these models either ignore the foreground/background or only utilize one to facilitate the other, which may miss some important cues and result in more ambiguities between the foreground and background.

Moreover, temporal coherence is an important issue for primary video object segmentation, and directly applying image-based algorithms to videos is vulnerable to inconsistent segmentation. To reduce such inconsistency, costly processing methods are usually adopted, such as object/trajectory tracking and sophisticated energy optimization models [22,31,37,38]. Particularly, pixel-wise optical flows are widely used to propagate information between adjacent frames. Unfortunately, optical flows are often inaccurate in case of sudden motion changes or occlusions, by which errors may be accumulated along time. Moreover, usually only correspondences in adjacent temporal windows are established, which may prevent long-term information being propagated more effectively.

Considering all these issues, this paper proposes a novel approach that effectively models the complementary nature of the foreground/background in primary video object segmentation and efficiently propagates information temporally within a neighborhood reversible flow (NRF). Firstly, the problem of primary object segmentation is formulated into a novel objective function that explicitly considers foreground and background cues as well as their complementary relationships. In order to optimize the function and obtain the foregroundness and backgroundness prediction, a complementary segmentation network (CSNet) with multi-scale feature fusion and foreground/background branching is proposed. Then, to enhance the temporal consistency of initial predictions, NRF is further proposed to establish reliable, non-local inter-frame correspondences. These two techniques constitute the spatial and temporal modules of the proposed framework, as shown in Figure 2.

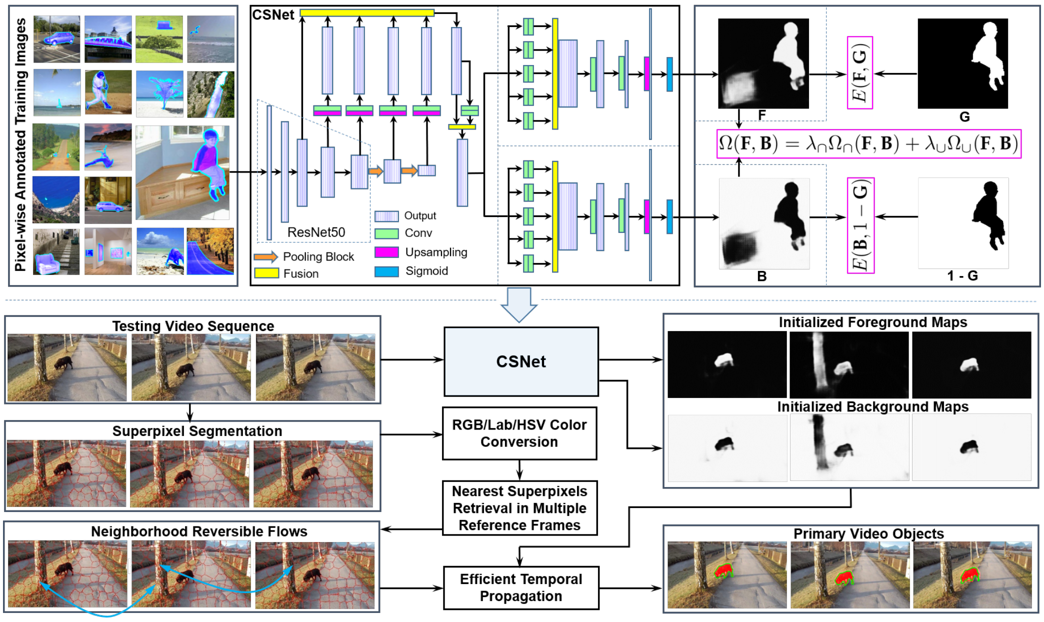
Pixel-wise Annotated Training Images
CSNet
ResNet50
Pooling Block
Fusion
Output
Conv
Upsampling
Sigmoid
F
G
$E(\mathbf{F},\mathbf{G})$
$\Omega(\mathbf{F},\mathbf{B}) = \lambda_{\cap}\Omega_{\cap}(\mathbf{F},\mathbf{B}) + \lambda_{\cup}\Omega_{\cup}(\mathbf{F},\mathbf{B})$
$E(\mathbf{B},1-\mathbf{G})$
B
1 - G
Testing Video Sequence
CSNet
Initialized Foreground Maps
Initialized Background Maps
Superpixel Segmentation
RGB/Lab/HSV Color Conversion
Nearest Superpixels Retrieval in Multiple Reference Frames
Neighborhood Reversible Flows
Efficient Temporal Propagation
Primary Video Objects

Figure 2. Framework of the proposed approach. The framework consists of two major modules. The spatial module trains CSNet to simultaneously initialize the foreground and background maps of each frame. This module operates on GPU to provide pixel-wise predictions for each frame. The temporal module constructs neighborhood reversible flow so as to propagate foregroundness and backgroundness along the most reliable inter-frame correspondences. This module operates on superpixels for efficient temporal propagation. Note that $E(\cdot)$ is the cross-entropy loss that enforce $F \to G$ and $B \to 1 - G$. The proposed complementary loss $\Omega(F, B)$ contains intersection loss $\Omega_{\cap}(F, B)$ and union loss $\Omega_{\cup}(F, B)$ for a complementary constraint. F, B and G are foreground, background and groundtruth, respectively. $\lambda_{\cap}$ and $\lambda_{\cup}$ are corresponding weights. Moreover, more details about CSNet are shown in Section 3.2.

In the spatial module, CSNet is trained on massive annotated images as an optimizer of the proposed complementary objective so as to simultaneously handle two complementary tasks, i.e., foregroundness and backgroundness estimation, with two separate branches. By using CSNet, we can obtain the initialized foreground and background maps on each individual frame. To efficiently and accurately propagate such spatial predictions between far-away frames, we further divide each frame into a set of superpixels and construct a neighborhood reversible flow so as to depict the most reliable temporal correspondences between superpixels in different frames. Within such flow, the initialized spatial foregroundness and backgroundness are efficiently propagated along the temporal dimension by solving a quadratic programming problem that has an analytic solution. In this manner, primary objects can efficiently pop out, and distractors can be further suppressed. Extensive experiments on three video datasets show that the proposed approach acts efficiently and achieves impressive performances compared with 22 state-of-the-art models.

This paper builds upon and extends our previous work in [39] with further discussion of the algorithm, analysis and expanded evaluations. We further formulate the segmentation problem into a new objective function based on the constraint relationship between foreground and background and optimize it using a new complementary deep network.

The main contributions of this paper include the following:

- We formulate the problem of primary object segmentation into a novel objective function based on the relationship between foreground and background and incorporate the objective optimization problem into end-to-end CNNs. By training specific CNNs, two dual tasks of foreground and background segmentation can be simultaneously addressed, and primary video objects can be segmented from complementary cues.
- We construct neighborhood reversible flow between superpixels which effectively propagates foreground and background cues along the most reliable inter-frame correspondences and leads to more temporally consistent results.
- Based on the proposed method, we achieve impressive performance compared with 22 image-based and video-based existing models, achieving state-of-the-art results.

In the rest of this paper, we first conduct a brief review of previous studies on primary/salient object segmentation in Section 2. Then, we present the technical details of the proposed spatial initialization module in Section 3 and temporal refinement module in Section 4. Experimental results are shown in Section 5. At last, we conclude with a discussion in Section 6.

2. Related Work

A great amount of the performance of primary video object segmentation is contributed by the good performance of each frame. In this section, we give a brief overview of recent works in salient object segmentation in images and primary/semantic object segmentation in videos.

2.1. Salient Object Segmentation in Images

Salient object segmentation in images is a research area that has been greatly developed in the past twenty years, in particular since 2007 [40].

Early approaches treated saliency object segmentation as an unsupervised problem and focused on low-level and mid-level cues, such as contrast [32,41], focusness [33], spatial properties [42,43], spectral information [44], objectness [35], etc. Most of the cues build upon foreground priors. For example, the widely used contrast prior believes that the salient regions present high contrast over the background in certain contexts [42,45], and the focusness prior considers that a salient object is often photographed in focus to attract more attention. From the opposite perspective, the background prior was first proposed by Wei et al. [43], who assumed the image boundaries are mostly background and built a saliency detection model based two background priors, i.e., boundary and connectivity. After that, some approaches [34,46–48] successively appeared. Unfortunately, these methods usually require a prior hypothesis about salient objects, and their performance heavily depend on the prior's reliability. Besides, the methods that only use purely low-level/mid-level cues face difficulties in detecting salient objects in complex scenes due to their unawareness of image content.

Recently, learning-based methods, especially deep networks methods (i.e., CNN-based models and FCN-based models), have attracted much attention because of their ability to extract high-level semantic information [4,13,49]. In [13], two neural networks, DNN-L and DNN-G, were proposed to respectively extract local features and conduct a global search for generating the final saliency map. In [5], Li and Yu introduced a neural network with fully connected layers to regress the saliency degree of each superpixel by extracting multiscale CNN features. While these CNN-based models with fully connected layers that operate at the patch level may result in blurry saliency maps, especially near the boundary of salient objects, in [50], fully convolutional networks considering pixel-level operations were applied for salient object segmentation. After that, various FCN-based salient object segmentation approaches were explored [51–53] and obtained impressive performance.

However, most of these methods focus on independent foreground or background features, and only several models [54,55] pay attention to both of them. To the best of our knowledge, few models explicitly model the constraint relationship between them, although it may be very helpful in complex scenes. Therefore, in this work, we simultaneously consider foreground and background cues as well as their complementary relationships and optimize their joint objective by using the powerful learning ability of deep networks.

2.2. Primary/Semantic Object Segmentation in Videos

Different from salient object segmentation in images, primary video object segmentation faces more challenges and criteria (e.g., spatiotemporal consistency) due to the additional temporal attributes.

