Deep Learning for Video Object Segmentation

Gao, Mingqi; Zheng, Feng; Yu, James; Shan, Caifeng; Ding, Guiguang; Han, Jungong

Published in:
Artificial Intelligence Review

DOI:
10.1007/s10462-022-10176-7

Publication date:
2023

Citation for published version (APA):
Gao, M., Zheng, F., Yu, J., Shan, C., Ding, G., & Han, J. (2023). Deep Learning for Video Object Segmentation: A Review. *Artificial Intelligence Review*, *56*(1), 457-531. https://doi.org/10.1007/s10462-022-10176-7

Artificial Intelligence Review
https://doi.org/10.1007/s10462-022-10176-7

Deep learning for video object segmentation: a review

Mingqi Gao[1,2] · Feng Zheng[2] · James J. Q. Yu[2] · Caifeng Shan[3] · Guiguang Ding[4] · Jungong Han[1,5]

Abstract
As one of the fundamental problems in the field of video understanding, video object segmentation aims at segmenting objects of interest throughout the given video sequence. Recently, with the advancements of deep learning techniques, deep neural networks have shown outstanding performance improvements in many computer vision applications, with video object segmentation being one of the most advocated and intensively investigated. In this paper, we present a systematic review of the deep learning-based video segmentation literature, highlighting the pros and cons of each category of approaches. Concretely, we start by introducing the definition, background concepts and basic ideas of algorithms in this field. Subsequently, we summarise the datasets for training and testing a video object segmentation algorithm, as well as common challenges and evaluation metrics. Next, previous works are grouped and reviewed based on how they extract and use spatial and temporal features, where their architectures, contributions and the differences among each other are elaborated. At last, the quantitative and qualitative results of several representative methods on a dataset with many remaining challenges are provided and analysed, followed by further discussions on future research directions. This article is expected to serve as a tutorial and source of reference for learners intended to quickly grasp the current progress in this research area and practitioners interested in applying the video object segmentation methods to their problems. A public website is built to collect and track the related works in this field: https://github.com/gaomingqi/VOS-Review.

Keywords Video object segmentation · Deep learning · Convolutional neural network

1 Introduction

Video Object Segmentation (VOS) is the task of separating foreground regions from backgrounds in video sequences (Cucchiara et al. 2003). Similar to object tracking (Yilmaz et al. 2006), VOS methods establish the correspondence of identical objects

✉ James J. Q. Yu
yujq3@sustech.edu.cn

✉ Jungong Han
jungong.han@warwick.ac.uk

Extended author information available on the last page of the article

Published online: 08 April 2022

across frames, but more detailed object representation can be achieved (pixel-level masks rather than bounding boxes). Therefore, VOS has played an important role in many real-world applications, e.g. visual surveillance, action recognition, video summarisation and video editing (Perazzi et al. 2016a). In early VOS methods based on hand-crafted features, the objectness (Zhang et al. 2013), optical flow (Papazoglou and Ferrari 2013; Tsai et al. 2016) and visual saliency (Faktor and Irani 2014; Wang et al. 2015) are the frequently used techniques to segment objects from video sequences. Although these methods achieved state-of-the-art results at that time, with the development of deep learning techniques and high performance computing, deep learning-based VOS methods have made great progress in terms of both accuracy and efficiency. Therefore, most of the recent VOS methods are implemented based on deep neural networks. The statistical data given by two authoritative VOS benchmarks (Perazzi et al. 2016a; Xu et al. 2018b) reveal that the performance of existing VOS approaches is improving year by year but has not yet attained saturation. With its potential applications and room for improvement in performance, deep learning-based VOS has become an active research topic in computer vision.

The existing VOS methods can be mainly grouped into four types: unsupervised, semi-supervised, interactive, and referring (or language-guided). This paper focuses on the two widely studied types among them: unsupervised VOS (UVOS) and semi-supervised VOS (SVOS). Note that 'unsupervised' and 'semi-supervised' in VOS and general machine learning tasks have different application scopes. In VOS, these terms indicate the level of supervision required during inference instead of training (Perazzi et al. 2016a). Specifically, UVOS methods perform segmentation without any ground truth labels or priors (unsupervised setting). The objects with prominent motion patterns or visual saliency can be segmented. SVOS methods, on the other hand, initiate with the ground truth labels available in a few frames (generally the first frame only, semi-supervised setting). These labels are manually annotated to indicate the objects to be segmented from the remaining frames. To avoid conceptual confusion, it is worth mentioning that some recent works term **unsupervised/semi-supervised VOS** as **automatic/semi-automatic VOS** or **zero-shot/one-shot VOS**.

Figure 1 illustrates the difference between the two VOS methods. It is observed that the target objects (the ones to segment) in UVOS and SVOS are defined automatically and manually, respectively. Most earlier UVOS methods perform single object segmentation since it is hard to discriminate object instances based on motion patterns and visual saliency. With the integration of the instance-level segmentation module, some recent methods dedicated to unsupervised multi-object segmentation have been proposed.

Recently, two review papers on video segmentation have appeared. Yao et al. (2020) provided a good survey on the video object segmentation and tracking methods based on hand-crafted features and deep learning. Wang et al. (2021b) comprehensively reviewed the deep learning-based techniques for video object segmentation and video semantic segmentation. Unlike their broad scopes, our paper focuses on deep learning-based unsupervised/semi-supervised VOS methods to provide more detailed classification, review, and validation experiments on this topic, thus allowing readers to understand better the mechanism, progress and development trends of these methods. Therefore, we recommend readers access the papers by Yao et al. (2020) and Wang et al. (2021b) for the review of interactive and referring (or language-guided) VOS and the paper by Wang et al. (2021b) to review video semantic segmentation methods. Our focus is the deep learning-based methods for unsupervised/semi-supervised VOS.

In summary, the main contributions of our work are as follows:

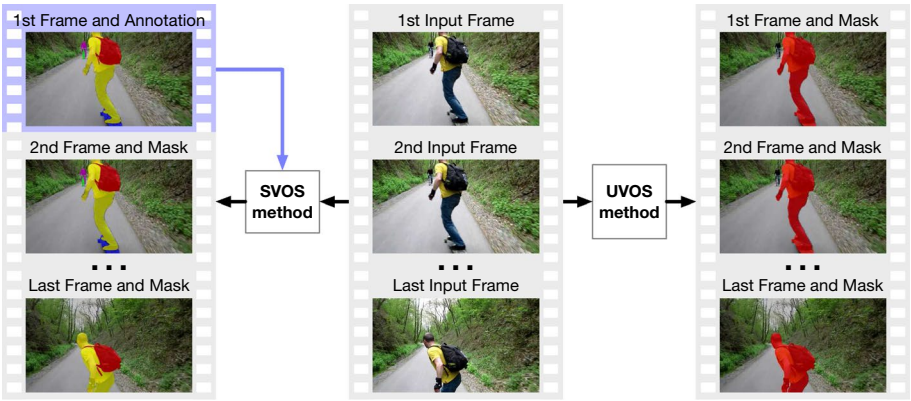

Fig. 1 Diagram of UVOS and SVOS methods, both of which take raw videos as inputs. UVOS methods segment the objects with dominant movement or visual saliency. In contrast, the target objects (the ones to segment) in SVOS depend on the human annotations in the first frame (highlighted in purple). Therefore, SVOS methods have more flexibility in defining target objects

- We provide a review and analysis of the datasets beneficial to train and evaluate UVOS and SVOS methods.
- We group the existing UVOS and SVOS methods into six categories according to spatial and temporal feature utilisation and provide an in-depth and organised review of their origins, development histories, architectures, pros, cons, and representative methods.
- We discuss the performances of the reviewed methods by analysing the evaluation results released on several benchmark datasets and testing some representative techniques on different types of challenging video sequences.
- We summarise several developing trends of the reviewed methods and draw some forecasts on possible advances in future.

Figure 2 visualises the table of contents of this paper. The remainder of this paper is structured as follows. Section 2 describes several background notions related to VOS, including semantic segmentation and typical network architectures used in VOS methods. Meanwhile, Sect. 3 introduces and summarises existing datasets, challenges and evaluation metrics. Section 4 reviews existing deep learning-based VOS methods, which are first, grouped based on their technical properties and then discussed in detail from aspects of architectures and contributions. The quantitative and qualitative results of representative VOS methods are provided in Sect. 5, where the forecasts for future works are also mentioned. In Sect. 6, we summarise this paper.

2 Background

In Sect. 2.1, we briefly review the historical background in the VOS field. Next, the main principle and representative methods in semantic image segmentation are introduced in Sect. 2.2. Section 2.3 summarises the basic architecture for the recent VOS methods. The milestone works in VOS is shown in Fig. 3.

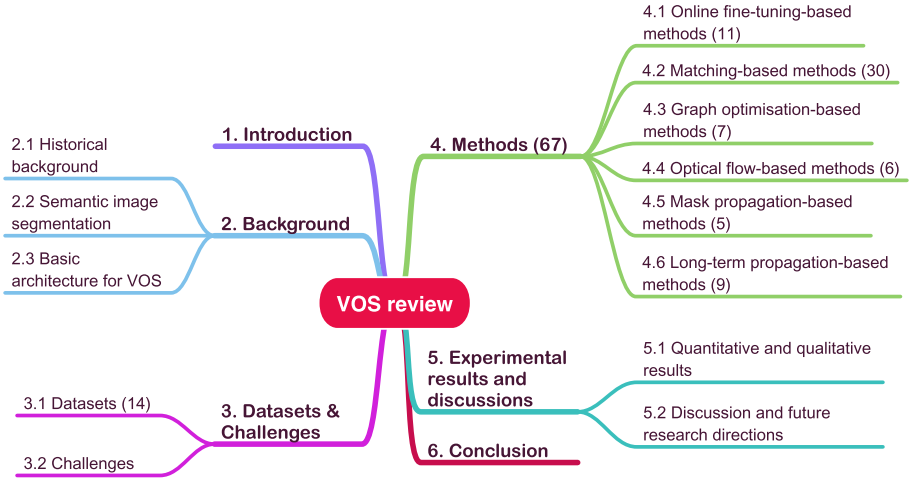

Fig. 2 The visualised table of contents of this paper. Note that the number behind Sect. 3.1 indicates how many datasets are discussed in this subsection. Same principle for the Sect. 4 and its subsections (note that there is one paper (DyeNet (Li and Change Loy 2018)) appearing twice in Sects. 4.2 and 4.5 because of its novelty in both feature matching-based and mask propagation-based VOS)

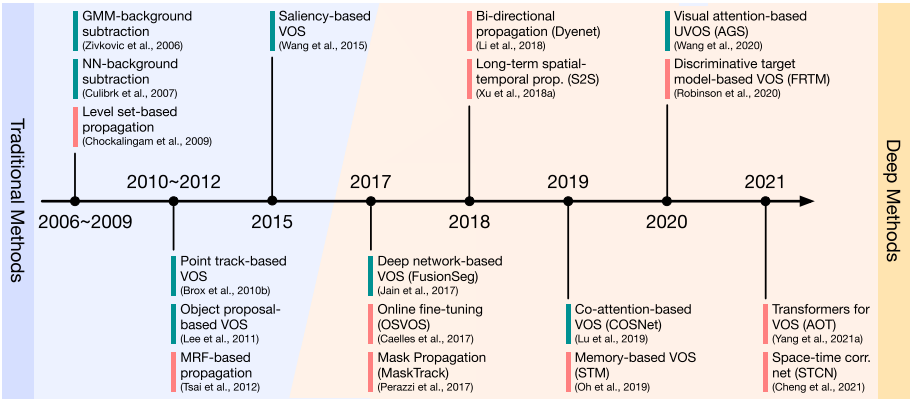

Fig. 3 A brief chronology of VOS methods, where some milestone works from 2006 to 2021 are highlighted. Green marks: upsupervised VOS methods; Red marks: semi-supervised VOS methods. More details about the traditional methods are shown in Sect. 2.1, while the deep methods are discussed in Sect. 4

2.1 Historical background in VOS

The early attempts (Chien et al. 2002; Kim and Hwang 2002) for VOS mainly focus on extracting the moving object from the video sequence, which is a key operation in several multimedia applications such as content-based video coding (Sikora 1997). **Background subtraction** is the most frequently used approach in these methods. The major steps can be summarised as: (1) build a background model with the difference between successive frames; (2) extract segmentation results by subtracting backgrounds from the current

frame. In early methods, the background model mostly comes from the pixel values or filtering results. To further improve the robustness against complex scenes, several statistical techniques, such as Gaussian mixture model (Zivkovic and Van Der Heijden 2006) and neural network (Culibrk et al. 2007), are considered in background modelling.