Motion information (e.g., motion vectors, feature point trajectories and optical flow) is usually used in the spatiotemporal domain to facilitate primary/semantic video object segmentation and enhance the spatiotemporal consistency of segmentation results [56–58]. For example, Papazoglou and Ferrari [23] first initialized foreground maps with motion information and then refined them in the spatiotemporal domain so as to enhance the smoothness of foreground objects. Zhang et al. [21] used optical flow to track the evolution of object shape and presented a layered directed acyclic graph-based framework for primary video object segmentation. In a further step, Tsai et al. [38] utilized a multi-level spatial-temporal graphical model with the use of optical flow and supervoxels to jointly optimize segmentation and optical flow in an iterative scheme. The re-estimated optical flow (i.e., object flow) was used to maintain object boundaries and temporal consistency. Nevertheless, there still exist several issues. Firstly, some models [57,59,60] are built upon certain assumptions, for instance, that foreground objects should move differently from their surroundings in a good fraction of the video or should be spatially dense and change smoothly across frames in shapes and locations, which may fail on certain videos that contain complex scenarios in which these assumptions may not hold. Secondly, the pixel-wise optical flows are usually computed between adjacent frames since their similarity can offer more accurate flow estimation, while it is disadvantageous to obtain more valuable inter-frame (e.g., two far-away frames) cues since adjacent frames may not offer useful cues due to occlusion, blur and out-of-view, etc.

Recently, a number of approaches have attempted to address video object segmentation via deep neural networks. Due to lacking sufficient video data with per-frame pixel-level annotations, most of them exploit temporal information over image segmentation approaches for video segmentation. One popular thought is to calculate a kind of correspondence flow and propagate it in inter-frames [61–63]. In [61], based on optical flow, a spatio-temporal transformer GRU was proposed to temporally propagate labeling information between adjacent frames for semantic video segmentation. In [63], a deep feature flow was presented to propagate deep feature maps from key frames to other frames, which was jointly trained with video recognition tasks. Although these methods are helpful for transferring image-based segmentation networks to videos, the propagation flows are still limited by adjacent frames or training complexity.

Therefore, in our work, we enhance inter-frame consistency by constructing a neighborhood reversible flow (NRF) instead of an optical flow to efficiently and accurately propagate the initialized predictions between adjacent key frames, which is simple but effective for popping out the consistent and primary object in the whole video.

3. Initialization with Complementary CNNs

In this section, starting from the complementary peculiarity of foreground and background, we reformulate the problem of primary video object segmentation into a new objective function. Then, we design complementary CNNs to conduct deep optimization of the objective function and yield the initial foreground and background estimation.

3.1. Problem Formulation

Typically, a frame $\mathcal{I}$ consists of the foreground area $\mathcal{F}$ and the background area $\mathcal{B}$ with $\mathcal{F} \cap \mathcal{B} = \emptyset$ and $\mathcal{F} \cup \mathcal{B} = \mathcal{I}$, i.e., the foreground and background should be complementary in image space. Considering that foreground objects and background distractors usually have different visual characteristics (e.g., clear versus fuzzy edges, large versus small sizes, high versus low objectness), we can attack the problem of primary object segmentation at the frame $\mathcal{I}$ from a complementary perspective, estimating foreground and background maps, respectively. In this manner, the intrinsic characteristics of foreground and background regions can be better captured by two models with different focuses. Keeping this in mind, we propose the following formulation to explicitly consider foreground and background cues

$$\begin{aligned} \min_{\mathbb{W}_F, \mathbb{W}_B} \; & \mathcal{L}(\mathbf{F}, \mathbf{B}, \mathbf{G}) + \Omega(\mathbf{F}, \mathbf{B}), \\ \text{s.t. } & \phi_F(\mathcal{I}; \mathbb{W}_F) = \mathbf{F},\ \mathbf{F}(p) \in \{0, 1\}, \forall\, p \in \mathcal{I} \\ & \phi_B(\mathcal{I}; \mathbb{W}_B) = \mathbf{B},\ \mathbf{B}(p) \in \{0, 1\}, \forall\, p \in \mathcal{I}, \end{aligned} \tag{1}$$

where $\mathbf{F}$ and $\mathbf{B}$ are two binary matrices representing $\mathcal{F}$ and $\mathcal{B}$. $\mathbf{G}$ is the ground-truth map that equals 1 for a foreground pixel and 0 for a background pixel. $\mathbb{W}_F$ and $\mathbb{W}_B$ are two sets of parameters for the foreground and background prediction models ϕ_F and ϕ_B. For the sake of simplifications, the values of $\mathbf{F}$ and $\mathbf{B}$ are assumed to be in the range [0, 1]. The first term $\mathcal{L}(\mathbf{F}, \mathbf{B}, \mathbf{G})$ is the empirical loss defined as

$$\mathcal{L}(\mathbf{F}, \mathbf{B}, \mathbf{G}) = E(\mathbf{F}, \mathbf{G}) + E(\mathbf{B}, 1 - \mathbf{G}), \tag{2}$$

where $E(\cdot)$ is the cross-entropy loss that enforces $\mathbf{F} \to \mathbf{G}$ and $\mathbf{B} \to 1 - \mathbf{G}$. Ideally, salient objects and background regions can be perfectly detected by minimizing these two losses. However, errors always exist even when two extremely complex models are used. In this case, conflicts and unlabeled areas may arise in the predicted maps (e.g., both $\mathbf{F}$ and $\mathbf{G}$ equal 1 or 0 at the same location).

To reduce such errors, we refer to the constraint relationship $\mathcal{F} \cap \mathcal{B} = \emptyset$ and $\mathcal{F} \cup \mathcal{B} = \mathcal{I}$ and incorporate the complementary loss $\Omega(\mathbf{F}, \mathbf{B})$:

$$\Omega(\mathbf{F}, \mathbf{B}) = \lambda_\cap \Omega_\cap(\mathbf{F}, \mathbf{B}) + \lambda_\cup \Omega_\cup(\mathbf{F}, \mathbf{B}), \tag{3}$$

where $\Omega_\cap(\cdot)$ and $\Omega_\cup(\cdot)$ are two losses with non-negative weights $\lambda_\cap$ and $\lambda_\cup$ to encode the constraint $\mathcal{F} \cap \mathcal{B} = \emptyset$ and $\mathcal{F} \cup \mathcal{B} = \mathcal{I}$, respectively. Here, $\lambda_\cap$ and $\lambda_\cup$ are both set to 0.4. The intersection loss term $\Omega_\cap(\cdot)$ tries to minimize the conflicts between $\mathbf{F}$ and $\mathbf{B}$:

$$\Omega_\cap(\mathbf{F}, \mathbf{B}) = \frac{1}{\|\mathcal{I}\|} \sum_{p \in \mathcal{I}} (\mathbf{F}(p) \cdot \mathbf{B}(p))^{\sigma_\cap}, \tag{4}$$

where $\|\mathcal{I}\|$ indicates the number of pixels in the image $\mathcal{I}$, and p is a pixel with predicted foregroundness $\mathbf{F}(p)$ and backgroundness $\mathbf{B}(p)$. $\sigma_\cap$ is a positive weight to control the penalty of conflicts. The minimum value of (4) will be reached when $\mathbf{F}(p) \cdot \mathbf{B}(p) = 0$, implying that at least one map has zero prediction at every location.

Similarly, the union loss term $\Omega_{\cup}(\cdot)$ tries to maximize the complementary degree between **F** and **B**:

$$\Omega_{\cup}(\mathbf{F}, \mathbf{B}) = \frac{1}{\|\mathcal{I}\|} \sum_{p \in \mathcal{I}} (\mathbf{F}(p) + \mathbf{B}(p) - 1)^{\sigma_{\cup}}. \tag{5}$$

We can see that the minimum complementary loss can be reached when $\mathbf{F}(p) + \mathbf{B}(p) = 1$ (i.e., perfect complementary predictions). The parameter $\sigma_{\cup}$ is a positive weight to control the penalty of non-complementary predictions.

3.2. Deep Optimization with Complementary CNNs

Given the empirical loss (2) and the complementary loss (3), we can derive two models, $\phi_F(\cdot)$ and $\phi_B(\cdot)$, for per-frame initialization of the foreground and background maps by solving the optimization problem of objective functions (1). Toward this end, we need to first determine the form of the models and the algorithm for optimizing their parameters. Considering the impressive capability of convolutional neural networks (CNN), we propose to solve the optimization problem in a deep learning paradigm.

The architecture of the proposed CNN can be found in Figure 3, which starts from a shared trunk and ends up with two separate branches, i.e., a foreground branch and a background branch. The main configurations and details are shown in Table 1. For simplicity, only the foreground branch is illustrated in Table 1, as the background one adopts the same architecture. Note that this network simultaneously handles two complementary tasks as well as their relationships and is denoted as a complementary segmentation network (CSNet). The parameters of the shared trunk are initialized from the ResNet50 networks [64], which are used to extract low- to high-level features that are shared by foreground objects and background distractors. We remove the pooling layer and the fully connected layer after the `RELU` layer of `res5c` and introduce two pooling blocks (see Figure 3) to provide features from additional levels and reduce parameters. In order to integrate both the local and global contexts, we sum up different levels of feature outputs by layer `Res3`, `Res4` and `Res5` and two pooling blocks by appropriate up/down-sampling operations. After that, a residual block with a 3 × 3 `CONV` layer and a 1 × 1 `CONV` layer is used to post-process the integrated features as well as increase their nonlinearity. Finally, the shared trunk takes a 320 × 320 image as the input and outputs a 40 × 40 feature map with 512 channels.

Table 1. Main configurations for CSNet. Note that x and y are integers in the range [1, 5].

Type Name	Patch Size/Stride/Pad/Dilation/Group	Output Size
Conv1_pb1	3 × 3/1/1/1/32	10 × 10 × 256
Conv2_pb1	1 × 1/1/0/1/1	10 × 10 × 2048
Pool1	avg pool, 2 × 2, stride 2	5 × 5 × 2048
Conv3_pb1	3 × 3/1/1/1/32	5 × 5 × 256
Conv4_pb1	1 × 1/1/0/1/1	5 × 5 × 2048
Conv1_pb2	3 × 3/1/1/1/32	5 × 5 × 256
Conv2_pb2	1 × 1/1/0/1/1	5 × 5 × 2048
Pool2	avg pool, 2 × 2, stride 2	3 × 3 × 2048
Conv3_pb2	3 × 3/1/1/1/32	3 × 3 × 256
Conv4_pb2	1 × 1/1/0/1/1	3 × 3 × 2048
Interp1	bilinear upsampling	40 × 40 × 1024
Conv1	1 × 1/1/0/1/1	40 × 40 × 512
Interp2	bilinear upsampling	40 × 40 × 2048
Conv2	1 × 1/1/0/1/1	40 × 40 × 512
Interp3	bilinear upsampling	40 × 40 × 2048

Table 1. *Cont.*

Type Name	Patch Size/Stride/Pad/Dilation/Group	Output Size
Conv3	1 × 1/1/0/1/1	40 × 40 × 512
Interp4	bilinear upsampling	40 × 40 × 2048
Conv4	1 × 1/1/0/1/1	40 × 40 × 512
Conv5	3 × 3/1/1/1/32	40 × 40 × 256
Conv6	1 × 1/1/0/1/1	40 × 40 × 512
Conv7_xf	1 × 1/1/0/1/1	40 × 40 × 512
Conv8_yf	3 × 3/1/y/y/32	40 × 40 × 256
Conv9f	1 × 1/1/0/1/1	40 × 40 × 256
Conv10f	3 × 3/1/1/1/8	40 × 40 × 64
Deconv1f	3 × 3/4/1/1/1	161 × 161 × 1

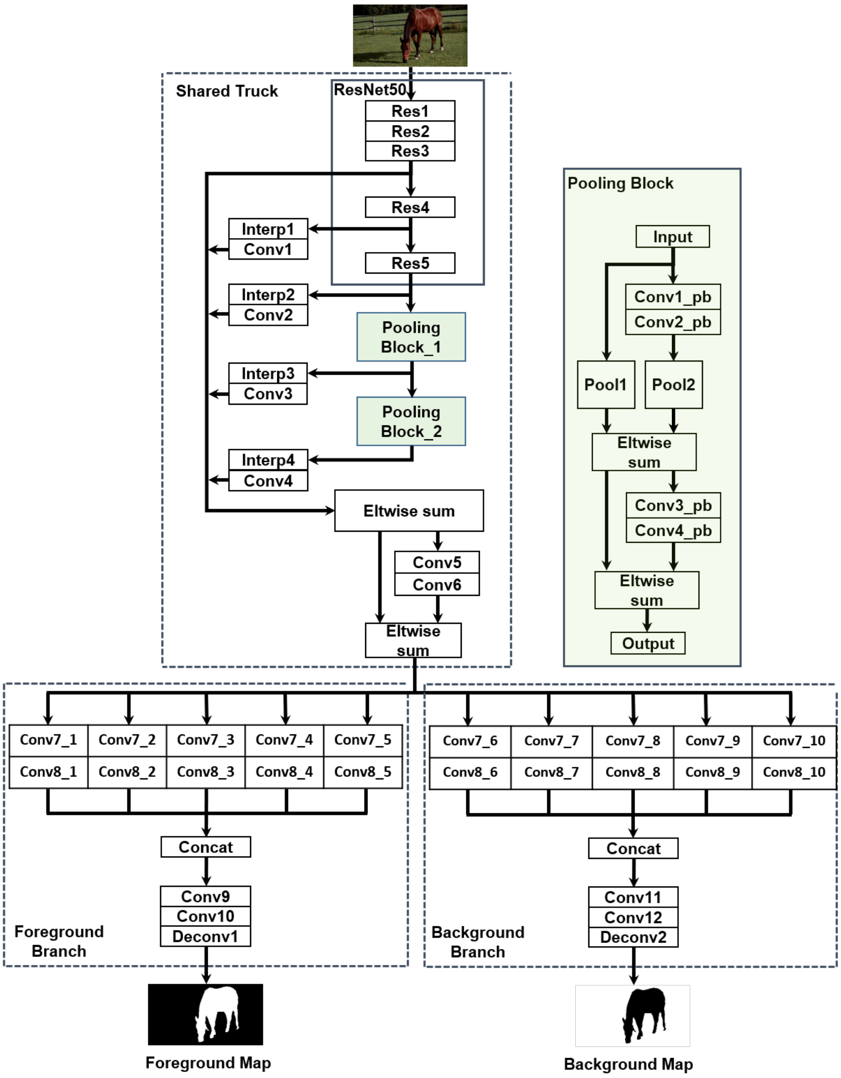

Figure 3. Architecture of the proposed CSNet. Note that layers `Res1` and `Res2/3/4/5` correspond to layers `conv1` and `conv2_x/3_x/4_x/5_x` in [64], respectively. More details are shown in Table 1.

After the shared trunk, the features are fed into two separate branches that address two complementary tasks, i.e., foreground and background estimation. Note that the two branches share the architecture with the input, but produce complementary outputs. In each branch, the shared features pass through a sequential of convolution blocks. These blocks all consist of 1 × 1 and 3 × 3 `CONVs`, but with different dilations. As such, we concatenate the output of each block to constitute feature maps at 40 × 40 resolution with 1280 channels.

These features, which have a wide range of spatial context and abstraction levels, are finally fed into several `CONV` layers for dimensional reduction and post-processing, and upsampled to produce output segmentation maps at size 161 × 161. With such designs, the foreground branch mainly focuses on detecting salient objects, while the background one suppresses distractors. In addition to the empirical loss defined in (2), two additional losses, (4) and (5), are also adopted to penalize the conflicts and complementary degree of the output maps for more accurate predictions.

In the training stage, we collect massive images with labeled salient objects from four datasets for image-based salient object detection [5,46,65,66]. We down-sample all images to 320 × 320 and their ground-truth saliency maps into 161 × 161. For the pretrained ResNet50 trunk, the learning rate is set to 5×10^{-7}, while for the two branches they are 5×10^{-6}. We train the network with a mini-batch of 4 images using the SGD optimizer with a momentum of 0.9 and a weight decay of 0.0005.

4. Efficient Temporal Propagation with Neighborhood Reversible Flow

The per-frame initialization of foregroundness and backgroundness can only provide a location prediction of the primary objects and background distractors at the spatial domain. However, the concept of primary objects is defined from a more global spatiotemporal perspective, not only salient in the intra-frame but also consistent in the inter-frame and throughout the whole video. As mentioned earlier, the primary video object should be spatiotemporally consistent, i.e., the saliency foreground regions should not change dramatically along the time dimension. This implies that there still exists a large gap between the frame-based initialization results and the video-based primary objects. Therefore, we need to further infer the primary objects that consistently pop out in the whole video [15] according to the spatiotemporal correspondence of visual signals. In this process, two key challenges need to be addressed, including:

(1) How to find the most reliable correspondences between various (nearby or far-away) frames;
(2) How to infer out the consistent primary objects based on spatiotemporal correspondences and the initialization results?

To address these two challenges, we propose a neighborhood reversible flow algorithm to find and propagate a neighborhood reversible subset from the inter-frames. Details of our solutions will be discussed in the following part of this section.

4.1. Neighborhood Reversible Flow

The proposed neighborhood reversible flow (NRF) propagates information along reliable correspondences established among several key frames of the video, thus preventing errors from accumulating quickly and involving larger temporal windows for more effective context exploitation. Instead of pixel-level correspondence, NRF operates on superpixels to achieve region-level matching and higher computational efficiency.

Given a video $\mathbb{V} = \{\mathcal{I}_u\}_{u=1}^{K}$, we first apply the SLIC algorithm [67] to divide a frame $\mathcal{I}_u$ into N_u superpixels, denoted as $\{\mathcal{O}_{ui}\}$. For each superpixel, we compute its average RGB, Lab and HSV colors as well as the horizontal and vertical positions. These features are then normalized into the same dynamic range $[0, 1]$.

Based on the features, we need to address two fundamental problems: (1) how to measure the correspondence between a superpixel $\mathcal{O}_{ui}$ from the frame $\mathcal{I}_u$ and a superpixel $\mathcal{O}_{vj}$ from the frame $\mathcal{I}_v$, and (2) which frames should be referred to for a given frame? Inspired by the concept of neighborhood reversibility in image search [68], we can compute the pair-wise $\ell 1$ distances between $\{\mathcal{O}_{ui}\}_{i=1}^{N_u}$ and $\{\mathcal{O}_{vj}\}_{j=1}^{N_v}$. After that, we denote the k-nearest neighbors of $\mathcal{O}_{ui}$ in the frame $\mathcal{I}_v$ as $\mathcal{N}_k(\mathcal{O}_{ui}|\mathcal{I}_v)$. As a consequence, two superpixels $\mathcal{O}_{ui}$ and $\mathcal{O}_{vj}$ are k-neighborhood reversible if they reside in the list of k-nearest neighbors of each other. That is,

$$\mathcal{O}_{ui} \in \mathcal{N}_k(\mathcal{O}_{vj}|\mathcal{I}_u) \text{ and } \mathcal{O}_{vj} \in \mathcal{N}_k(\mathcal{O}_{ui}|\mathcal{I}_v). \tag{6}$$

From (6), we find that the smaller the k, the more tightly two superpixels are temporally correlated. Therefore, the correspondence between $\mathcal{O}_{ui}$ and $\mathcal{O}_{vj}$ can be measured as

$$s_{ui,vj} = \begin{cases} \exp(-2k/k_0), & \text{if } k \leq k_0 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where k_0 is a constant to suppress weak flow, and k is a variable. A small k_0 will build sparse correspondences between $\mathcal{I}_u$ and $\mathcal{I}_v$ (e.g., $k_0 = 1$), while a large k_0 will cause dense correspondences. In this study, we empirically set $k_0 = 15$ and represent the flow between $\mathcal{I}_u$ and $\mathcal{I}_v$ with a matrix $\mathbf{F}_{uv} \in \mathbb{R}^{N_u \times N_v}$, in which the component at (i,j) equals $f_{ui,vj}$. Note that we further normalize $\mathbf{F}_{uv}$ so that each row sums up to 1. Considering the highly redundant visual content between adjacent frames, for each video frame $\mathcal{I}_u$, we pick up its adjacent keyframes $\{\mathcal{I}_t | t \in \mathbb{T}_u\}$ to ensure sufficient variation in content and depict reliable temporal correspondences. In this paper, we refer to the interval d_k of annotated video frames, which usually contain the most critical information of the whole video, to determine the interval of adjacent keyframes. Later, we estimate the flow matrixes between a frame $\mathcal{I}_u$ and the frames $\{\mathcal{I}_t | t \in \mathbb{T}_u\}$, where $\mathbb{T}_u$ can be empirically set to $\{u - 2 \times d_k, u - d_k, u + d_k, u + 2 \times d_k\}$.