With the success of the method for object proposal generation (Endres and Hoiem 2010), the VOS methods based on **Object Proposals** (Lee et al. 2011; Zhang et al. 2013; Ma and Latecki 2012) are developed to segment objects from the video sequence. The major steps can be summarised as: (1) generate object proposals for all video frames; (2) group and rank the generated proposals to derive the recurring object. In comparison, object proposal-based methods can handle more challenging sequences (e.g., complex background, static object) than background subtraction-based methods. However, these methods generally run slowly due to inefficient operations for proposal generation and grouping. Therefore, the object proposals are rarely used in subsequent methods and replaced by the spatial-temporal boundaries (Papazoglou and Ferrari 2013; Wang et al. 2015) to estimate salient object locations.

Since the objects mostly move smoothly in the video sequence, temporal continuity could be beneficial during segmentation. However, only short-term continuity is explored in the early methods. To build the long-term relationships over frames, the **Point Trajectory**-based methods (Brox and Malik 2010b; Ochs and Brox 2012; Fragkiadaki et al. 2012) are developed. The major steps can be summarised as: 1) build point trajectories based on the motion information (e.g., optical flow); 2) measure the affinities between the trajectories and cluster them for the segmentation results. In early trajectory-based methods, the affinity and clusters are mostly generated upon the local information of trajectories, which makes the final results vulnerable to error trajectories. To address the problem, both local and global information of trajectories are explored in subsequent methods (Chen et al. 2015b).

Besides automatic approaches for moving object segmentation, **VOS with few annotations** (Zhong and Chang 1999; Chockalingam et al. 2009) has also drawn some attention in the early stages, which can be seen as the earlier semi-supervised VOS methods. As mentioned above, the goal of SVOS is to 'propagate' the annotated masks/contours to the remaining frames. Before the rise of deep learning-based methods, the research of SVOS (Fan et al. 2015; Wang et al. 2017b) is mainly focused on discriminative feature descriptors and reliable temporal correspondences to achieve coherent information propagation throughout the sequence.

2.2 Semantic image segmentation

In most of existing VOS methods, the segmentation process is conducted frame by frame, thus the deep learning-based techniques for image analysis are beneficial to VOS. For instance, the deep learning-based semantic image segmentation classifies each pixel into a predefined semantic category based on encoded features (Garcia-Garcia et al. 2018; Ghosh et al. 2019). Likewise, VOS is also a pixel-level classification task, thus it is necessary to introduce the main principle and representative methods in image segmentation before discussing VOS methods.

Recently, Convolutional Neural Networks (CNNs) have shown superior performance in many computer vision tasks, e.g., image classification (Krizhevsky et al. 2012; Simonyan and Zisserman 2015) and object detection (Girshick et al. 2014; Girshick 2015). To bring such success to image segmentation, Long et al. (2015) made a few changes to the CNN

originally designed for image classification, in which the last few fully connected layers of the CNN are replaced by fully convolutional layers and upsampling modules. In this way, the transformed network, named fully convolutional network (FCN), can take the input of the arbitrary size and generate the output with the corresponding size. During training, FCN is initialised with the weights pre-trained on ImageNet (Russakovsky et al. 2015, a dataset for image classification), and then fine-tuned on a segmentation dataset (PASCAL (Everingham et al. 2012)). For each pixel in the input image, FCN generates a set of probabilities indicating how likely the pixel belongs to all semantic categories. Intuitively, FCN adjusts conventional CNNs to generate category scores for all pixels in the input image, rather than the image only.

Due to its superior performance and end-to-end trainability, FCN has been a mainstream network architecture in semantic image segmentation. After that, many models have been proposed for further improvement, with most efforts focusing on how to restore the resolution of the segmentation map to that of the input image, such as DeconvNet (Noh et al. 2015), U-Net (Ronneberger et al. 2015), and SegNet (Badrinarayanan et al. 2017). In the model family DeepLab proposed by Chen et al. (2015a, 2017a, b, 2018a), the performance for semantic image segmentation has been taken to new heights. To reduce the loss in resolution and improve the accuracy of boundary localisation, the early versions of DeepLab (Chen et al. 2015a, 2017a) have been combined with dilated convolution (Yu and Koltun 2016), spatial pyramid pooling (He et al. 2015) and fully connected conditional random field (CRF) (Krähenbühl and Koltun 2011). By integrating image-level feature maps into parallel dilated convolutional module, DeepLabv3 (Chen et al. 2017b) further improves the segmentation accuracy, while simultaneously removing the time-consuming CRF from the model. In the latest version of DeepLab ('v3+') (Chen et al. 2018a), Xception (Chollet 2017) is implemented as a backbone network for feature extraction. Also, an encoder-decoder structure is adopted to achieve more accurate region boundaries. Although many new algorithms have been proposed in recent years for semantic image segmentation, DeepLab models are still the most frequently used architectures in VOS (see Tables 3, 4, and 5) due to the stability of their performance. For more details about deep learning-based semantic image segmentation, please refer to the papers written by Garcia-Garcia et al. 2018 and Ghosh et al. (2019).

2.3 Basic architecture for VOS

Existing deep learning-based VOS solutions, including semantic image segmentation, are primarily based on the FCN architecture. As shown in Fig. 4, the fundamental architecture for VOS methods consists of two sub-modules: encoder and decoder, which perform the tasks of feature extraction and resolution restoration, respectively. In this way, existing methods formulate VOS as an object segmentation problem frame-by-frame, thus the basic architecture of VOS looks similar to that of image segmentation. To derive the object masks with semantic consistency and temporal continuity, extra guidance from the video sequence is given to the architecture. The concrete form of the guidance depends on the used techniques, which will be introduced later.

The network architectures used in discussed VOS methods are listed in Tables 3, 4, and 5, from which it is observed that several typical classification networks are implemented for feature encoding, such as VGGNet (Simonyan and Zisserman 2015), ResNet (He et al. 2016) and its variant (Wu et al. 2019), and DenseNet (Huang et al. 2017). In addition, the DeepLab series (Chen et al. 2015a, 2017a, b, 2018a) are also frequently utilised

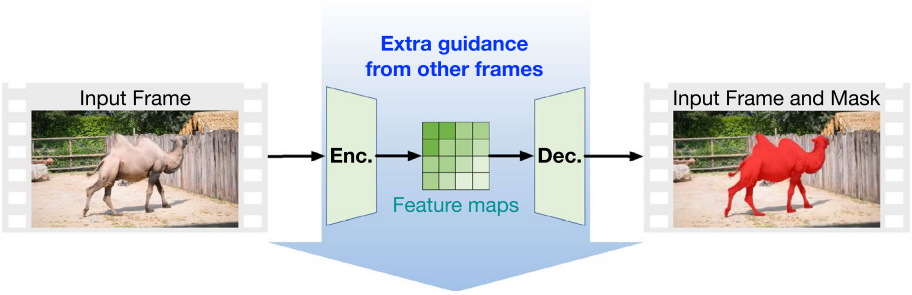

Fig. 4 The diagram of basic architecture for VOS methods. Enc.: encoder; Dec.: decoder. The green rectangle represents the feature map generated by the encoder. In most VOS methods, the segmentation is achieved by performing the target object extraction frame by frame, where each frame is segmented according to the spatial-temporal clues provided from other frames in the same sequence. Best viewed in colour

for semantic feature embedding. As for the decoding process, most of existing methods accomplish it by upsampling the encoded feature maps (through bilinear interpolation or transposed convolution) and combining low-level features. More details about these architectures can be found in Sect. 4.

3 Datasets and challenges

Considering the high demand of deep learning systems for data, this section browses the existing datasets for VOS, followed by corresponding evaluation metrics and main challenges.

3.1 Datasets

Table 1 shows 14 video datasets as well as their main properties. Based on these properties, the listed datasets are discussed in detail, especially in aspects of challenges and applicable settings, to guide the researchers interested in VOS to choose proper datasets for training and evaluating their own methods.

3.1.1 Hopkins-155

This dataset (Tron and Vidal 2007) was designed for evaluating point-based motion segmentation algorithms, where a set of points (39-550 points) instead of the whole pixels in each video frame are annotated. The involved sequences are grouped into three categories: 1) checkerboard: the moving objects are covered by checkerboard pattern to assure the number of tracked points; 2) traffic scenes: consisting of outdoor traffic scenes; 3) articulated/non-rigid objects: consisting of the sequences with the motions of joints, face, and people walking. As an earlier benchmark, Hopkins-155 provides the community with a chance to evaluate the robustness of the segmentation methods against rotation, translation, and degenerate motions. Given the sparse annotation and limited challenges, however, it is not encouraged to use Hopkins-155 to train and evaluate deep learning-based VOS methods.

Table 1 Summary of existing datasets for VOS

Datasets	D. type		O. num		DA	Resolution	Videos	Annotations	Categories	Objects
	R	S	S	M						
Hopkins-155 (Tron and Vidal 2007)	✓			✓	✓	320 × 240 – 640 × 480	155	4615	–	345*
BMS-26 (Brox and Malik 2010b)	✓			✓		350 × 288 – 640 × 480	26	189	2	47
FBMS-59 (Ochs et al. 2013)	✓			✓		350 × 288 – 960 × 540	59	720	11	139
SegTrackV1 (Tsai et al. 2012)	✓		✓		✓	320 × 240 – 414 × 352	6	244	6	6
SegTrackV2 (Li et al. 2013)	✓			✓	✓	320 × 240 – 640 × 360	14	1154	12	24
YouTube-Objects (Prest et al. 2012)	✓		✓			320 × 240 – 960 × 540	126	2127	10	126
JumpCut (Fan et al. 2015)	✓	✓	✓		✓	640 × 400 – 1280 × 720	22	6331	12	22
DAVIS-2016 (Perazzi et al. 2016a)	✓		✓		✓	854 × 480	50	3455	–	50
DAVIS-2017 (Pont-Tuset et al. 2017)	✓			✓	✓	854 × 480	150	10,459	–	376
DAVIS-2017-U (Caelles et al. 2019)	✓			✓	✓	854 × 480	150	10,731	–	449
YouTube-VOS-2018 (Xu et al. 2018b)	✓			✓		1280 × 720	4453	197,272	94	7754
YouTube-VOS-2019 (Xu et al. 2019b)	✓			✓		1280 × 720	4519	>190,000	94	8614
YouTube-VIS (Yang et al. 2019a)	✓			✓		1280 × 720	2883	>131,000	40	4883
SAIL-VOS (Hu et al. 2019)		✓		✓	✓	1280 × 800	201	111,654	162	1,896,295

D type data type of contained video sequence, *R* real data, *S* synthetic data; *O. num* number of objects annotated in each video sequence, *S* single object, *M* multiple objects, *DA* dense annotation, i.e. all involved video frames are annotated; 'Videos': number of video sequences; 'Annotations': number of annotated frames; 'Categories': number of the object categories involved; 'Objects': number of the annotated objects. Note that 'Objects' in Hopkins-155 is marked with '*' because except for the moving objects, some background scenes are also annotated to be tracked

3.1.2 BMS (Berkeley Motion Segmentation Dataset) series

This dataset series was designed for moving object segmentation, and is composed of two versions of sets: BMS-26 (Brox and Malik 2010b) and FBMS-59 (Ochs et al. 2013, Freiburg-BMS). BMS-26 consists of 26 video sequences, where human and car are the most frequently used object categories. FBMS-59 extends BMS-26 by increasing the number of video sequences to 59 and involving more object categories. In both datasets, the challenges such as occlusion and motion pattern variation are covered, thus the robustness of VOS methods against them can be evaluated on these datasets. With respect to the annotated data for training, however, since parts of video sequences are with low spatial resolution and only a sparse subset of frames was annotated, it is difficult to achieve a robust VOS system from these datasets only.