4.2. Temporal Propagation of Spacial Features

The flow $\{\mathbf{F}_{uv}\}$ depicts how superpixels in various frames are temporally correlated, which can be used to further propagate the spatial foregroundness and backgroundness. Typically, such temporal refinement can obtain impressive performance by solving a complex optimization problem with constraints like spatial compactness and temporal consistency. However, the time cost will also grow surprisingly high [20]. Considering the requirement of efficiency in many real-world applications, we propose to minimize an objective function that has an analytic solution. For a superpixel $\mathcal{O}_{ui}$, its foregroundness x_{ui} and backgroundness y_{ui} can be initialized as

$$x_{vj} = \frac{\sum_{p \in \mathcal{O}_{ui}} \mathbf{X}_u(p)}{|\mathcal{O}_{ui}|},\ y_{ui} = \frac{\sum_{p \in \mathcal{O}_{ui}} \mathbf{Y}_u(p)}{|\mathcal{O}_{ui}|}, \tag{8}$$

where p is a pixel with foregroundness $\mathbf{X}_u(p)$ and backgroundness $\mathbf{Y}_u(p)$. $|\mathcal{O}_{ui}|$ is the area of $\mathcal{O}_{ui}$. For the sake of simplification, we represent the foregroundness and backgroundness scores of all superpixels in the uth frame with column vectors $\mathbf{x}_u$ and $\mathbf{y}_u$, respectively. As a result, we can propagate such scores from $\mathcal{I}_v$ to $\mathcal{I}_u$ according to $\mathbf{F}_{uv}$:

$$\mathbf{x}_{u|v} = \mathbf{F}_{uv}\mathbf{x}_v,\ \mathbf{y}_{u|v} = \mathbf{F}_{uv}\mathbf{y}_v,\ \ \forall v \in \mathbb{T}_u. \tag{9}$$

After the propagation, the foregroundness vector $\hat{\mathbf{x}}_u$ and backgroundness vector $\hat{\mathbf{y}}_u$ can be refined by solving

$$\begin{aligned} \hat{\mathbf{x}}_u &= \arg\min_{\mathbf{x}}\ \|\mathbf{x} - \mathbf{x}_u\|_2^2 + \lambda_c \sum_{v \in \mathbb{T}_u} \|\mathbf{x} - \mathbf{x}_{u|v}\|_2^2, \\ \hat{\mathbf{y}}_u &= \arg\min_{\mathbf{y}}\ \|\mathbf{y} - \mathbf{y}_u\|_2^2 + \lambda_c \sum_{v \in \mathbb{T}_u} \|\mathbf{y} - \mathbf{y}_{u|v}\|_2^2, \end{aligned} \tag{10}$$

where λ_c is a positive constant whose value is empirically set to 0.5. Note that we adopt only the $\ell 2$ norm in (10) so as to efficiently compute an analytic solution

$$\begin{aligned} \hat{\mathbf{x}}_u &= \frac{1}{1 + \lambda_c \cdot |\mathbb{T}_u|}\left(\mathbf{x}_u + \lambda_c \sum_{v \in \mathbb{T}_u} \mathbf{F}_{uv}\mathbf{x}_v\right), \\ \hat{\mathbf{y}}_u &= \frac{1}{1 + \lambda_c \cdot |\mathbb{T}_u|}\left(\mathbf{y}_u + \lambda_c \sum_{v \in \mathbb{T}_u} \mathbf{F}_{uv}\mathbf{y}_v\right). \end{aligned} \tag{11}$$

By observing (9) and (11), we find that the propagation process is actually calculating the average foregroundness and backgroundness scores within a local temporal slice under the guidance of neighborhood reversible flow. After the temporal propagation, we turn superpixel-based scores into pixel-based ones according to

$$\mathbf{M}_u(p) = \sum_{i=1}^{N_u} \delta(p \in \mathcal{O}_{ui}) \cdot \hat{x}_{ui} \cdot (1 - \hat{y}_{ui}), \tag{12}$$

$$\min_{\mathbf{x}} \ \|\mathbf{x} - \mathbf{x}_u\|_2^2 + \lambda_c \sum_{v \in \mathbb{T}_u} \|\mathbf{x} - \mathbf{F}_{uv}\mathbf{x}_v\|_2^2 \tag{13}$$

where $\mathbf{M}_u$ is the importance map of $\mathcal{I}_u$ that depict the presence of primary objects. $\delta(p \in \mathcal{O}_{ui})$ is an indicator function, which equals 1 if $p \in \mathcal{O}_{ui}$ and 0 otherwise. Finally, we calculate an adaptive threshold which equals the 20% of the maximal pixel importance to binarize each frame, and a morphological closing operation is then performed to fill in the black area in the segmented objects.

5. Experiments

In this section, we first illustrate experimental settings about datasets and evaluation metrics in Section 5.1. Then, based on the datasets and metrics, we quantitatively compare our primary video object segmentation method with 22 state-of-the-art approaches in Section 5.2. After that, in Section 5.3, we further demonstrate the effectiveness of our approach by offering more detailed exploration and dissecting various parts of our approach as well as running time and failure cases.

5.1. Experimental Settings

We test the proposed approach on three widely used video datasets; their ways of defining primary video objects are different. Details of these datasets are described as follows:

(1) **SegTrack V2** [59] is a classic dataset in video object segmentation that is frequently used in many previous works. It consists of 14 densely annotated video clips with 1066 frames in total. Most primary objects in this dataset are defined as ones with *irregular motion patterns*.

(2) **Youtube-Objects** [1] contains a large amount of Internet videos, and we adopt its subset [69] that contains 127 videos with 20,977 frames. In these videos, 2153 keyframes are sparsely sampled and manually annotated with pixel-wise masks according to the video tags. In other words, primary objects in **Youtube-Objects** are defined from the perspective of *semantic attributes*.

(3) **VOS** [15] contains 200 videos with 116,093 frames. On 7467 uniformly sampled keyframes, all objects are pre-segmented by 4 subjects, and the fixations of another 23 subjects are collected in eye-tracking tests. With these annotations, primary objects are automatically selected as the ones whose average fixation densities over the whole video fall above a predefined threshold. If no primary objects can be selected with the predefined threshold, objects that receive the highest average fixation density will be treated as the primary ones. Different from **SegTrack V2** and **Youtube-Objects**, primary objects in **VOS** are defined from the perspective of *human visual attention*.

On these three datasets, the proposed approach, denoted as **CSP**, is compared with 22 state-of-the-art models for primary and salient object segmentation, including: **RBD** [34], **SMD** [70], **MB+** [48], **DRFI** [2], **BL** [71], **BSCA** [47], **MST** [72], **ELD** [4], **MDF** [5], **DCL** [53], **LEGS** [13], **MCDL** [7] and **RFCN** [73], **ACO** [22], **NLC** [74], **FST** [23], **SAG** [57], **GF** [60], **PN+** [75], **DFI** [76], **FSal** [77] and **PFS** [78].

In the comparisons, we adopt two sets of evaluation metrics, including the intersection-over-union (IoU) and the precision-recall-F_β. Similar to [15], the precision, recall and IoU scores are first computed on each video and finally averaged over the whole dataset so as to generate the mean average precision (mAP), mean average recall (mAR) and mean average IoU (mIoU). In this manner, the influence of short and long videos can be balanced. Furthermore, a unique F_β score can be obtained based on mAR, mAP and a parameter β, the square of which is set as 0.3 to emphasize precision more than recall in the evaluation.

5.2. Comparisons with State-of-the-Art Models

The performances of our approach and 22 state-of-the-art models on three video datasets are shown in Table 2. Some representative results of our approach are demonstrated in Figure 4. From Table 2, we find that on **Youtube-Objects** and **VOS**, for such larger datasets, our approach obtains the best F_β and mIoU scores, while on **SegTrack V2**, our approach ranks the second place (worse than **NLC**). This can be explained by the fact that **SegTrack V2** contains only 14 videos, among which most primary objects have irregular motion patterns. Such videos often perfectly meet the assumption of **NLC** on motion patterns of primary objects, making it the best approach on **SegTrack V2**. However, when the scenarios being processed extend to datasets such as **VOS** that are constructed without such "constraints" on motion patterns, the performance of **NLC** drops sharply, as its assumption may sometimes fail (e.g., **VOS** contains many videos only with static primary objects and distractors as well as slow camera motion; see Figure 4). These results further validate that it is quite necessary to conduct comparisons on larger datasets with daily videos (such as **VOS**) so that models with various kinds of assumptions can be fairly evaluated.

Table 2. Performances of our approach and 22 models. Bold and underline indicate the 1st and 2nd performance in each column.

Models	SegTrackV2 (14 Videos)				Youtube-Objects (127 Videos)				VOS (200 Videos)			
	mAP	mAR	F_β	mIoU	mAP	mAR	F_β	mIoU	mAP	mAR	F_β	mIoU
DRFI [2]	0.464	0.775	0.511	0.395	0.542	0.774	0.582	0.453	0.597	0.854	0.641	0.526
RBD [34]	0.380	0.709	0.426	0.305	0.519	0.707	0.553	0.403	0.652	0.779	0.677	0.532
BL [71]	0.202	**0.934**	0.246	0.190	0.218	**0.910**	0.264	0.205	0.483	**0.913**	0.541	0.450
BSCA [47]	0.233	0.874	0.280	0.223	0.397	0.807	0.450	0.332	0.544	0.853	0.594	0.475
MB+ [48]	0.330	0.883	0.385	0.298	0.480	0.813	0.530	0.399	0.640	0.825	0.675	0.532
MST [72]	0.450	0.678	0.488	0.308	0.538	0.698	0.568	0.396	0.658	0.739	0.675	0.497
SMD [70]	0.442	0.794	0.493	0.322	0.560	0.730	0.592	0.424	0.668	0.771	0.690	0.533
MDF [5]	0.573	0.634	0.586	0.407	0.647	0.776	0.672	0.534	0.601	0.842	0.644	0.542
ELD [4]	0.595	0.767	0.627	0.494	0.637	0.789	0.667	0.531	0.682	0.870	0.718	0.613
DCL [53]	0.757	0.690	0.740	0.568	0.727	0.764	0.735	0.587	0.773	0.727	0.762	0.578
LEGS [13]	0.420	0.778	0.470	0.351	0.549	0.776	0.589	0.450	0.606	0.816	0.644	0.523
MCDL [7]	0.587	0.575	0.584	0.424	0.647	0.613	0.638	0.471	0.711	0.718	0.713	0.581
RFCN [73]	0.759	0.719	0.749	0.591	0.742	0.750	0.744	0.592	0.749	0.796	0.760	0.625
NLC [74]	**0.933**	0.753	**0.884**	**0.704**	0.692	0.444	0.613	0.369	0.518	0.505	0.515	0.364
ACO [22]	0.827	0.619	0.767	0.551	0.683	0.481	0.623	0.391	0.706	0.563	0.667	0.478
FST [23]	0.792	0.671	0.761	0.552	0.687	0.528	0.643	0.380	0.697	0.794	0.718	0.574
SAG [57]	0.431	0.819	0.484	0.384	0.486	0.754	0.529	0.397	0.538	0.824	0.585	0.467
GF [60]	0.444	0.737	0.489	0.354	0.529	0.722	0.563	0.407	0.523	0.819	0.570	0.436
PN+ [75]	0.734	0.633	0.708	0.577	0.759	0.690	0.742	0.559	**0.808**	0.882	0.824	**0.754**
DFI [76]	0.711	0.663	0.699	0.579	0.729	0.799	0.744	0.617	0.792	0.906	0.816	0.746
FSal [77]	0.645	0.725	0.662	0.561	0.344	0.358	0.347	0.170	0.313	0.330	0.317	0.152
PFS [78]	0.604	0.581	0.598	0.410	0.736	0.704	0.728	0.549	0.692	0.639	0.679	0.471
CSP	0.789	0.778	0.787	0.669	**0.778**	0.820	**0.787**	**0.675**	0.805	0.910	**0.827**	0.747

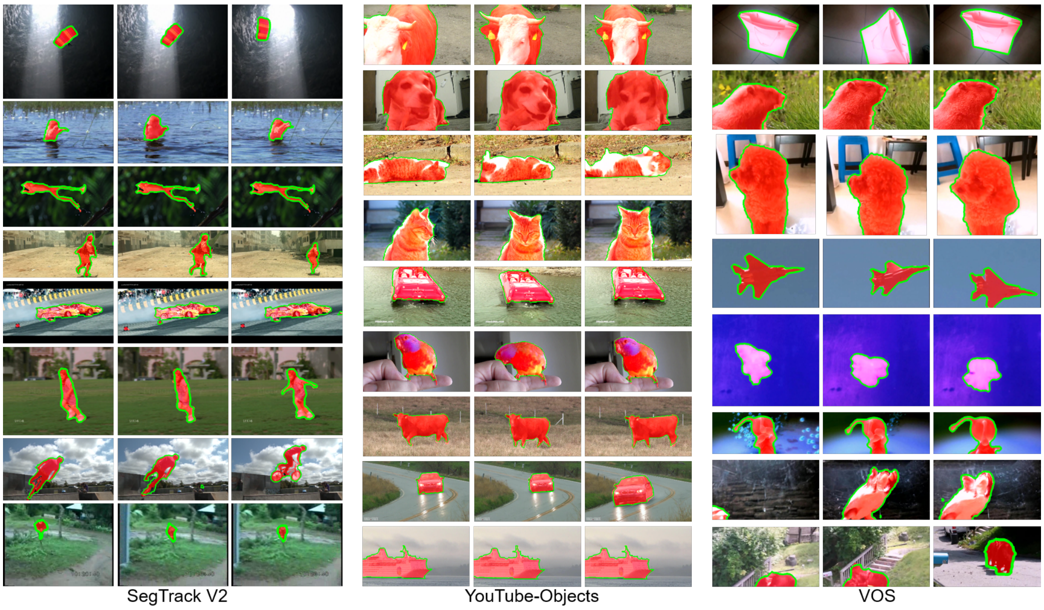

Figure 4. Representative results of our approach. Red masks are the ground truth, and green contours are the segmented primary objects.

Moreover, there exist some approaches (**BL** and **MB+**) on the three datasets that outperform our approach in recall, and some other approaches (**NLC**, **ACO**, **PN+** and **FST**) may achieve better or comparable precision with our approach on **SegTrack V2**. However, the other evaluation scores of the approaches are much worse than our method on the three datasets. That is, none of these approaches simultaneously outperforms our approach in both recall and precision, so our approach often has better overall performance, especially on larger datasets. This may imply that the proposed approach is more balanced than previous works. By analyzing the results on the three datasets, we find that this phenomenon may be caused by the conduction of complementary tasks in CSNet. By propagating both foregroundness and backgroundness, some missing foreground information may be retrieved, while the mistakenly popped-out distractors can be suppressed again, leading to balanced recall and precision.

From Table 2, we also find that there exist inherent correlations between salient image object detection and primary video object segmentation. As shown in Figure 4, primary objects are often the most salient ones in many frames, which explains why deep models such as **ELD**, **RFCN** and **DCL** outperform many video-based models such as **NLC**, **SAG** and **GF**. However, there are several key differences between the two problems. First, primary objects may not always be the most salient ones in all frames (as shown in Figure 1). Second, inter-frame correspondences provide additional cues for separating primary objects and distractors, which depict a new way to balance recall and precision. Third, primary objects may be sometimes close to the video boundary due to camera and object motion, making the boundary prior widely used in many salient object detection models not valid (e.g., the bear in the last row of the last column of Figure 4). Last but not least, salient object detection needs to distinguish a salient object from a fixed set of distractors, while primary object segmentation needs to consistently pop out the same primary object from a varying set of distractors. To sum up, primary video object segmentation is a more challenging task that needs to be further explored from the spatiotemporal perspective.

5.3. Detailed Performance Analysis

Beyond performance comparison, we also conduct several experiments on **VOS**, the largest one of the three datasets, to find out how the proposed approach works in segmenting primary video objects. Moreover, an additional metric, i.e., temporal stability measure T [14], is applied to evaluating the relevant aspect in primary video object segmentation in addition to the aforementioned four metrics. After all, mIoU only measures how well the pixels of two masks match, while F_β measures the accuracy of contours. None of them consider the temporal aspect. However, video object segmentation is conducted in spatiotemporal dimensions. Therefore, the additional temporal stability measure is a appropriate choice to evaluate the temporal consistency of segmentation results. The main quantifiable results can be found in Table 3. In Table 3, the first group is our previous work in [39], and the second group is our current work extended from [39]. In order to illustrate the effect of each component in our approach, the two groups of tests are based on the same parameters, except for the last case, R-Init. + NRFp, which is the final test result obtained generally by using some data argumentations and parameter adjustments on the base of the case R-Init. + NRF.

Table 3. Detailed performances of our approaches. The first test group is our previous work in [39], and the second group is our current work. V-Init/R-Init.: corresponding results initialized by previous/current network. FG (FGp)/BG (BGp): foreground/background estimation with (without) the constraint of complementary loss. NRF (NRFp): neighborhood reversible flow (with multi-test). CE: cross-entropy. Comple.: complementary loss. mT: mean temporal stability metric, the smaller the better. Bold and underline indicate the 1st and 2nd performance in each column.

Test Cases	Backbone	Objective				Evaluation				
	VGG16/ResNet50	CE	Comple.	NRF	Multi-Test	mAP	mAR	F_β	mIoU	mT
V-Init. FG	VGG16	✓				0.750	0.879	0.776	0.684	0.117
V-Init. BG	VGG16	✓				0.743	0.884	0.771	0.680	0.117
V-Init. (FG + BG)	VGG16	✓				0.791	0.834	0.800	0.689	0.121
V-Init. + NRF	VGG16	✓		✓		0.789	0.870	0.806	0.710	0.109
R-Init. FG	ResNet50	✓				0.763	0.901	0.791	0.710	0.128
R-Init. BG	ResNet50	✓				0.764	0.899	0.791	0.711	0.128
R-Init. (FG + BG)	ResNet50	✓				0.808	0.863	0.820	0.724	0.127
R-Init. FGp	ResNet50	✓	✓			0.768	0.925	0.800	0.726	0.124
R-Init. BGp	ResNet50	✓	✓			0.763	**0.927**	0.796	0.723	0.124
R-Init. (FGp + BGp)	ResNet50	✓	✓			**0.814**	0.883	**0.829**	0.739	0.122
R-Init. + NRF	ResNet50	✓	✓	✓		0.803	0.901	0.824	0.739	0.108
R-Init. + NRFp	ResNet50	✓	✓	✓	✓	0.805	0.910	0.827	**0.747**	**0.097**

5.3.1. Performance of Complementary CNNs

In this section, some detail analysis will be given to further verify the effectiveness of the proposed complementary CNN branches and complementary loss.

Impact of two complementary branches. To explore the impact of two complementary network branches, we evaluated the foreground maps and background maps initialized by the two complementary branches, as well as their fusion maps. As shown in Table 3, in the first group the evaluation scores of case V-Init. FG and V-Init. BG are equally matched for their same branch structure, while the ones of their fusion maps increased to different degrees, which suggests that the complementary characteristics of initialized foreground maps and background maps can contribute to and constrain each other to generate more accurate prediction. Then what will happen if we abandon the background branch? To this end, we conducted two additional experiments in our previous work [39]. First, if we cut down the background branch and retrain only the foreground branch, the final performance decreases by about 0.9%. Second, if we re-train a network with two foreground branches, the final F_β and mIoU scores decrease from 0.806 to 0.800 and 0.710 to 0.700, respectively. These experiments indicate that, beyond learning more weights, the background branch does learn some useful cues that are ignored by the foreground branch, which are expected to be high-level visual patterns of typical background distractors. These results also validate the idea of training deep networks by simultaneously handling two complementary tasks.

Therefore, network structures with two similar branches are still adopted in this extension work. What is different is that the new network structure is assisted with more designs based on the deeper and more effective ResNet50 instead of simple VGG16. From Table 3, we can find that the initialized results are distinctly improved when the backbone VGG16 is replaced by ResNet50. The aforementioned four evaluation scores are all increased, e.g., the F_β and mIoU scores increase from 0.776 to 0.791 and 0.684 to 0.710, respectively, although the temporal stability performance is affected. This reveals the better performance of our new network structure, and at the same time, hints at the fact that a favourable per-frame initialization cannot stand for a good video initialization because of the temporal consistency attribute in video. Thus, it is necessary to conduct optimization in thhe temporal dimension, which will be explained in the next subsection.