3.1.3 SegTrack series

This is a small-scale dataset series, designed for video object segmentation and tracking; it consists of 2 versions of sets: SegTrack v1 (Tsai et al. 2012) and SegTrack v2 (Li et al. 2013). SegTrack v1 contains only 6 video sequences, but all frames are annotated with pixel-level masks. After adding more video sequences and annotated objects, SegTrack v2 extends the previous version. The video sequences in both datasets are challenging in the sense that fast motion and object deformation appear frequently. Similar to the BMS series, SegTrack also has relatively low spatial resolution. In addition, as shown in Table 1, the number of videos, categories and objects in SegTrack series are limited. Therefore, training deep learning-based VOS methods on this dataset series only is not encouraged.

3.1.4 YouTube-Objets

This dataset (Prest et al. 2012) was originally designed for video object detection, where all video sequences (containing a total of 570,000 frames) are downloaded from the internet, grouped into 10 categories. To make this dataset available for VOS, Jain and Grauman (2014) selected a subset of video frames (over 20,000 frames) and annotated pixel-level masks in every 10-th frame. The resulting dataset consists of 126 video sequences with 2,127 annotated frames, and has become the largest VOS dataset at that time. However, due to its sparse annotations and uneven category distribution, it is not an appropriate dataset for VOS method training.

3.1.5 JumpCut

This dataset (Fan et al. 2015) consists of 22 video sequences with 6,331 frames, all of which are annotated with pixel-level masks. Besides the sequences captured in the real world, the dataset includes a small amount of animation frames. Based on the involved object categories (mainly human and animals) and challenges (fast motion and static objects), the dataset is divided into different groups for better organisation. Owing to its challenging settings and long-range dense annotations, JumpCut has been a desirable

dataset for VOS evaluation. Also, JumpCut is suitable for model training, especially when collaborating with other small-scale datasets.

3.1.6 DAVIS (Densely Annotated VIdeo Segmentation) series

This high-resolution dataset series has evolved over the years into three versions: DAVIS-2016 (Perazzi et al. 2016a), DAVIS-2017 (Pont-Tuset et al. 2017) and DAVIS-2017-U (Caelles et al. 2019), corresponding to different kinds of VOS tasks, respectively. Compared with the aforementioned datasets, DAVIS datasets have more sequences, annotations and challenges, which makes them prevalent for training and evaluation. DAVIS-2016, designed for single-object SVOS and UVOS tasks, is the first released dataset among the series. By adding more sequences and annotations, DAVIS-2017 is proposed for multi-object SVOS. With the concept of multi-object UVOS being concerned, DAVIS-2017-U (un-supervised version) is released recently, where the video frames in the original DAVIS-2017 are re-annotated.

Besides datasets, the DAVIS team has organised a yearly challenge[1] relating to VOS since 2017, which significantly booms the development of VOS methods.

3.1.7 YouTube-VOS series

This is a large-scale dataset series for VOS, with long-range video sequences; it contains three versions: YouTube-VOS 2018 (Xu et al. 2018b), YouTube-VOS 2019 (Xu et al. 2019b) and YouTube-VIS (Yang et al. 2019a). The first two versions are designed for multi-object SVOS, while the latter one serves for multi-object UVOS. From Table 1, it can be found that the number of video sequences in YouTube-VOS is dozens of times as many as that in DAVIS, which indicates more diverse objects and context are considered. Moreover, each video sequence in the datasets has a greater number of frames than any other datasets, allowing VOS methods to model and exploit long-range temporal dependency between frames. Because the amount of the data is huge, the YouTub-VOS team only managed to provide the pixel-level object masks for every 5-th frame.

To better validate the generalisation ability of VOS models, YouTube-VOS groups the contained object categories into two sets: 'seen' and 'unseen', where the objects belonging to 'unseen' categories only residents in the testing set, and the ones belonging to 'seen' categories residents in both training and testing sets. By comparing the segmentation results on 'seen' and 'unseen' objects, the performance of VOS models on generalisation can be evaluated. To echo DAVIS, the YouTube-VOS team organises a challenge[2] on VOS annually since 2018.

3.1.8 SAIL-VOS (Semantic Amodal Instance Level Video Object Segmentation)

This is a synthetic dataset for VOS (Hu et al. 2019), where all video frames and corresponding masks are collected from the Grand Theft Auto V, an action-adventure game. The pictures in the game are rendered to be as realistic as possible, thus it is useful for training and evaluating VOS methods. In addition, since all video sequences are generated by the

[1] https://davischallenge.org.
[2] https://youtube-vos.org.

game simulator, the obtained object masks are completely credible, even if they are experiencing a heavy occlusion.

3.1.9 Evaluation metrics

In VOS, the commonly used metrics for performance evaluation are Jaccard index $\mathcal{J}$ (Everingham et al. 2010), F-measure $\mathcal{F}$ (Martin et al. 2004), and the mean of them $\mathcal{J}\&\mathcal{F}$:

$$\begin{cases} \mathcal{J} = \dfrac{|M \cap G|}{|M \cup G|} \\ \mathcal{F} = \dfrac{2P_c R_c}{P_c + R_c} \\ \mathcal{J}\&\mathcal{F} = \dfrac{(\mathcal{J} + \mathcal{F})}{2} \end{cases}, \tag{1}$$

where G and M refer to the ground truth mask and segmented mask, respectively. $\mathcal{J}$ evaluates the region similarity between these two masks. P_c and R_c are precision and recall computed from the points in contours $c(M)$ and $c(G)$. Therefore, $\mathcal{F}$ evaluates the accuracy of boundary localisation. $\mathcal{J}\&\mathcal{F}$ measures the overall VOS performance.

3.1.10 Summary

Sections 3.1.1–3.1.8 review the datasets for training and evaluating VOS methods. The earlier datasets, including Hopkins-155 (Tron and Vidal 2007), BMS series (Brox and Malik 2010b; Ochs et al. 2013), and SegTrack series (Tsai et al. 2012; Li et al. 2013), were designed originally to evaluate non-deep learning methods. In the deep learning era, these datasets can still rate the performance of VOS methods in handling object deformation and occlusion. However, **they have been rarely employed in more recent methods**, limited by the data diversity, number of challenges, and video length.

YouTube-Objects (Prest et al. 2012; Jain and Grauman 2014) and JumpCut (Fan et al. 2015) consists of long-range and high-resolution videos. Therefore, these datasets are popular in evaluating the performance of earlier VOS methods in spatial-temporal feature embedding. However, the limitations in data diversity and challenges remain in these datasets. **Only a few recent UVOS methods evaluated their performance on YouTube-Objects and JumpCut**.

Unlike other datasets, SAIL-VOS (Hu et al. 2019) consists of synthetic videos. Although there are still gaps between the rendered and actual video frames, one character of SAIL-VOS cannot be ignored: the occlusions are entirely reliable and under control, which, therefore, can improve the robustness of VOS methods against occlusions. However, **no reviewed methods employ SAIL-VOS during training or evaluation**.

DAVIS series (Perazzi et al. 2016a; Pont-Tuset et al. 2017; Caelles et al. 2019) and YouTube-VOS series (Xu et al. 2018b, 2019b; Yang et al. 2019a) are the **most frequently used datasets for training and evaluating recent VOS methods** (For single-object UVOS: DAVIS-2016; For multi-object UVOS: DAVIS-2017-U, YouTube-VIS; For SOVS: DAVIS-2016, 2017, YouTube-VOS-2018, 2019). This is because these datasets consider large-scale video sequences, diverse object categories, more challenges, and high-quality

Table 2 The average and state-of-the-art (SOTA) SVOS performance (measured by $\mathcal{J}\&\mathcal{F}$ in Eq. 1) on 4 benchmark datasets. s: seen, u: unseen

Dataset series	Subsets	Average $\mathcal{J}\&\mathcal{F}$		SOTA $\mathcal{J}\&\mathcal{F}$	
DAVIS	2016 val	81.9		90.7	
	2017 val	71.4		85.4	
	2017 test-dev	60.6		78.1	
YouTube-VOS	2018 val	72.4 (s)	53.1 (u)	83.0 (s)	79.6 (u)

annotations. Due to the difference between DAVIS and YouTube series in annotations, different VOS properties could be evaluated on these datasets, respectively. For example, it is encouraged to assess the temporal stability of VOS methods on DAVIS (densely annotated dataset) instead of YouTube-VOS or VIS (sparsely annotated dataset). Relevant codes are available[3]. As for the generalisation performance in VOS, the YouTube series is preferred because the series consists of a large number of videos and part of object categories only appear in the validation set. Moreover, the number of long sequences in the YouTube series is far more than DAVIS, thus facilitating the evaluation of the robustness and sequential modelling of VOS methods.

To further discuss the difference between DAVIS and YouTube series, we measure the performance of all the reviewed SVOS methods on these dataset series. UVOS setting is not considered since too few relevant methods were tested on YouTube datasets. Table 2 shows the comparison results, including the average and state-of-the-art performance on the DAVIS-2016 validation set, DAVIS-2017 validation set, DAVIS-2017 test-dev set, and YouTube-VOS-2018 validation set. It is observed that the performance on DAVIS-2016 tends to be saturated. This is because each video in the dataset has only one annotated object and relatively few challenges. In recent SVOS papers, the DAVIS-2017 test-dev set has become increasingly favoured due to more challenging sequences (e.g., more shape complexity, occlusions, and dynamic background) than the DAVIS-2017 validation set. It is also observed that YouTube-VOS 2018 is the second most challenging dataset in Table 2. Unlike DAVIS, YouTube-VOS divides the evaluation metrics into two subsets: seen and unseen, to measure the performance of SVOS methods on the objects whose categories appear and disappear in the training set, respectively. In general, for each SVOS method, the performance on the seen set is higher than that on the unseen set. The gaps between them measure the generalisation performance.

3.2 Common challenging factors

This section introduces several challenges to the UVOS and SVOS fields, including property changes, occlusions, the conflict between similar instances, ambiguous backgrounds, temporal consistency, and the balance between efficiency and accuracy. These challenges motivate most current methods and are visualised in Figs. 5 and 6.

[3] https://github.com/davisvideochallenge/davis-2017.

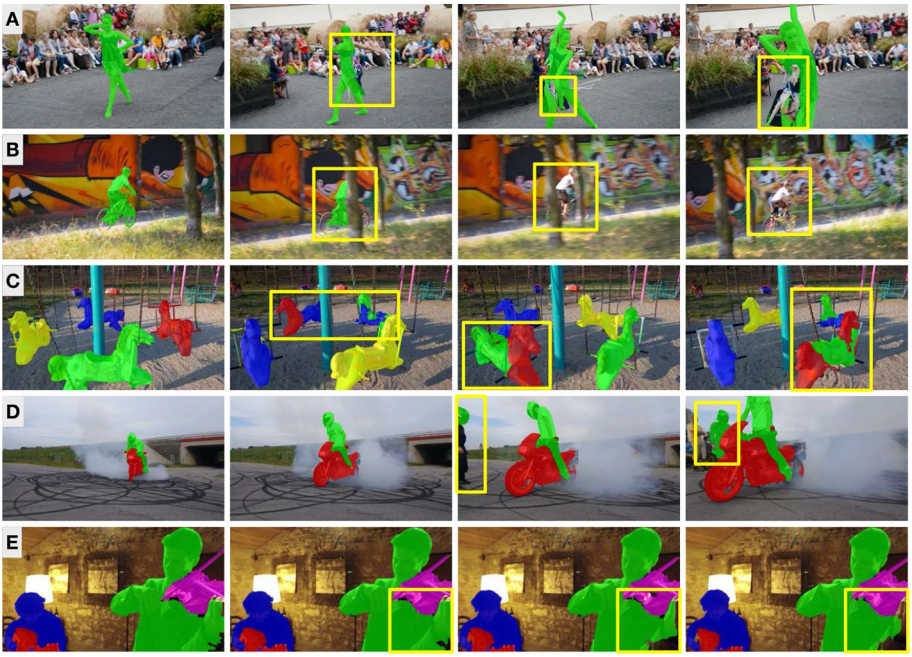

Fig. 5 False results raised by the introduced challenges, where each row shows the effect of one challenging factor on the existing VOS methods. **A** Property changes; **B** occlusions; **C** discrimination between similar objects; **D** ambiguous backgrounds; **E** temporally consistent VOS (this row consists of continuous frames without fast motion, occlusion, and significant appearance changes). Yellow boxes highlight the false results

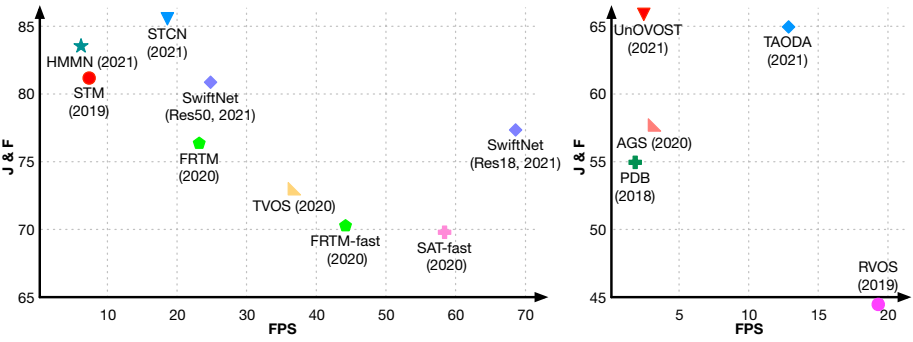

Fig. 6 Segmentation accuracy and efficiency of the recent SVOS (left) and UVOS (right) methods on DAVIS-2017 validation set

3.2.1 Object property change

This challenge mainly affects the VOS methods based on visual similarities. During inference, these methods segment the regions with similar visual features to the target objects annotated (most are SVOS methods) or predicted (most are UVOS methods) in

Table 3 Summary of the discussed VOS methods (part 1/3)

Methods	Sup.	S. techs			T. techs			Architecture	Main contribution(s)
		O	M	G	O	P	L		
OSVOS (Caelles et al. 2017)	S	✓						VGG-16	First method based on online fine-tuning
MaskTrack (Perazzi et al. 2017)	S	✓			✓	✓		DLv2	First method based on mask propagation
VPN (Jampani et al. 2017)	S						✓	DLv1, BNN	BNN-based long-term propagation
CTN (Jang and Kim 2017)	S				✓	✓		VGG-16	Optical flow-based mask refinement
MP-Net (Tokmakov et al. 2017a)	U				✓			CNN, SharpMask	CNN-based motion patterns for UVOS
FusionSeg (Jain et al. 2017)	U				✓			ResNet-101	Fusion between motion and appearance
OnAVOS (Voigtlaender and Leibe 2017)	S	✓				✓		ResNet variant	Fine-tuning with online adaptation
PLM (Yoon et al. 2017)	S	✓	✓			✓		CNN	Multi-scale pixel-level matching
CCNN (Li et al. 2017b)	U		✓					VGG-16	Complementary segmentation modules
SegFlow (Cheng et al. 2017)	B	✓			✓			ResNet-101, FN 1.0	Multi-task for VOS and optical flow
VM-VOS (Tokmakov et al. 2017b)	U					✓	✓	DLv1, ConvGRU	Bidirectional GRU-based VOS
MaskRNN (Hu et al. 2017)	S	✓			✓	✓	✓	VGG-16, FN 2.0	BPTT-based mask propagation
OSVOS-S (Maninis et al. 2018)	S	✓						VGG-16, Mask-RCNN	Semantic propagation for SVOS
STVOS (Wang et al. 2018)	S		✓	✓				–	Point trajectory-based propagation
CINM (Bao et al. 2018)	S	✓			✓			OSVOS, FN 2.0, DLv2	Spatial-temporal MRF for SVOS
PML (Chen et al. 2018b)	S		✓					DLv2	Pixel-level matching for fast SVOS
FAVOS (Cheng et al. 2018)	S		✓					SiamFC, ResNet-101	Tracking object parts for SVOS
RCAL-VOS (Han et al. 2018)	S	✓				✓		DenseNet-56	Reinforcement learning for SVOS
MGCRN (Hu et al. 2018a)	B	✓			✓	✓		ResNet-101, FN 2.0	Motion-based cascade refinement
IET-VOS (Li et al. 2018b)	U			✓	✓			DLv2, FN 2.0	Formulate UVOS as finding seed tracks
RGMP (Oh et al. 2018)	S		✓		✓		✓	ResNet-50	Combine matching and mask propagation
MoNet (Xiao et al. 2018)	S	✓			✓	✓		DLv2, FN 2.0	Feature alignment via optical flow
OSMN (Yang et al. 2018)	S					✓		VGG-16	Introduce network modulation into VOS

Table 3 (continued)

Methods	Sup.	S. techs			T. techs			Architecture	Main contribution(s)
		O	M	G	O	P	L		
LSE-VOS (Ci et al. 2018)	S	✓				✓		ResNet-101	Pseudo labels and location embeddings
V-Match (Hu et al. 2018c)	S		✓			✓		ResNet-101	Efficient matching via soft matching

Sup supervision types, *U* unsupervised, *S* semi-supervised; *B* both of them, *S. techs* techniques for spatial features, *O* online fine-tuning, *M* matching, *G* graph; *T. techs* techniques for temporal features, *O* optical flow; *P* mask propagation; *L* long-term temporal propagation. *CNN* customised convolutional neural networks; *DL* DeepLab; *FN* FlowNet, *ResNet-V* ResNet variant

the first frame. However, the properties of the target objects (e.g., appearance, shape, scale and location) might change with the progress of video frames. Such variations are reflected in visual features, leading to false results on the corresponding regions. Figure 5A illustrates the effect of the challenge on the qualitative results.

3.2.2 Occluded by distractors

This challenge mainly affects the propagation-based VOS methods, which consider the objects predicted in the previous frame to estimate current frame segmentation. However, in actual scenes, the target objects are probably occluded by distractors, leading to partial losses on object regions. Such losses provide the incomplete estimation to the subsequent frames and make the occluded parts difficult to restore, even if the occlusions stop. Figure 5B illustrates the effect of the challenge on the qualitative results.

3.2.3 Distraction from similar objects/backgrounds

This challenge mainly affects the VOS methods based on visual similarities, saliency or motion patterns. During inference, these methods segment the objects with specific features, e.g., similar visual features to the target objects (most are SVOS methods) or prominent saliency/motion patterns (most are UVOS methods). However, these features are not always discriminative throughout video sequences. Taking one target object as an example, there are probably some ambiguous regions (might be backgrounds or other target objects) in the sequence, which do not belong to the object but have similar features. These regions could force the VOS methods to assign the labels of the example object to them, leading to false discrimination between similar target objects or over-segmentation. Figures 5C and D illustrate the effect of the challenge on the qualitative results.

3.2.4 Temporally consistent VOS

This challenge mainly affects the VOS methods using less motion information. During inference, these methods essentially perform image segmentation for each video frame. Therefore, it is difficult to maintain the temporally consistent of the segmented objects, i.e., the evolution of object masks predicted from continuous frames is not smooth (in the

Table 4 Summary of the discussed VOS methods (part 2/3)

Methods	Sup.	S. techs			T. techs			Architecture	Main contribution(s)
		O	M	G	O	P	L		
Dye-Net (Li and Change Loy 2018)	S	✓	✓		✓	✓	✓	ResNet-101, RPN, FN 2.0	ROI-matching, bidirectional propagation
SCO-VOS (Koh et al. 2018)	B			✓				FCIS	Clique optimisation for UVOS
MSGSTP (Hu et al. 2018b)	B			✓		✓		–	Saliency diffusion for UVOS
MBN-VOS (Li et al. 2018c)	U			✓	✓	✓		BNN, CNN, FN 2.0	Motion-based BNN & graph cut for VOS
S2S (Xu et al. 2018a)	S	✓					✓	VGG-16, ConvLSTM	Build long-term dependency over frames
PDB (Song et al. 2018)	U						✓	ResNet-50, ConvLSTM	Pyramid dilated, bidirectional ConvLSTM
PReMVOS (Luiten et al. 2018)	S	✓	✓		✓	✓		ResNet-V, Mask R-CNN, DLv3+, FN 2.0	Proposal generation, refinement, merging
LucidTracker (Khoreva et al. 2019)	S	✓			✓	✓		DLv2, FN 2.0	Data augmentation-based fine-tuning
BubbleNets (Griffin and Corso 2019)	S	✓						ResNet-50	Determine the optimal frame to annotate
FEELVOS (Voigtlaender et al. 2019)	S		✓			✓		DLv3	Combine global & local feature matching
SiamMask (Wang et al. 2019a)	S		✓			✓		ResNet-50	Light-weight tracking & segmentation
COSNet (Lu et al. 2019)	U		✓					DLv3	Use co-attention mechanism for UVOS
A-GAME (Johnander et al. 2019)	S					✓		ResNet-101	Gaussian mixture model-based SVOS
STCNN (Xu et al. 2019a)	S	✓					✓	ResNet-101, GAN	GAN and attention-based SVOS
MHPVOS (Xu et al. 2019c)	S	✓		✓	✓	✓		MR-CNN, DLv3+, FN 2.0	Tree structure optimisation for VOS
RVOS (Ventura et al. 2019)	B						✓	ResNet-101, ConvLSTM	Both spatial and temporal propagation
AGSS-VOS (Lin et al. 2019)	S		✓		✓	✓	✓	RGMP + FN 2.0	Efficient multi-object matching
RANet (Wang et al. 2019d)	S		✓			✓		ResNet-101	Feature ranking-based matching
DTN (Zhang et al. 2019)	S		✓		✓	✓		ResNet-50, FN 2.0	Local ROI generation, dynamic seg. network

Table 4 (continued)

Methods	Sup.	S. techs			T. techs			Architecture	Main contribution(s)
		O	M	G	O	P	L		
DMM-Net (Zeng et al. 2019a)	S	✓	✓				✓	Mask R-CNN, ConvLSTM	Optimal matching module
AD-Net (Yang et al. 2019b)	U		✓					DLv3	Global consistency, self-attention mechanism
AGNN (Wang et al. 2019b)	U			✓				GNN, DLv3, ConvGRU	Attentive GNN-based VOS
CapsVOS (Duarte et al. 2019)	S		✓				✓	CapsuleNet, ConvLSTM	Introduce CapsuleNet into SVOS
AGS (Wang et al. 2019c)	U						✓	ResNet-101, ConvLSTM	Prove key role of visual attention in UVOS

Sup supervision types, *U* unsupervised, *S* semi-supervised; *B* both of them, *S. techs* techniques for spatial features, *O* online fine-tuning, *M* matching, *G* graph; *T. techs* techniques for temporal features, *O* optical flow; *P* mask propagation; *L* long-term temporal propagation. *CNN* customised convolutional neural networks; *DL* DeepLab; *FN* FlowNet, *ResNet-V* ResNet variant

absence of occlusions, fast motion, or significant property change), which is unacceptable in some applications such as video editing. Figure 5E illustrates the effect of the challenge on the qualitative results.

3.2.5 Balance between VOS accuracy and efficiency

This challenge mainly affects the VOS methods serving real-time applications. Generally, these methods should perform the segmentation at least 24 FPS (Frames Per Second) while achieving high-quality object masks. However, the two goals cause conflict in the design of segmentation models. Efficient VOS prefers lightweight architecture. Instead, more sophisticated algorithms and network modules are generally required for accurate VOS. Figure 6 shows the performance of state-of-the-art methods in both accuracy and efficiency. It is observed that the balance between them remains under-explored.

4 Methods

This section reviews the existing deep learning-based SVOS and UVOS methods. As mentioned in Sect. 1, the SVOS methods segment the objects annotated in a few video frames (generally the first frame). In contrast, the UVOS methods segment the objects with prominent visual saliency or motion patterns. To generalise this discussion, we refer to the objects to segment in SVOS and UVOS methods as the "target objects" in the rest of the paper.