Effect of the complementary loss. Except for the specific network, another main difference is that the two complementary CNN bracnches in CSNet are also constrained by our complementary loss. To verify the effectiveness, we optimize two sets of foreground and background prediction models based on the new network structure, one with the constraint of the penalty term in the objective function, and the other without. Based on the two sets of models, we can initialize a foreground and a background map for each video frame. The quantitative evaluations of initialization results respectively correspond to the cases R-Init. FGp/BGp/(FGp + BGp) and the cases R-Init. FG/BG/(FG + BG) in Table 3. From Table 3 we can find that, compared to the predictions without the penalty term constraint, the foreground and background models with the additional complementary loss can achieve better performance in predicting both foreground maps and background maps, shown as better F_β and mIoU scores. Moreover, some visual examples are shown in Figure 5. Obviously, if we only use the empirical loss (2), some background regions may be wrongly classified into foreground (e.g., the first three columns in Figure 5), while some foreground details may be missed (e.g. the last three columns in Figure 5). By incorporating the additional complementary loss (3), these mistakes can be fixed (see Figure 5d,e). Thus, the complementary loss is effective for boundary localization and suppressing background distractors. These results validate the effectiveness of handling two complementary tasks with explicit consideration of their relationships.

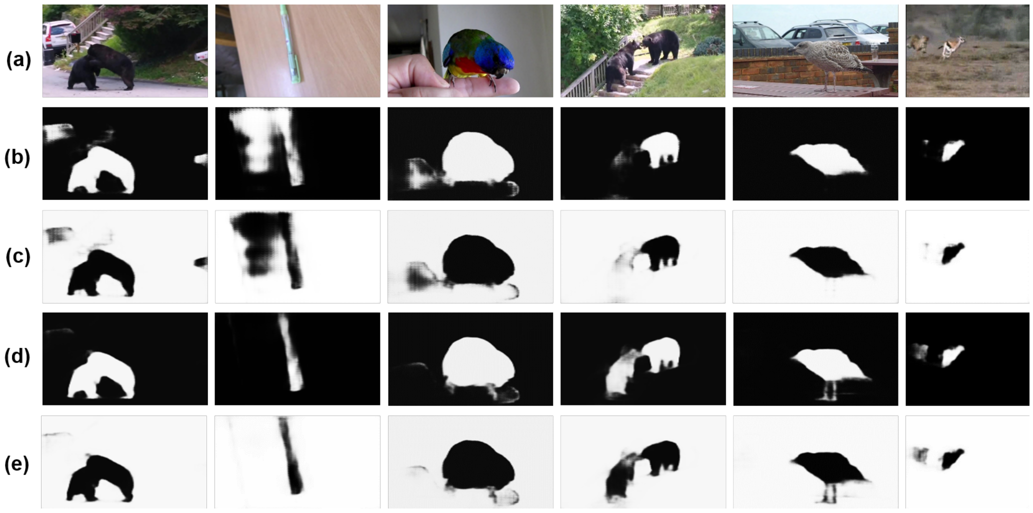

Figure 5. Foreground and background maps initialized by CSNet as well as their interaction and union maps, (**a**) video frames, (**b**) foreground maps and (**c**) background maps generated by CSNet without the complementary loss. (**d**) Foreground maps and (**e**) background maps generated by CSNet with the complementary loss.

Combining the two differences, the F_β and mIoU scores of initialization results output by our previous network CCNN (the case V-Init. (FG + BK)) increase by about 3.6% and 7.2%, i.e., from 0.800 to 0.829 and 0.689 to 0.739, respectively, with the increased mAP and mAR scores. This also means that the combination of complementary loss and two complementary branches of foreground and background are valid and ingenious.

In particular, the complementary CNN branches in our two networks both show impressive performance in predicting primary video objects over the other 6 deep models when their pixel-wise predictions are directly evaluated on **VOS**. By analyzing the results, we find that this may be caused by two reasons: (1) using more training data, and (2) simultaneously handling complementary tasks, whose effectiveness is just verified. To explore the first reason, we retrained the CCNN on the same MSRA10K dataset used by most deep models. In this case, the F_β (mIoU) scores of the foreground and background maps predicted by CCNN decrease to 0.747 (0.659) and 0.745 (0.658), respectively. Note that both branches still outperform **RFCN** on **VOS** in terms of mIoU (but F_β is slightly worse).

5.3.2. Effectiveness of Neighborhood Reversible Flow

Through the above complementary network branches, the salient foreground and background maps in the intra-frame are well obtained, while the initialization operation cannot ensure the temporal consistency of segmented objects, e.g., the initialized predictions by CSNet outperform the ones by the CCNN in term of mAP, mAR, F_β and mIoU but become inferior in temporal stability mT. Thus, the thought of improving the temporal relationship was proposed in Section 4, i.e., finding and propagating the reliable inter-frame correspondences by applying the neighborhood reversible flow to make the consistent salient subsets enhanced and accidental distractors suppressed. Consequently, the final primary video objects with spatiotemporal consistency are yielded.

Effectiveness of Neighborhood Reversible Flow. To prove this thought, we compare the initialized results by CCNN (CSNet) with the optimized results (V-Init.(R-Init.) + NRF) by neighborhood reversible flow. As shown in Table 3, the temporal stability measure mT of optimization results in CCNN (CSNet) cases decrease from 0.121 (0.122) to 0.109 (0.108) compared with the initialized predictions (V-Init. (FG + BK), R-Init. (FGp + BKp)). At the same time, the other evaluation scores are also improved, e.g., the mIoU score increases from 0.689 to 0.710. The superiority will become more obvious if we directly compare V-Init. + NRF with V-Init. FG, i.e., conduct the fusion operation on foreground and background in the process of neighborhood reversible flow just like we really do. This means that by propagating the neighborhood reversible flow, the spatial subsets of primary objects in intra-frame can be refined from a temporal perspective, and the inter-frame temporal consistency can be enhanced. Finally, the primary video objects with favourable spatiotemporal consistency can pop out. As shown in Figure 6, the primary objects in most video frames are initialized as the horce, while the objects that only lasts for a short while are mistakenly classified into foreground due to their spatial saliency in certain frames. Fortunately, the distractors are well suppressed by the optimization of neighborhood reversible flow (see the third rows in Figure 6). Thus, via propagating salient cues in inter-frames, background objects could be effectively suppressed, only preserving the real primary one.

To further demonstrate the effectiveness of neighborhood reversible flow, we tested our approach with two new settings based on the CCNN. In the first setting, we replaced the correspondence from Equation (7) with the cosine similarity between superpixels. In this case, the F_β and mIoU scores of our approach on **VOS** drop to 0.795 and 0.696, respectively. Such performance is still better than the initialized foreground maps but worse than the performance when using the neighborhood reversible flow (F_β = 0.806, mIoU = 0.710). This result indicates the effectiveness of neighborhood reversibility in temporal propagation.

In the second setting, we set $\lambda_c = +\infty$ in Equation (10), implying that primary objects in a frame are solely determined by the foreground and background propagated from other frames. When the spatial predictions of each frame are actually ignored in the optimization process, the F_β (mIoU) scores of our approach on **VOS** only decrease from 0.806 (0.710) to 0.790 (0.693), respectively. This result proves that the inter-frame correspondences encoded in the neighborhood reversible flow are quite reliable for efficient and accurate propagation along the temporal dimension.

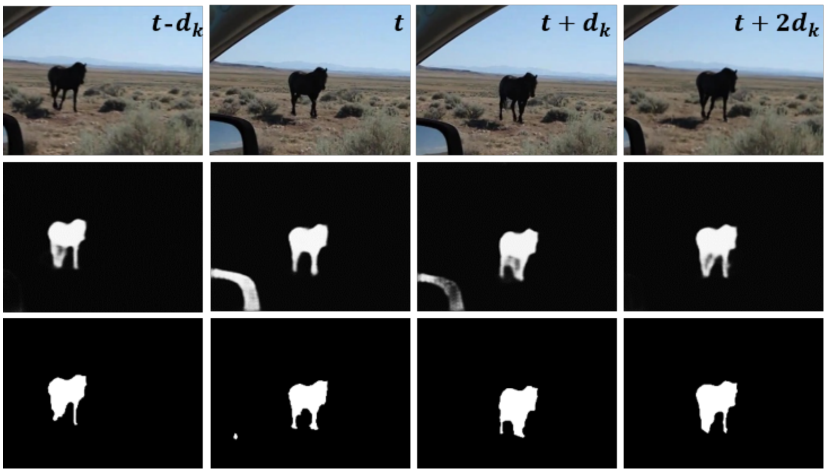

Figure 6. Performance of the proposed neighborhood reversible flow. The first row is the video sequences from **Youtube-Objects** [1], the second row is the corresponding initialized foreground maps, and the third row is the optimized results by neighborhood reversible flow.

It is worth mentioning that in the previous initialization process, the predictions are all pixel-wise, while the temporal optimization via neighborhood reversible flow is conducted on superpixel wise foreground/background maps in order to reduce time consumption, i.e., the predictions need to be converted from the pixel to superpixel and finally converted to pixel. However, the superpixel-wise predictions are relatively coarse and may affect the following process. To explore this effect, we converted the foreground/background maps and their fusion maps in cases R-Init. FGp/BKp from pixel-wise to superpixel-wise, as shown in Table 4. Fortunately, both F_β and mIoU scores of foreground (background) maps only slightly decrease by 0.003 (0.004), and the mT scores increase by 0.009, while the negative effect on fusion maps mainly manifests in mT scores, which can be improved by the propagation of neighborhood reversible flow. Thus, the trade-off is worthy. This also hints at the important effect of neighborhood reversible flow on temporal stability or consistency.

Table 4. Performance of superpixel-wise initialization by CSNet on **VOS**. FGp: foreground branch, BKp: background branch. Sup. is short for superpixel.

Step	mAP	mAR	F_β	mIoU	T
R-Init. FGp (Sup.)	0.765	0.924	0.797	0.723	0.133
R-Init BKp (Sup.)	0.759	0.926	0.792	0.719	0.133
R-Init FGp + BKp (Sup.)	0.814	.881	0.829	0.738	0.129

Parameter setting. In the experiment based on CCNN, we smoothly varied two key parameters used in NRF, including the k_0 in constructing neighborhood flow and the λ_c that controls the strength of temporal propagation. As shown in Figure 7, a larger k_0 tends to bring slightly better performance, while our approach performs the best when $\lambda_c = 0.5$. In these experiments, we set $k_0 = 15$ and $\lambda_c = 0.5$ when constructing the neighborhood reversible flow.