Generally, existing methods exploit the spatial and temporal features from input sequences to solve the SVOS and UVOS problems. The former features help maintain

Table 5 Summary of the discussed VOS methods (part 3/3)

Methods	Sup.	S. techs			T. techs			Architecture	Main contribution(s)
		O	M	G	O	P	L		
STM (Oh et al. 2019)	S		✓			✓		ResNet-50	Use intermediate frames for matching
TVOS (Zhang et al. 2020)	S		✓			✓		ResNet-50	Apply transductive inference in SVOS
SAT (Chen et al. 2020)	S		✓					ResNet-50	Dynamic tracking for fast SVOS
FRTM (Robinson et al. 2020)	S	✓						ResNet-50	Discriminative target model for SVOS
LWL (Bhat et al. 2020)	S	✓						ResNet-50	Differentiable and efficient few-shot learner
EGMN (Lu et al. 2020a)	B		✓	✓		✓		ResNet-50, ConvGRU	Graph-based memory for SVOS
KMN (Seong et al. 2020)	S		✓			✓		ResNet-50	Mutual matching between ref. & target frames
AFB-URR (Liang et al. 2020)	S		✓			✓		ResNet-50	Adaptive memory & refine by uncertainty
UnOVOST (Luiten et al. 2020)	U		✓	✓		✓		Mask R-CNN	Tracklet-based forest path cutting
CFBI+ (Yang et al. 2021b)	S		✓			✓		DLv3+	Discriminative features & multi-scale matching
SSTVOS (Duke et al. 2021)	S		✓			✓	✓	ResNet-101, Transformer	Transformer-based VOS & sparse attention
SwiftNet (Wang et al. 2021a)	S		✓			✓		ResNet-50	Adaptive memory & light architecture
LCM (Hu et al. 2021)	S		✓			✓		ResNet-50	Positional encoding & object relation
RMNet (Xie et al. 2021)	S		✓		✓	✓		ResNet-50, TinyFlowNet	Local to local matching
TAODA (Zhou et al. 2021)	U	✓					✓	ResNet-50, Mask R-CNN	Discriminative multi-object UVOS
HMMN (Seong et al. 2021)	S		✓			✓		ResNet-50	Multi-scale memory matching
STCN (Cheng et al. 2021)	S		✓					ResNet-50	Light architecture & efficient L2 distance
AOT (Yang et al. 2021a)	S		✓			✓		ResNet-50 / Swin-Transformer, multi-layer transformers	Uniform framework for multi-object VOS

Sup supervision types, *U* unsupervised, *S* semi-supervised; *B* both of them, *S. techs* techniques for spatial features, *O* online fine-tuning, *M* matching, *G* graph; *T. techs* techniques for temporal features, *O* optical flow; *P* mask propagation; *L* long-term temporal propagation. *CNN* customised convolutional neural networks; *DL* DeepLab; *FN* FlowNet, *ResNet-V* ResNet variant

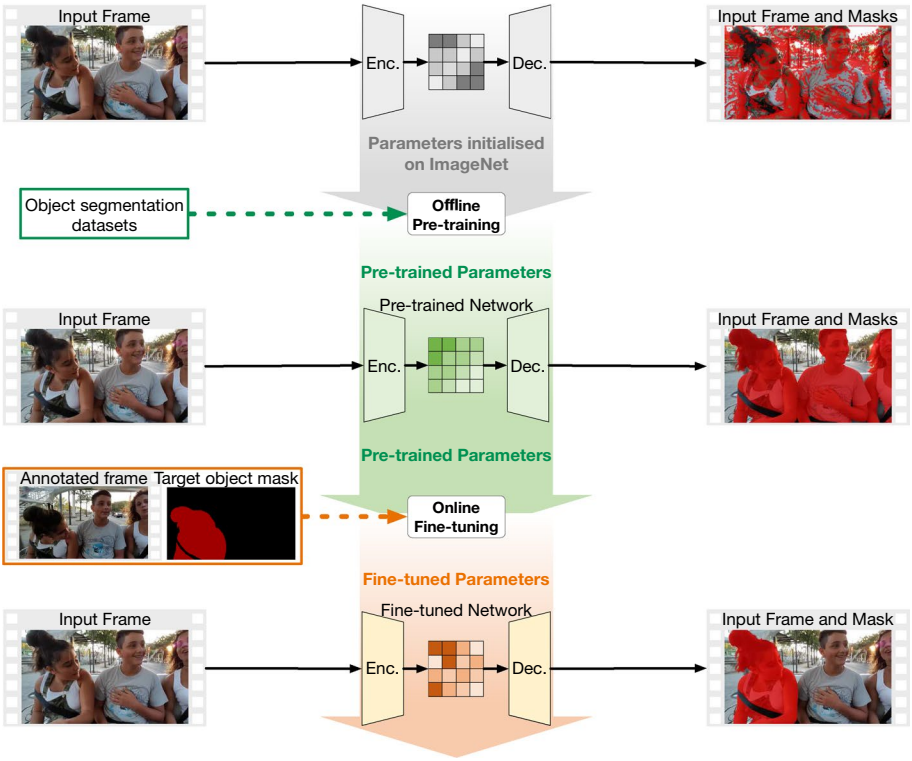

Fig. 7 Diagram of VOS method based on online fine-tuning. There are three main stages to shift the output domain of the segmentation network from general knowledge to the annotated object: (1) Initialise the network (coloured in gray) with the parameters pre-trained on ImageNet (Russakovsky et al. 2015; (2) pre-train the network (coloured in green) on object segmentation datasets (e.g. MS-COCO (Lin et al. 2014) and DAVIS (Perazzi et al. 2016a)); (3) Fine-tune the network (coloured in yellow) on the annotated frame. Since pre-training and fine-tuning are performed before and during the inference, we call them "offline" and "online" processes, respectively. Best viewed in colour

the consistent identities of the predicted objects throughout the video sequence. The latter features allow VOS methods to adapt to target object changes over time. Based on the existing model architectures, we elaborate on several techniques utilising these features. See Tables 3, 4, and 5 for more details.

The remainder of this section is organised as follows: Sects. 4.1, 4.2, and 4.3 introduce the techniques for spatial features, including online fine-tuning, feature matching, and graph optimisation, as well as their representative methods. Sections 4.4, 4.5, and 4.6 introduce the techniques for temporal features, including optical flow, mask propagation, and long-term temporal information, as well as their representative works.

4.1 Online fine-tuning-based methods

Tables 3 and 4 demonstrate that most earlier SVOS methods are implemented with online fine-tuning. Given the annotated objects and a network for general object segmentation, this technique fine-tunes the network with the annotated objects. In this way, the network can "memorise" the properties of the annotated objects (e.g., appearance,

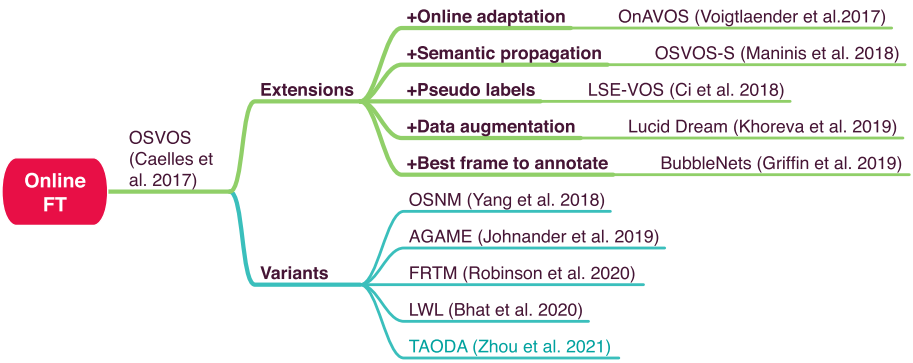

Fig. 8 Development roadmap of representative online fine-tuning-based methods. OSVOS is the first method using online fine-tuning for SVOS. After that, several methods are derived from OSVOS. Their main modifications to OSVOS are highlighted by bold words with the prefix "+". Recently, several variants of online fine-tuning are proposed for efficient VOS. Note that the blue and black words indicate the methods performing UVOS and SVOS, respectively

shape, and categories) and transfer its output domain from general objects to the annotated objects. The diagram of this technique is shown in Fig. 7, from which the segmentation for an input video sequence can be summarised: (1) fine-tune the pre-trained network on the annotated frame; (2) use the fine-tuned network to perform segmentation on the rest of the frames. In general, this technique does not apply to UVOS since no annotated frame is available in UVOS. However, with the integration of instance-level approaches, some UVOS methods perform instance segmentation for the first frame. Then, the predicted objects are utilised to fine-tune the segmentation network.

This section discusses several representatives and variants of online fine-tuning-based methods. In Sect. 4.1.1, we first introduce OSVOS (One-shot VOS), the first method for online SVOS fine-tuning. Next, the methods derived from OSVOS are reviewed in Sect. 4.1.2. Finally, Sect. 4.1.3 introduces several variants of online fine-tuning for efficient VOS. Section 4.1.4 summarises the discussed methods, whose development roadmap is shown in Fig. 8.

4.1.1 OSVOS (One shot video object segmentation)

OSVOS, proposed by Caelles et al. (2017), is the earliest SVOS method based on online fine-tuning. The pre-training and fine-tuning for the segmentation network are the same as shown in Fig. 7. During inference, OSVOS firstly fine-tunes the pre-trained network with the first frame annotation. Next, each frame to segment is fed into the fine-tuned network, achieving the initial prediction, which is then refined by a contour detection network.

The competitive results achieved on DAVIS-2016 (Perazzi et al. 2016a) and YouTube-Objects (Prest et al. 2012) prove the effectiveness of online fine-tuning on SVOS. However, there are still challenges that OSVOS cannot handle well, for example, the object property change and ambiguous regions. Since OSVOS performs fine-tuning under the guidance of the first frame annotations only, the fine-tuned network is easy to get overfitting and therefore cannot sufficiently adapt to the object changes. In addition, the fine-tuned network is prone to be misled by the regions that look similar to the annotated objects.

Table 6 Summary of OSVOS and the extensions

Methods	Frames	Data augmentation	Adaptation
OSVOS	1st	Standard	NOT considered
OnAVOS	ALL	Standard	Fine-tune on the high-confident results
OSVOS-S	1st	Standard	Fuse the outputs of the fine-tuned network and Mask R-CNN
LSE-VOS	ALL	Standard	Fine-tune on the pre-segmented results
LucidTracker	1st	Generate 2,500 pairs of temporal continuous frames	NOT considered
BubbleNet	Optimal	standard	NOT considered

Frames: Frames used to fine-tune the network; Data augmentation: Strategies used to augment the available data, where standard means the commonly used strategy in segmentation tasks; Adaptation: Strategies used to adapt the network to the object change

4.1.2 Extensions

On top of OSVOS, several extension works have been developed to sufficiently utilise online fine-tuning and handle the challenges above. Table 6 briefly summarises OSVOS and the extensions in terms of their properties in fine-tuning.

OnAVOS (Online Adaptive VOS, Voigtlaender and Leibe 2017) improves OSVOS with the online adaptive strategy. Unlike OSVOS, OnAVOS fine-tunes the network further with the high-confident results and definite backgrounds from the predicted frames. **Discussion**: OnAVOS has more adaptability and can handle the distraction from backgrounds. However, such refinement is time-consuming since multiple online fine-tuning is required during inference.

OSVOS-S (*OSVOS-Semantic*, Maninis et al. 2018) improves OSVOS with semantic information. The segmentation is achieved by merging the masks predicted by OSVOS and Mask R-CNN (He et al. 2017), where the target object categories are inferred from the first frame annotation and used to filter out irrelevant objects. **Discussion**: Masks with fewer missing parts can be achieved since the advantage of Mask R-CNN in object mask generation. However, such improvement is achieved at the cost of high GPU computation due to multiple deep networks.

Location-Sensitive Embeddings for VOS (Ci et al. 2018, abbreviated as 'LSE-VOS') performs SVOS by segmenting ALL video frames twice. The first round of results serves as the training samples for further fine-tuning. In addition, an LSE module distinguishes target objects from ambiguous backgrounds. **Discussion**: Similar to OnAVOS, more training samples bring better adaptability to the network, and the LSE module improves the robustness against clutter backgrounds. However, LSE-VOS is inefficient since twice SVOS is required.

Lucid Data Dreaming (Khoreva et al. 2019, abbreviated as 'LucidTracker') enhances the data augmentation, where 2,500 video clips are generated from the first frame annotation. Each clip contains two temporally continuous image-mask pairs to support temporal learning. **Discussion**: Unlike other online fine-tuning-based methods, LucidTrack can generate the training data with more diversity, enabling the fine-tuned network to step closer to the target domain, even if the network is randomly initialised. However, data generation from only one frame is also a burden for LucidTrack since the resulting network adaptability is limited.

Table 7 Summary of the representative variants of online fine-tuning

Methods	Techniques	Parameters
OSNM	Conditional batch normalisation	γ and β in Equation 2
A-GAME	Gaussian mixture model	$\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ in Equation 3
FRTM	Target model (predict coarse mask)	Target model parameters
LWL	Target model (predict multi-channel mask)	Target model parameters
TAODA	Target model (predict coarse mask)	Target model parameters

Techniques: Techniques used to replace online fine-tuning; Parameters: Parameters to optimise according to the annotated frames

BubbleNets (Griffin and Corso 2019) improves the online fine-tuning-based VOS from a novel perspective. Until 2018, we found almost all such methods (e.g., OSVOS (Caelles et al. (2017)), OnAVOS (Voigtlaender and Leibe 2017), OSVOS-S (Maninis et al. 2018), LSE-VOS (Ci et al. 2018)) fine-tune their networks with the first frame annotation. For offline applications with all frames available (e.g., video editing), annotating the first frame probably can not bring the best fine-tuning results. The authors identified the problem and proposed BubbleNets, predicting the optimal frame to annotate. **Discussion**: With BubbleNets, the average performance of OSVOS and OnAVOS improves by 3.5% and 5.95% on the DAVIS-2016 and DAVIS-2017 validation sets. Such improvement shows that BubbleNets can be an incremental module for online fine-tuning-based methods.

4.1.3 Variants

The above extensions improve the VOS results. However, the segmentation is inefficient since online fine-tuning is costly in time and computation. To address this issue while shifting the network output domain, current methods implement more efficient algorithms to optimise part of segmentation networks instead of fine-tuning the whole network via backward propagation. Similar to online fine-tuning, these methods also modulate the network parameters according to the annotated frames. Therefore, we consider them as variants of online fine-tuning-based methods. Table 7 briefly summarises these variants.

OSNM (*Object Segmentation via Network Modulation*, Yang et al. 2018) modulates the network with a conditional batch normalisation (CBN, De Vries et al. 2017)-based module:

$$\mathbf{y} = \gamma \mathbf{x} + \boldsymbol{\beta}, \tag{2}$$

where the intermediate feature $\mathbf{x}$ is converted to $\mathbf{y}$ under the guidance of the first frame annotations γ and previous frame masks $\boldsymbol{\beta}$. **Discussion**: The network output domain can be shifted with a single forward pass, which is much more efficient than online fine-tuning. But limited by the less information modulated, the segmentation accuracy is not as perfect as efficiency.

A-GAME (*A Generative Appearance Model for End-to-end VOS*, Johnander et al. 2019) adapts its network to the target object with a Gaussian mixture model:

$$p(z_p = k|\mathbf{x}_p, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{p(z_p = k) \cdot \mathcal{N}(\mathbf{x}_p|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_k p(z_p = k) \cdot \mathcal{N}(\mathbf{x}_p|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}, \tag{3}$$

where the probability that a pixel p belongs to the k^{th} component (either target object or background) depends on its appearance $\mathbf{x}_p$, $p(z_p = k) = 1/K$ (K: number of components), $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$. The last two magnitudes are initialised from the first frame annotation and then updated with the past frame outputs. **Discussion**: Unlike OSNM, A-GAME exerts a more explicit and continuous influence on the segmentation network, leading to higher-quality results.

FRTM (*Fast and Robust Target Models for VOS*, Robinson et al. 2020) designs a discriminative linear model for generating target-specific predictions, which are then refined by a segmentation network. During inference, only the target model requires training, achieved by performing the Gauss-Newton-based optimisation (Tjaden et al. 2018) on the first frame annotation and subsequent frame predictions. **Discussion**: Due to the lightweight target model and efficient optimisation, FRTM performs SVOS faster than online fine-tuning-based methods. In addition, unlike OSMN and A-GAME, FRTM has much more target-specific parameters, boosting the discrimination between target objects and backgrounds.

LWL (*Learn What to Learn*, Bhat et al. 2020) designs a similar pipeline to FRTM. However, the target model in LWL predicts a multi-channel mask for each object and is optimised by a differentiable approach. Specifically, LWL generates the ground truth masks and weight matrices (used to balance the target/background regions) from two learnable modules rather than the annotation. **Discussion**: LWL trains these two modules offline, enabling them to learn how to generate the optimisation goals (i.e., learn what to learn) for the target model, according to the loss of final segmentation results. Therefore, LWL achieves more robust results than FRTM while keeping competitive efficiency.

TAODA (*Target-Aware Object Discovery and Association for UVOS*, Zhou et al. 2021) implements a similar target model to FRTM to generate coarse object masks. Differently, the target model is initialised with the instances predicted in the first frame due to no annotations available in UVOS. **Discussion**: With the instance-level segmentation module, TAODA can perform multi-object UVOS. Unlike other UVOS methods, the target model provides a good prior for each object to segment, reducing the efforts for associating objects throughout the sequence.

4.1.4 Summary

This section introduces the online fine-tuning-based VOS methods and their variants. These methods originate from OSVOS (Caelles et al. 2017), which shifts the network output domain from general knowledge to the target objects by fine-tuning the network parameters with the first frame annotations. The extension works and variants are mainly motivated by the following issues in OSVOS: (1) Only the first frame annotations are considered, limiting the adaptivity of the fine-tuned network and resulting in overfitting; (2) Online fine-tuning is time-consuming, limiting the VOS efficiency.

The extension works improve OSVOS in adaptivity and robustness. For example, OnA-VOS (Voigtlaender and Leibe 2017) and LSE-VOS (Ci et al. 2018) improve the adaptivity by incorporating the high-confident results from the past frames. However, multiple online fine-tuning is required during inference, reducing the VOS efficiency further. Without extra online fine-tuning, OSVOS-S (Maninis et al. 2018) and LucidTracker (Khoreva et al. 2019) enhance the segmentation robustness. OSVOS-S incorporates the knowledge from general object segmentation to refine the VOS results. In contrast, LucidTracker achieves this by

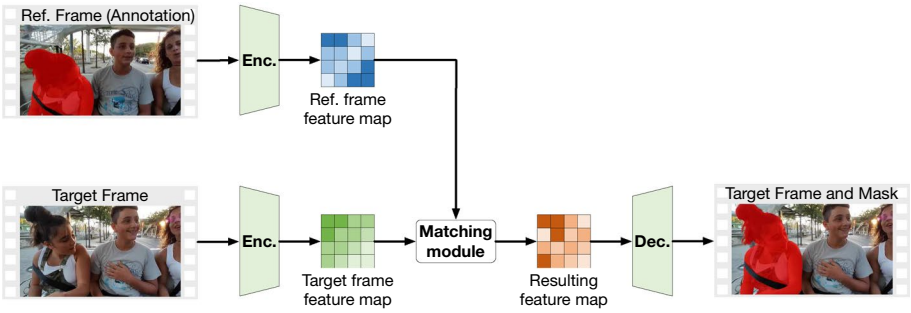

Fig. 9 Diagram of matching-based VOS methods, which apply to both SVOS and UVOS applications. Since there are no annotations available in UVOS, the top branch mainly takes extra inputs (i.e., object masks) for SVOS methods

generating far more diverse samples from the annotations. These methods promote the VOS performance but still suffer less efficiency since extra computation is required for Mask R-CNN or data augmentation. Unlike the above methods, BubbleNets (Griffin and Corso 2019) is an incremental method, which can be incorporated into the above methods to predict the optimal frame to annotate. Indeed, such an optimal frame is also generated at the cost of efficiency. Therefore, the online fine-tuning-based VOS applies to offline applications that emphasise segmentation accuracy instead of efficiency.

The variants focus more on the VOS efficiency. Without online fine-tuning, the discussed variants (OSNM (Yang et al. 2018), A-GAME (Johnander et al. 2019), FRTM (Robinson et al. 2020), LWL (Bhat et al. 2020), and TAODA (Zhou et al. 2021)) have developed to shift the network output domain with more efficient algorithms. Although achieving better efficiency, the accuracy gaps remain between the earlier variants (OSNM and A-GAME) and the extension works. From Tables 16 and 17, it is observed that the target model-based variants (FRTM and LWL) further improve the SVOS performance. Also, such improvement is brought to UVOS (TAODA). Based on Table 7, we find the reason for the performance improvement might lie in the number of adjustable parameters during inference since the target models in FRTM, LWL, and TAODA have much more adjustable parameters than OSNM and A-GAME. In other words, the number might reflect the ability to encode discriminative/target-specific features in VOS methods.

4.2 Matching-based method

This method performs VOS by measuring the correspondence between the target and reference frames, as shown in Fig. 9. The "target frame" indicates the video frame to segment and "reference frame" has different meanings by VOS types. In SVOS, the "reference frame" consists of the annotated frame and part/none of past frames. The goal of segmentation methods is to propagate the reference frame masks to the target frame, according to the measured correspondence. In contrast, the "reference frame" in UVOS might include any past/future frames. UVOS methods locate the objects appearing in both target and reference frames based on the measured correspondence.

The key to reliable and robust correspondence is discriminative feature embeddings. Most existing methods achieve this by training backbone networks on large-scale image/video datasets or utilising off-the-shelf object detection/segmentation approaches. Since all

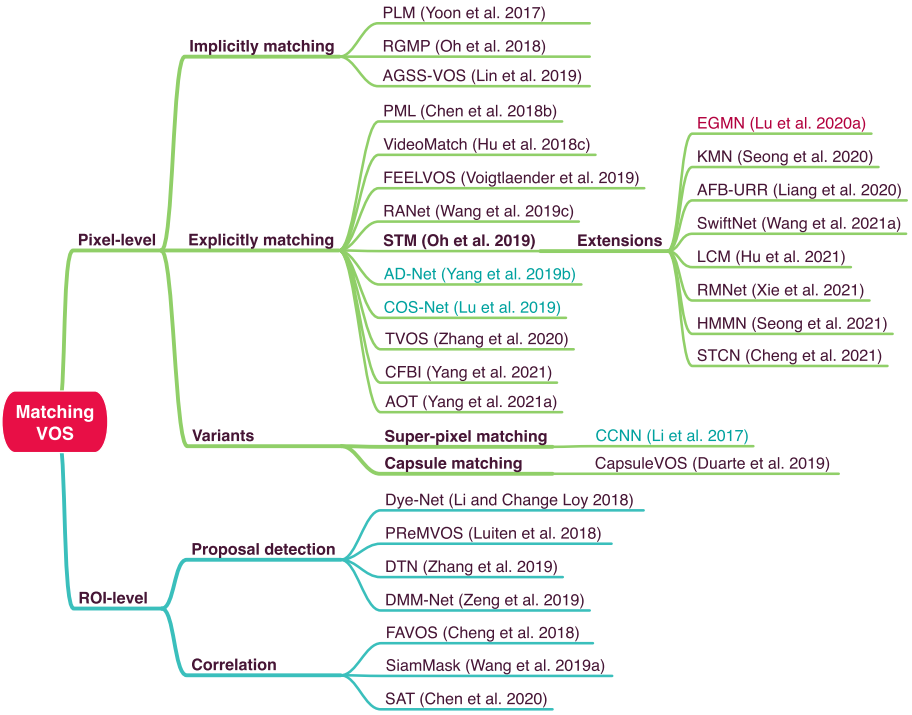

Fig. 10 Development roadmap of representative matching-based methods. Note that the blue, black, and red words indicate the methods performing UVOS, SVOS, and both, respectively. STM (Oh et al. 2019) is marked in bold words due to its breakthrough innovation and numerous extensions. Best viewed in colour

these works can be completed offline, the matching-based method executes much faster than online fine-tuning while achieving better results. Therefore, the most recent VOS methods (especially SVOS) are based on feature matching, as shown in Tables 3, 4, and 5.

This section introduces the representative matching-based methods, whose development roadmap is shown in Fig. 10. Based on different feature types, the discussed methods are grouped into two categories: pixel-level matching and ROI (region of interest)-level matching. The former (Sect. 4.2.1) measures the pixel-level feature correspondence between frames. By contrast, the latter (Sect. 4.2.2) only focuses on the ROI-level correspondence. Section 4.2.3 summarises the discussed methods.