Selection of color spaces. In constructing the flow, we represented each superpixel with three color spaces. As shown in Table 5, a single color space performs slightly worse than their combinations. Actually, using multiple color spaces has been proved to be useful in detecting salient objects [2], as multiple color spaces make it possible to assess temporal correspondences from several perspectives with a small increase in time cost. Therefore, we choose to use RGB, Lab and HSV color spaces in characterizing a superpixel.

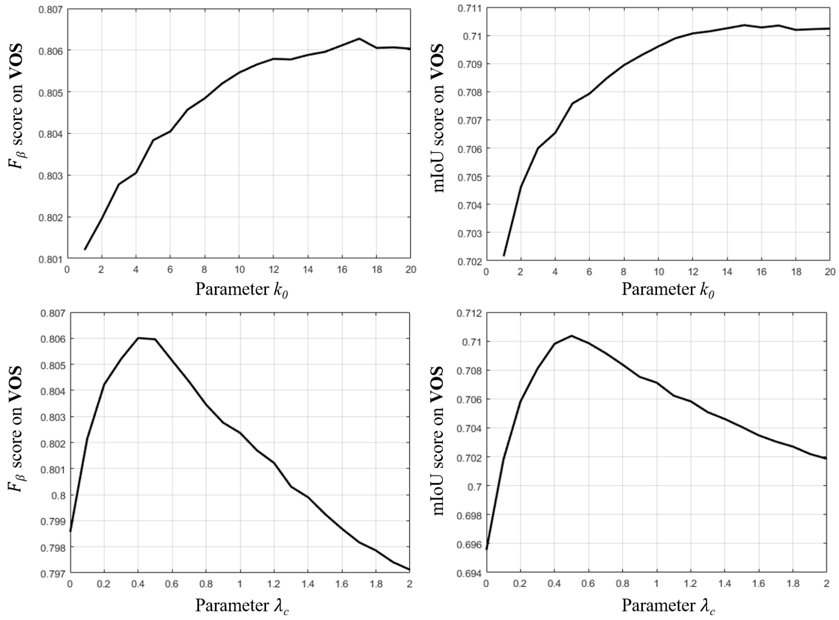

Figure 7. Influence of parameters k_0 and λ_c on our approach.

Table 5. Performance of our CCNN-based approach on **VOS** when using different color spaces in constructing neighborhood reversible flow.

Color Space	mAP	mAR	F_β	mIoU
RGB	0.785	0.862	0.801	0.703
Lab	0.786	0.860	0.802	0.702
HSV	0.787	0.866	0.804	0.707
RGB + Lab + HSV	0.789	0.870	0.806	0.710

5.3.3. Running Time

We tested the speed of the proposed approach with a 3.4 GHz CPU (only using single thread) and an NVIDIA TITAN Xp GPU (without batch processing). The average time costs of each key step of our approach in processing 400 × 224 frames are shown in Table 6. Note that the majority of the implementation runs on the Matlab platform, with several key steps written in C (e.g., superpixel segmentation and feature conversion between pixels and superpixels). We find that our approach takes only 0.20 s to process a frame if not using multi-test, and no more than 0.75 s even using multi-test, which is much faster than many video-based models (e.g., 19.0 s for NLC, 6.1 s for ACO, 5.8 s for FST, 5.4 s for SAG and 4.7 s for GF). This may be caused by the fact that we only build correspondences on superpixels with the neighborhood reversibility, which is very efficient. Moreover, we avoid using complex optimization objectives and constraints. Instead, we use only simple quadratic optimization objectives so as to obtain analytic solutions. The high efficiency of our approach makes it possible to be used in some real-world applications.

Table 6. Speed of key steps in our approach. Mark and means using multi-test.

Key Step	Speed (s/frame)
Initialization (+)	0.05 (0.36)
Superpixel & Feature (+)	0.12 (0.12)
Build Flow & Propagation (+)	0.02 (0.26)
Primary Object Seg. (+)	0.01 (0.01)
Total (+)	∼0.20 (0.75)

5.3.4. Failure Cases

Beyond the successful cases, we also show in Figure 8 some failures. We find that failures can be caused by the way of defining primary objects. For example, the salient hand in Figure 8a is not labeled as a primary object, as the corresponding videos from **Youtube-Objects** are tagged with "dog". Moreover, shadows (Figure 8b) and reflections (Figure 8c) generated by the target object and environment are some other reasons that may cause unexpected failures due to their similar saliency with the target object. It is also easy to fail when parts of the regions of the target salient object are similar to the background (Figure 8d). Specifically, successful segmentation is very hard for some minuscule objects, e.g., a crab in water (Figure 8e). Such failures need further exploration in the future.

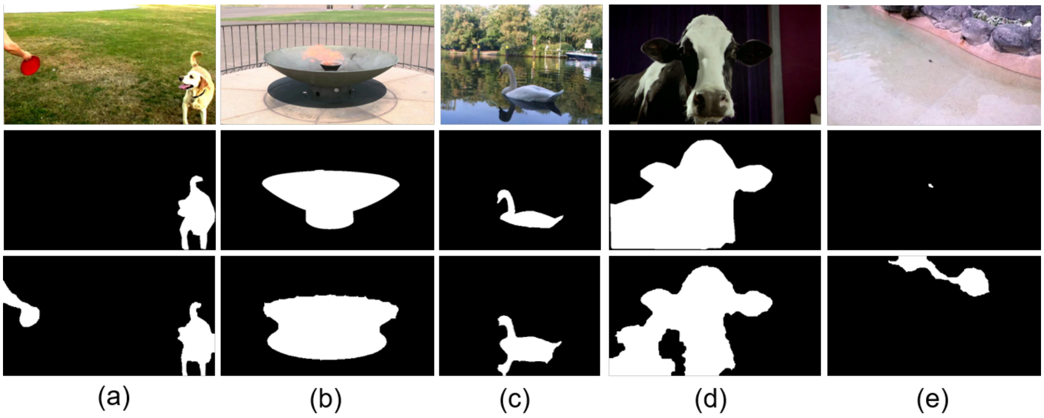

Figure 8. Failure cases of our approach. Rows from top to bottom: video frames, ground-truth masks and our results.

6. Conclusions

In this paper, we propose a simple yet effective approach for primary video object segmentation. Based on the complementary relationship of the foreground and the background, the problem of primary object segmentation is turned into an optimization problem of objective function. According to the proposed objective function, a complementary convolutional neural network is designed and trained on massive images from salient object datasets to handle complementary tasks. Then, by the trained models, the foreground and background in a video frame can be effectively predicted from the spatial perspective. After that, such spatial predictions are efficiently propagated via the inter-frame flow that has the characteristic of neighborhood reversibility. In this manner, primary objects in different frames can gradually pop out, while various types of distractors can be well suppressed. Extensive experiments on three video datasets have validated the effectiveness of the proposed approach.

In the future work, we intend to improve the proposed approach by fusing multiple methods of defining primary video objects such as motion patterns, semantic tags and human visual attention. Moreover, we will try to develop a completely end-to-end spatiotemporal model for primary video object segmentation by incorporating the recursive mechanism.

Author Contributions: Methodology, J.W. and J.L.; Project administration, J.W.; Validation, L.X.; Writing—original draft, J.W.; Writing—review & editing, J.L. and L.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under grant number 62132002 and 11790305.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Prest, A.; Leistner, C.; Civera, J.; Schmid, C.; Ferrari, V. Learning object class detectors from weakly annotated video. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3282–3289.
2. Jiang, H.; Wang, J.; Yuan, Z.; Wu, Y.; Zheng, N.; Li, S. Salient object detection: A discriminative regional feature integration approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2083–2090.
3. Zhang, D.; Meng, D.; Han, J. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 865–878. [CrossRef] [PubMed]
4. Lee, G.; Tai, Y.W.; Kim, J. Deep saliency with encoded low level distance map and high level features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 660–668.
5. Li, G.; Yu, Y. Visual saliency based on multiscale deep features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5455–5463.
6. Fan, D.P.; Li, T.; Lin, Z.; Ji, G.P.; Zhang, D.; Cheng, M.M.; Fu, H.; Shen, J. Re-thinking co-salient object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 4339–4354. [CrossRef] [PubMed]
7. Zhao, R.; Ouyang, W.; Li, H.; Wang, X. Saliency detection by multi-context deep learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1265–1274.
8. Zhuge, M.; Fan, D.P.; Liu, N.; Zhang, D.; Xu, D.; Shao, L. Salient object detection via integrity learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**. [CrossRef] [PubMed]
9. Wang, G.; Chen, C.; Fan, D.; Hao, A.; Qin, H. Weakly Supervised Visual-Auditory Saliency Detection with Multigranularity Perception. *arXiv* **2021**, arXiv:2112.13697.
10. Fan, D.P.; Zhang, J.; Xu, G.; Cheng, M.M.; Shao, L. Salient objects in clutter. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**. [CrossRef]
11. Zhao, J.X.; Liu, J.J.; Fan, D.P.; Cao, Y.; Yang, J.; Cheng, M.M. EGNet: Edge guidance network for salient object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 8779–8788.
12. Fan, D.P.; Cheng, M.M.; Liu, J.J.; Gao, S.H.; Hou, Q.; Borji, A. Salient objects in clutter: Bringing salient object detection to the foreground. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 186–202.
13. Wang, L.; Lu, H.; Ruan, X.; Yang, M.H. Deep networks for saliency detection via local estimation and global search. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3183–3192.
14. Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Van Gool, L.; Gross, M.; Sorkine-Hornung, A. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
15. Li, J.; Xia, C.; Chen, X. A benchmark dataset and saliency-guided stacked autoencoders for video-based salient object detection. *IEEE Trans. Image Process.* **2018**, *27*, 349–364. [CrossRef]
16. Maerki, N.; Perazzi, F.; Wang, O.; Sorkine-Hornung, A. Bilateral space video segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
17. Ramakanth, S.A.; Babu, R.V. SeamSeg: Video object segmentation using patch seams. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 376–383. [CrossRef]
18. Seguin, G.; Bojanowski, P.; Lajugie, R.; Laptev, I. Instance-level video segmentation from object tracks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
19. Tsai, Y.H.; Zhong, G.; Yang, M.H. Semantic co-segmentation in videos. In Proceedings of the 14th European Conference on Computer Vision, Munich, Germany, 8–14 September 2016.
20. Zhang, Y.; Chen, X.; Li, J.; Wang, C.; Xia, C. Semantic object segmentation via detection in weakly labeled video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3641–3649.
21. Zhang, D.; Javed, O.; Shah, M. Video Object Segmentation through Spatially Accurate and Temporally Dense Extraction of Primary Object Regions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 628–635. [CrossRef]
22. Jang, W.D.; Lee, C.; Kim, C.S. Primary object segmentation in videos via alternate convex optimization of foreground and background distributions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 696–704.
23. Papazoglou, A.; Ferrari, V. Fast object segmentation in unconstrained video. In Proceedings of the IEEE International Conference on Computer Vision, Portland, OR, USA, 23–28 June 2013; pp. 1777–1784.
24. Xiao, F.; Jae Lee, Y. Track and segment: An iterative unsupervised approach for video object proposals. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 933–942.