4.2.1 Pixel-level matching

This method performs VOS by measuring the pixel-level correspondence between frames. As shown in Fig. 11, there are two schemes to implement the pixel-level matching module: implicitly matching and explicitly matching. The former designs a network module to predict the cross-frame similarities implicitly, and the latter explicitly matches the features between each pair of pixels between frames. Given different implementations, we first introduce the methods based on implicit matching, followed by those based on explicit matching. Finally, two variants of pixel-level matching are discussed. Table 8 briefly summarises the discussed methods.

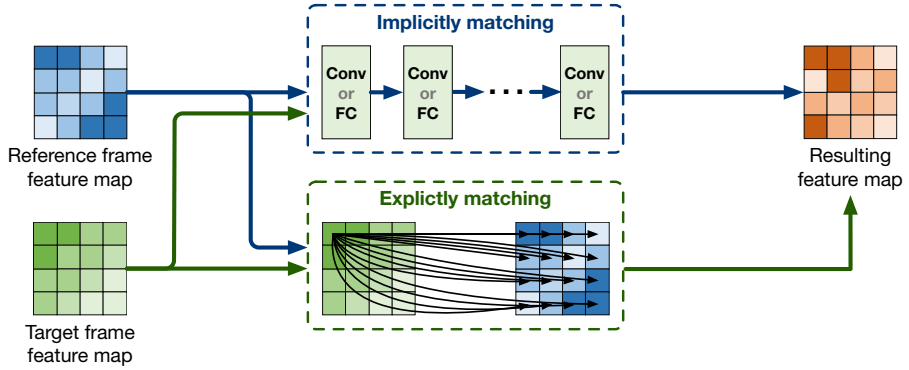

Fig. 11 Diagram of different pixel-level matching modules. Given the reference and target feature maps, the top module feeds them into a set of CNN/FC layers to perform implicit matching. In contrast, the bottom module directly measures the similarities of all pairs of points between feature maps

Table 8 Summary of the discussed pixel-level matching methods

Methods	Reference Frames	Matching Schemes	Measurements
PLM	1	Local	Fully connected layers
RGMP	1	Local	Convolutional layers
AGSS-VOS	1	Local	Convolutional layers
PML	PR (1st frame, high-confident results)	Global	L2 similarity
VideoMatch	1	Global	Cosine similarity
FEELVOS	1, t-1	Local, Global	$1-\frac{2}{1+\exp(\|\|e_p-e_q\|\|^2)}$
RANet	1	Global	Dot product
STM	1, t-1, (1, t-1, 5)	Global	Dot product
EGMN	1, t-1, 4 intermediate frames (uniformly sampled)	Global	Dot product
KMN	1, t-1, (1, t-1, 5)	Local, Global	Dot product
AFB-URR	PR (1st frame, diverse features)	Global	Dot product
SwiftNet	PR (1st frame, diverse features)	Global	Dot product
LCM	1, t-1, (1, t-1, 5)	Local, Global	Dot product
RMNet	1, t-1, (1, t-1, 5)	Local, Global	Dot product
HMMN	1, t-1, (1, t-1, 5)	Local, Global	Dot product
STCN	1, (1, t-1, 5)	Global	L2 similarity
TVOS	[t-4, t-1], 5 frames sparsely sampled from [1, t-5]	Local, Global	Weighted dot product
CFBI+	1, t-1	Local, Global	$1-\frac{2}{1+\exp(\|\|e_p-e_q\|\|^2+b)}$
AOT	1, t-1, (1, t-1, 5)	Local, Global	Dot product
COSNet	5 frames uniformly sampled from the input sequence	Global	Weighted dot product
AD-Net	1, t	Global	Dot product

The first item indicates the indices of reference frames. Note that "PR" (Point-based Reference) means the reference consists of feature points instead of feature maps. "(1, t-1, 5)" indicates the frames sampled between the 1st and previous frame (these two frames not included) with the interval=5. e_p and e_q are the pixel-level features from different frames. b is a trainable bias

(1) Implicit matching. PLM (Pixel-Level Matching, Yoon et al. 2017*)* is one of the earliest matching-based SVOS methods. The method performs multi-scale matching between the previous and target frames to segment the target objects. To suppress ambiguous backgrounds, only the regions around the previous frame objects are considered. **Discussion**: Although achieving good efficiency, the segmentation results are vulnerable to fast motion and occlusion since PLM only performs local matching between frames.

RGMP (*Reference-Guided Mask Propagation,*Oh et al. 2018) performs matching between the first and target frames. Unlike PLM, RGMP encodes object masks together with video frames, implicitly merging the object appearance and location priors. **Discussion**: Due to the enhanced data augmentation and effective feature fusion, RGMP performs well in both efficiency and accuracy. However, RGMP is inefficient in multi-object segmentation since repeat runs are required.

AGSS-VOS (*Attention Guided Single-Shot VOS*, Lin et al. 2019) improves the matching module in RGMP to process ALL objects in a single feed-forward path. Multiple target objects are then separated by a lightweight target-specific module. **Discussion**: Although the latter module requires repeat runs, its relatively lightweight structure still mitigates computation costs. However, the overall architecture becomes more complicated due to integrating extra modules for object separation and optical flow.

(2) Explicit matching. PML (*Pixel-wise Metric Learning*, Chen et al. 2018b) is one of the earliest deep methods using pixel-wise similarities. Each pixel in the target frame is labelled based on its top-k similarities with all reference pixels, which are initialised with the first frame annotation and updated by high-confident results. **Discussion**: Since no decoding is required after the matching-based labelling, PML achieves high efficiency in SVOS. However, PML cannot handle ambiguous backgrounds because it relies heavily on matching results.

VideoMatch (Hu et al. 2018c) implements a soft matching strategy for SVOS. Unlike PML, VideoMatch utilises the averaged top-k similarities to generate the final results. Also, an outlier removal module is proposed to filter out ambiguous backgrounds far from the previous results. **Discussion**: VideoMatch is robust against clutter scenes than PML. However, such robustness is limited since VideoMatch still relies heavily on matching results.

FEELVOS (*Fast End-to-End Learning for VOS*, Voigtlaender et al. 2019) considers matching results as implicit guidance instead of directly propagating labels based on them (as in PML and VideoMatch). Specifically, FEELVOS performs global/local matching between the first/previous frames and the target frame. The matching results are then fused with semantic features to predict the final results. **Discussion**: Unlike the above methods, FEELVOS further mitigates the distractions from backgrounds. However, the adaptability in FEELVOS is limited since only the first and previous frames are referenced.

RANet (*Ranking Attention Network*, Wang et al. 2019d) only considers the high-confident instead of all reference features during matching. To this end, RANet implements a ranking attention module to select the reliable features contributing to the segmentation. **Discussion**: Compared with other methods, RANet filters out the potential noises, achieving more robustness against complex scenes. However, like FEELVOS, RANet suffers less adaptability due to the limited reference frames.

STM (**Space-Time Memory network**, Oh et al. 2019) makes a great breakthrough in SVOS. Besides the first and previous frames, STM also considers the intermediate frames between them as the reference to adapt the object changes better. Unlike the above methods, STM performs SVOS with an attention-like scheme: (1) Encode key/value features from the target and reference frames; (2) Measure the key similarities between frames; (3)

Consider the similarities as weights to sum the reference values; (4) Concatenate the target values with the summed values and feed them into a decoder to predict the final outputs. Owing to its efficient yet comprehensive matching scheme, STM achieved state-of-the-art performance on all the benchmark datasets at the time of publication while keeping relatively low computation costs.

The success of STM motivates many recent methods, which improve the SVOS performance from the following aspects: (1) **Matching scheme**: STM only considers global matching between frames, which is vulnerable to similar target objects/backgrounds; (2) **Encoder architecture**: STM applies different encoders to the target and reference frames due to different input channels (Target frame: RGB; Reference frames: RGB+mask), increasing the architecture redundancy; (3) **Memory management**: With the progress of the segmentation, the number of intermediate frames in STM improves increasingly, resulting in out-of-memory when segmenting long videos; (4) **Temporal correspondence between reference frames**: STM considers all reference frames individually and ignores the temporal correspondence between them; (5) **Similarity metrics**: STM employs dot product for similarity measurement. However, the recent work demonstrates that dot product might reduce the utilisation of the reference features. (6) **Multi-scale matching**: STM only performs matching between the coarsest feature maps and ignores the fine-grained correspondence, limiting further performance improvement. These extension works are briefly introduced as follows:

EGMN (*Episodic Graph Memory Network*, Lu et al. 2020a) improves STM in temporal correspondence. To this end, the method builds a fully-connected graph over the reference frames and performs information propagation between them. **Discussion**: Due to the enhanced reference, EGMN outperforms STM on some benchmarks. However, such improvement is achieved at the cost of efficiency since the graph-based propagation is time-consuming.

KMN (*Kernelized Memory Network*, Seong et al. 2020) improves STM in matching scheme. The method assumes each reference point has a candidate area in the target frame, represented by a 2D Gaussian kernel centred at its most similar point in the target frame. **Discussion**: The Gaussian kernels limit the effects of some ambiguous objects/backgrounds on the segmentation results, enabling KMN to outperform STM. However, the regions with strong ambiguity (e.g., objects with the same class as the target object) remain if they are mutually matched with the reference points.

AFB-URR (*Adaptive Feature Bank-Uncertain Region Refinement*, Liang et al. 2020) improves STM in memory management. To this end, the method creates a fixed-size memory bank for efficient space utilisation. The final outputs are refined by an uncertainty-based module. **Discussion**: SVOS for long videos is supported in AFB-URR. Under the same training setting, AFB-URR achieves similar performance to STM with less memory space and computation.

SwiftNet (Wang et al. 2021a) improves STM in memory management and encoder architecture. The method utilises a similar strategy to AFB-URR to build the memory bank, but different update triggers are used. In addition, since all reference frames (except the first frame) were encoded when they were the target frames, SwiftNet implements a lightweight module to encode the reference features on top of the encoded target features instead of scratch. **Discussion**: Unlike the above STM-based methods, SwiftNet predicts competitive results with fewer parameters and therefore achieves the state-of-the-art balance between SVOS accuracy and efficiency.

LCM (Hu et al. 2021) improves STM in matching scheme. Unlike the above STM-based methods, which perform global matching between all reference frames and the target

frame, LCM performs local matching between the previous and target frames. The local matching is achieved by appending the positional encoding (Parmar et al. 2018) to the reference and target keys. **Discussion**: LCM implicitly encourages the label propagation between the spatially neighbouring points, which fits the nature of the object movement in videos and helps suppress the distant ambiguous backgrounds.

RMNet (*Regional Memory Network*, Xie et al. 2021) improves STM in matching scheme. Unlike LCM, RMN performs local matching with a more straightforward approach, only involving the reference and target frame features within ROIs. The ROIs in the reference and target frames are generated based on the predicted object masks and optical flow, respectively. **Discussion**: RMN is robust against distant ambiguous backgrounds. However, such improvement is achieved at the cost of extra computation (optical flow).

HMMN (*Hierarchical Memory Matching Network*, Seong et al. 2021) improves STM in multi-scale matching. Besides the feature matching on the coarsest scale (with a stride of 16), HMMN considers the pixel-wise correspondence on fine-grained scales (with strides of 4 and 8). Moreover, HMMN refines a kernelised approach (as in KMN) with temporal constraints to generate candidate locations for each reference point. **Discussion**: Multi-scale matching brings significant performance improvement to the STM-based SVOS. On top of the initial kernelised approach, the temporal smoothness further mitigates the distraction from ambiguous objects/backgrounds.

STCN (*Space-Time Correspondence Network*, Cheng et al. 2021) improves STM in encoder architecture and similarity metrics. In STCN, two issues in all the above STM-based methods are recognised: (1) the reference and target keys (the features to match) are computed by different encoders, which distracts the pixel-wise correspondence between frames; (2) cosine distance limits the utilisation of reference features. To address the issues, STCN encodes keys with a shared architecture, followed by a lightweight module to encode values. As for the similarity metric, STCN implements an efficient L2 distance to replace the cosine distance. In addition, STCN removes the previous frame from the reference to avoid drifting errors. **Discussion**: With the above efforts, STCN significantly improves STM in efficiency and accuracy. Since the effective and simple architecture and the remaining challenges (e.g., vulnerable to ambiguous backgrounds; feature matching on the coarsest scale only), STCN can serve as a new baseline method in SVOS.