25. Ji, G.P.; Fu, K.; Wu, Z.; Fan, D.P.; Shen, J.; Shao, L. Full-duplex strategy for video object segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 10–17 October 2021; pp. 4922–4933.
26. Fan, D.P.; Wang, W.; Cheng, M.M.; Shen, J. Shifting More Attention to Video Salient Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
27. Chiu, W.C.; Fritz, M. Multi-class video co-segmentation with a generative multi-video model. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 321–328. [CrossRef]
28. Fu, H.; Xu, D.; Zhang, B.; Lin, S.; Ward, R.K. Object-based multiple foreground video co-segmentation via multi-state selection graph. *IEEE Trans. Image Process.* **2015**, *24*, 3415–3424. doi: 10.1109/TIP.2015.2442915. [CrossRef]
29. Zhang, D.; Javed, O.; Shah, M. Video object co-segmentation by regulated maximum weight cliques. In Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 551–566.
30. Yu, C.P.; Le, H.; Zelinsky, G.; Samaras, D. Efficient video segmentation using parametric graph partitioning. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3155–3163.
31. Wang, W.; Shen, J.; Yang, R.; Porikli, F. Saliency-aware video object segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 20–33. [CrossRef]
32. Cheng, M.M.; Mitra, N.J.; Huang, X.; Torr, P.H.; Hu, S.M. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 569–582. [CrossRef]
33. Jiang, P.; Ling, H.; Yu, J.; Peng, J. Salient region detection by ufo: Uniqueness, focusness and objectness. In Proceedings of the IEEE International Conference on Computer Vision, Portland, OR, USA, 23–28 June 2013; pp. 1976–1983.
34. Zhu, W.; Liang, S.; Wei, Y.; Sun, J. Saliency Optimization from Robust Background Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
35. Zhang, J.; Sclaroff, S. Saliency detection: A boolean map approach. In Proceedings of the IEEE International Conference on Computer Vision, Portland, OR, USA, 23–28 June 2013; pp. 153–160.
36. Ge, W.; Guo, Z.; Dong, Y.; Chen, Y. Dynamic background estimation and complementary learning for pixel-wise foreground/background segmentation. *Pattern Recognit.* **2016**, *59*, 112–125. [CrossRef]
37. Koh, Y.J.; Kim, C.S. Primary Object Segmentation in Videos Based on Region Augmentation and Reduction. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Volume 1, p. 7.
38. Tsai, Y.H.; Yang, M.H.; Black, M.J. Video segmentation via object flow. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3899–3908.
39. Li, J.; Zheng, A.; Chen, X.; Zhou, B. Primary video object segmentation via complementary cnns and neighborhood reversible flow. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1417–1425.
40. Liu, H.; Jiang, S.; Huang, Q.; Xu, C.; Gao, W. Region-based visual attention analysis with its application in image browsing on small displays. In Proceedings of the 15th ACM International Conference on Multimedia, Augsburg, Germany, 24–29 September 2007; pp. 305–308.
41. Perazzi, F.; Krähenbühl, P.; Pritch, Y.; Hornung, A. Saliency filters: Contrast based filtering for salient region detection. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 733–740.
42. Goferman, S.; Zelnik-Manor, L.; Tal, A. Context-aware saliency detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1915–1926. [CrossRef]
43. Wei, Y.; Wen, F.; Zhu, W.; Sun, J. Geodesic saliency using background priors. In Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 29–42.
44. Schauerte, B.; Stiefelhagen, R. Quaternion-based spectral saliency detection for eye fixation prediction. In Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 116–129.
45. Liu, T.; Yuan, Z.; Sun, J.; Wang, J.; Zheng, N.; Tang, X.; Shum, H.Y. Learning to detect a salient object. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 353–367.
46. Yang, C.; Zhang, L.; Lu, H.; Ruan, X.; Yang, M.H. Saliency detection via graph-based manifold ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 3166–3173.
47. Qin, Y.; Lu, H.; Xu, Y.; Wang, H. Saliency detection via cellular automata. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 110–119.
48. Zhang, J.; Sclaroff, S.; Lin, Z.; Shen, X.; Price, B.; Mech, R. Minimum barrier salient object detection at 80 fps. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1404–1412.
49. Li, X.; Zhao, L.; Wei, L.; Yang, M.H.; Wu, F.; Zhuang, Y.; Ling, H.; Wang, J. Deepsaliency: Multi-task deep neural network model for salient object detection. *IEEE Trans. Image Process.* **2016**, *25*, 3919–3930. [CrossRef]
50. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
51. Liu, N.; Han, J. Dhsnet: Deep hierarchical saliency network for salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 678–686.
52. Kuen, J.; Wang, Z.; Wang, G. Recurrent attentional networks for saliency detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3668–3677.

53. Li, G.; Yu, Y. Deep contrast learning for salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 478–487.
54. Yu, H.; Li, J.; Tian, Y.; Huang, T. Automatic interesting object extraction from images using complementary saliency maps. In Proceedings of the 18th ACM International Conference on Multimedia, Firenze, Italy, 25–29 October 2010; pp. 891–894.
55. Tian, Y.; Li, J.; Yu, S.; Huang, T. Learning complementary saliency priors for foreground object segmentation in complex scenes. *Int. J. Comput. Vis.* **2015**, *111*, 153–170. [CrossRef]
56. Liu, Z.; Zhang, X.; Luo, S.; Le Meur, O. Superpixel-based spatiotemporal saliency detection. *IEEE Trans. Circuits Syst. Video Technol.* **2014**, *24*, 1522–1540. [CrossRef]
57. Wang, W.; Shen, J.; Porikli, F. Saliency-aware geodesic video object segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3395–3402.
58. Ochs, P.; Malik, J.; Brox, T. Segmentation of moving objects by long term video analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1187–1200. [CrossRef]
59. Li, F.; Kim, T.; Humayun, A.; Tsai, D.; Rehg, J.M. Video segmentation by tracking many figure-ground segments. In Proceedings of the IEEE International Conference on Computer Vision, Portland, OR, USA, 23–28 June 2013; pp. 2192–2199.
60. Wang, W.; Shen, J.; Shao, L. Consistent video saliency using local gradient flow optimization and global refinement. *IEEE Trans. Image Process.* **2015**, *24*, 4185–4196. [CrossRef]
61. Nilsson, D.; Sminchisescu, C. Semantic video segmentation by gated recurrent flow propagation. *arXiv* **2016**, arXiv:1612.08871.
62. Gadde, R.; Jampani, V.; Gehler, P.V. Semantic video cnns through representation warping. *CoRR* **2017**, *8*, 9.
63. Zhu, X.; Xiong, Y.; Dai, J.; Yuan, L.; Wei, Y. Deep feature flow for video recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Volume 1, p. 3.
64. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
65. MSRA10K. Available online: http://mmcheng.net/gsal/ (accessed on 1 January 2016).
66. Yan, Q.; Xu, L.; Shi, J.; Jia, J. Hierarchical Saliency Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1155–1162. [CrossRef]
67. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC Superpixels Compared to State-of-the-art Superpixel Methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [CrossRef]
68. Jegou, H.; Schmid, C.; Harzallah, H.; Verbeek, J. Accurate image search using the contextual dissimilarity measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 2–11. [CrossRef]
69. Jain, S.D.; Grauman, K. Supervoxel-consistent foreground propagation in video. In Proceedings of the 13th European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 656–671.
70. Peng, H.; Li, B.; Ling, H.; Hu, W.; Xiong, W.; Maybank, S.J. Salient object detection via structured matrix decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 818–832. [CrossRef]
71. Tong, N.; Lu, H.; Ruan, X.; Yang, M.H. Salient object detection via bootstrap learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1884–1892.
72. Tu, W.C.; He, S.; Yang, Q.; Chien, S.Y. Real-time salient object detection with a minimum spanning tree. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2334–2342.
73. Wang, L.; Wang, L.; Lu, H.; Zhang, P.; Ruan, X. Saliency detection with recurrent fully convolutional networks. In Proceedings of the 14th European Conference on Computer Vision, Munich, Germany, 8–14 September 2016; pp. 825–841.
74. Faktor, A.; Irani, M. Video Segmentation by Non-Local Consensus Voting. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 1–5 September 2014.
75. Liu, J.J.; Hou, Q.; Liu, Z.A.; Cheng, M.M. Poolnet+: Exploring the potential of pooling for salient object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**. [CrossRef]
76. Liu, J.J.; Hou, Q.; Cheng, M.M. Dynamic Feature Integration for Simultaneous Detection of Salient Object, Edge and Skeleton. *IEEE Trans. Image Process.* **2020**, *29*, 8652–8667. [CrossRef]
77. Liu, Y.; Zhang, X.Y.; Bian, J.W.; Zhang, L.; Cheng, M.M. SAMNet: Stereoscopically Attentive Multi-scale Network for Lightweight Salient Object Detection. *IEEE Trans. Image Process.* **2021**, *30*, 3804–3814. [CrossRef]
78. Ma, M.; Xia, C.; Li, J. Pyramidal feature shrinking for salient object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; Volume 35, pp. 2311–2318.