Recently, most methods have achieved competitive results via the memory-based architecture. However, there are still methods exploring different ways for VOS, increasing the diversity of this field. These methods are introduced as follows:

TVOS (*Transductive VOS*, Zhang et al. 2020) incorporates transductive inference (Zhou et al. 2004) into SVOS. Unlike STM, TVOS considers a fixed number of frames as the reference and the similarities are constrained by both spatial and temporal distances between pixels. **Discussion**: Therefore, TVOS suppresses more distracting regions. Since TVOS is only trained on video datasets, the overall performance is not competitive with STM and its variants but better than the methods with the same training settings.

CFBI (Collaborative VOS by Foreground-Background Integration, Yang et al. 2020) only considers the first and previous frames as the reference. Unlike other matching-based methods, CFBI treats the foreground and background regions equally, implicitly enhancing the discriminability of the encoded features. By integrating the Feature Pyramid Network (Lin et al. 2017) into the segmentation network, CFBI is extended to handle multi-scale target objects (CFBI+, Yang et al. 2021b). **Discussion**: Compared with STM and its variants, CFBI and CFBI+ achieve similar performance with fewer reference frames due to the powerful feature embedding.

Table 9 A summary of the discussed variants

Methods	Reference Frames	Matching Entities	Measurements
CCNN	t-30, t-15, t+15, and t+30	Super-pixels	L2 similarity
CapsuleVOS	1	Capsules	L2 similarity

The second item indicates the utilised entities during matching, the other two items have the same meanings as in Table 8

AOT (Associating Objects with Transformers, Yang et al. 2021a) introduces the multi-layer transformer module to build the correspondence between frames. Unlike most methods in this section, which process each target object separately, AOT implements a uniform mechanism to encode, match, and decode multiple objects. **Discussion**: This mechanism encourages AOT to better mine and exploit the relationships between objects, leading to superior performance on several benchmark datasets. The success of AOT provides a perfect example to show how transformers boost the matching-based SVOS methods.

Besides SVOS methods, feature matching also brings competitive results to UVOS methods. According to the pixel-level correlation between frames, matching-based UVOS methods can efficiently localise the frequently reappearing objects (even static objects) from the sequence.

COSNet (*Co-Attention Siamese Networks*, Lu et al. 2019) is a UVOS method, which resolves the salient object by a co-attention module:

$$\mathbf{S} = \mathbf{X}_r^{\mathrm{T}} \mathbf{W} \mathbf{X}_t, \tag{4}$$

where $\mathbf{W}$ is a learnable weight matrix, $\mathbf{X}_r$, $\mathbf{X}_t$ are the features from reference and target frames. **Discussion**: Equation 4 focuses on the pair-wise relationship between frames. Although aggregating multiple $\mathbf{S}$ can achieve more global relationships throughout the sequence, the resulting information is still limited since the module is trained with pairs of frames. To address this problem, Lu et al. (2020b) extended COSNet by training the module with groups of frames. Specifically, $\mathbf{X}_r$ is the concatenation of multiple reference frames instead of a single one. Therefore, the module is encouraged to encode more rich correspondence among multiple frames.

AD-Net (*Anchor-Diffusion Network*, Yang et al. 2019b) performs UVOS with both cross-correlation and auto-correlation to localise the salient object:

$$\mathbf{S}_{\text{cross}} = \mathbf{X}_1 \mathbf{X}_t^{\mathrm{T}}, \ \mathbf{S}_{\text{auto}} = \mathbf{X}_t \mathbf{X}_t^{\mathrm{T}}, \tag{5}$$

where $\mathbf{X}_1$, $\mathbf{X}_t$ are the features from the reference and target frames. **Discussion**: Unlike COSNet, AD-Net only considers the first frame as the reference, accelerating the correlation computation. Also, auto-correlation brings more discriminative feature embedding. However, AD-Net works only when the foreground object appears in the first frame and cannot apply to the sequences with an empty first frame.

(3) Variants. So far, all the discussed methods in this section are developed based on pixel-level matching, whose basic elements are feature points. Generally, each point represents a subregion with a regular shape in the video frame. Here some variants based on irregular elements (e.g., super-pixels) are discussed. Table 9 briefly summarises these variants.

CCNN (*Complementary CNNs*, Li et al. 2017b) performs UVOS based on the super pixel-wise similarities between frames. During inference, CCNN propagates the initial foreground/background labels (predicted based on visual saliency) along the paths built from the similarities. **Discussion**: Compared with grid regions, super-pixel regions fit better with the object boundaries, enabling more detailed results. However, temporal embedding is less explored in CCNN since the deep networks for initial result generation are trained on image datasets only.

CapsuleVOS (*Capsule-based VOS*, Duarte et al. 2019) is one of the earliest SVOS methods with capsule networks (Hinton et al. 2018). Since their abilities to establish part-to-object relationships, capsule networks have been successfully applied in pixel-wise classification tasks (e.g., object segmentation (LaLonde and Bagci 2018) and salient object detection (Liu et al. 2019)). In CapsuleVOS, a capsule implicitly represents an object/object part, whose potential region in the target frame is predicted by a routing algorithm, according to the capsule-level similarities between the reference and target frames. **Discussion**: Capsules bring more effective object modelling. In addition, CapsuleVOS supports parallel segmentation, where multiple video frames can be segmented at once during training and inference.

4.2.2 ROI-level matching

This section reviews the representative VOS methods based on ROI-level matching. Unlike pixel-level matching, this method can better handle the distractions from noisy pixels. Specifically, this method first extracts a set of object proposals from the target frame. Then, these proposals are filtered by comparing them with the objects in the reference frames. Finally, the method feeds the filtered proposed into an ROI-based segmentation module to predict object masks.

Based on different mechanisms for proposal generation, existing methods can be divided into two types: proposal detection and correlation. The former generates object proposals via off-the-shelf object detection (e.g., Region Proposal Network, Ren et al. 2015) or instance segmentation (e.g., Mask R-CNN, He et al. 2017) approaches. The latter is inspired by object tracking approaches (e.g., SiamFC, Bertinetto et al. 2016), which compute the ROI-level correlations between frames to generate object proposals. Fig. 12 shows the diagram of two types of ROI-level matching. Table 10 briefly summarises the discussed methods.

(1) Detection-based ROI matching. DyeNet (Li and Change Loy 2018) is one of the earliest methods based on ROI matching. The ROIs are generated by RPN. To localise the target object, DyeNet measures the similarities between the target frame ROIs and reference frame ROIs, which are initialised with the first frame annotations and then updated with the high-confident ROIs in subsequent frames. **Discussion**: Since the intermediate results are considered as the reference, DyeNet is adaptive to the changes of the target objects.

PReMVOS (*Proposal-generation, Refinement and Matching for VOS*, Luiten et al. 2018) performs SVOS with almost all spatial and temporal techniques. Unlike DyeNet, PReMVOS predicts coarse masks (by Mask R-CNN) instead of bounding boxes for each target object. These masks are then refined by previous frame masks, optical flow, and a DeepLab-based refinement module. **Discussion**: PReMVOS outperformed most VOS methods at the time but the computation cost is high due to the integration of multiple deep networks.

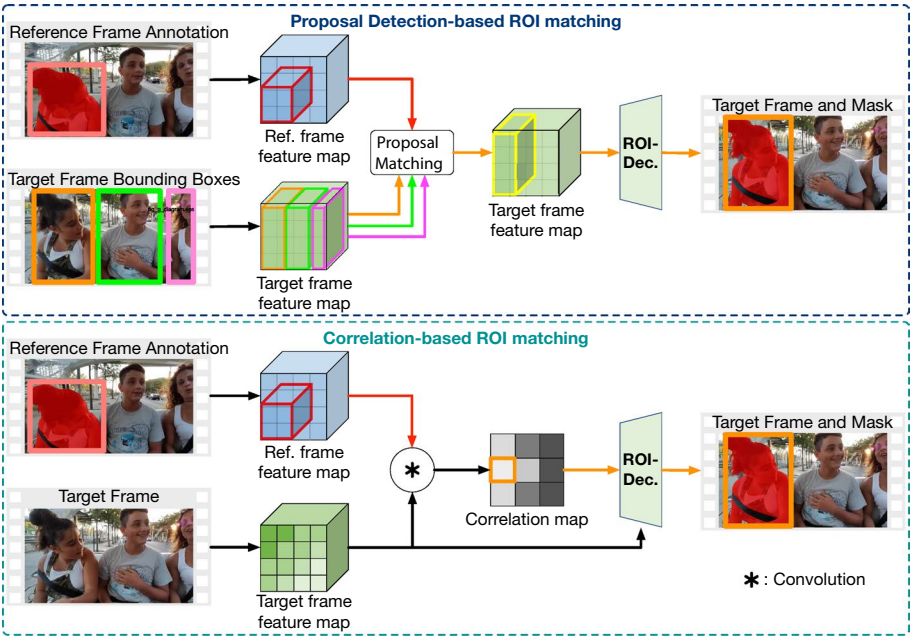

Fig. 12 Diagram of two types of ROI-level matching. **Top** (Proposal detection): Use a proposal detection method to generate object proposals (Yellow, green, and pink boxes) from the target frame, which are then compared with the objects annotated in the reference frame (Red box). The boxes with high similarities would be kept and further decoded to the segmentation results. **Bottom** (Correlation): Use a shared encoder to embed the reference and target frames, respectively. Then, consider the target object features (Red cube) in the reference frame as the kernel to convolve the target frame features (Green cube). From the resulting correlation map, the regions with high correlations (Red box in the correlation map) would be resolved to object proposals and further decoded to the segmentation results. The coloured arrows indicate the data flow only containing the features within the corresponding boxes. Best viewed in colour

Table 10 Summary of the discussed ROI-level matching methods

Methods	Reference	Proposal Types	Measurements
DyeNet	1, PR (confident boxes)	Bounding box	Cosine similarity
PReMVOS	1	Object mask	Normalised L2 similarity
DTN	1	Bounding box	L2 similarity
DMM-Net	1	Object mask	Differentiable matching
FAVOS	1	Bounding box (object part)	Cross correlation
SiamMask	1	Bounding box	Depth-wise cross correlation
SAT	1	Bounding box	Cross correlation

The first item indicates the indices of reference frames. Note that 'PR' (Proposal-based Reference) means the reference consists of proposals instead of frames

DMM-Net (*Differentiable Mask-Matching Network*, Zeng et al. 2019a) implements a more optimal matching scheme than the above methods relying on ROI-level feature similarities. The scheme is achieved by solving a linear programming problem:

$$\begin{aligned} &\min_{X} \ \&\text{Trace}(CX^{\mathrm{T}}) \\ &s.t. \ \&X\mathbf{1}_m = \mathbf{1}_n, \ X^{\mathrm{T}}\mathbf{1}_n \leq \mathbf{1}_m, \ X \geq 0 \end{aligned}, \tag{6}$$

where C, $X \in \mathbb{R}^{n \times m}$ are initial and target affinities between n and m masks in the reference and target frames. C is initialised with the cosine similarities/IoUs between ROI-level features/masks. $\mathbf{1}_n$, $\mathbf{1}_m$ are all-one vectors. **Discussion**: Due to the optimal matching scheme, DMM-Net is robust against ambiguous regions and dramatic object property changes.

DTN (*Dynamic Targeting Network*, Zhang et al. 2019) switches between ROI-based and mask propagation-based SVOS, according to the temporal continuity between frames (measured by optical flow). Unlike other ROI-based methods, DTN generates object proposals from the past frame masks instead of resorting Mask R-CNN. **Discussion**: Therefore, DTN achieves a good balance between SVOS accuracy and efficiency. However, the overall architecture in DTN is less optimal since an extra module (FlowNet) is required to trigger the dynamic mechanism.

(2) Correlation-based ROI matching. FAVOS (*Fast and Accurate VOS*, Cheng et al. 2018) combines SiamFC (Bertinetto et al. 2016) and an ROI-based segmentation module for SVOS. Unlike other ROI-based methods, FAVOS tracks and segments object parts instead of the entire objects. **Discussion**: Besides high efficiency, FAVOS is also robust against occlusion and object deformation. However, there is still room for further accuracy improvement since the segmentation module focuses more on efficiency and fails to utilise detailed features sufficiently.

SiamMask (Wang et al. 2019a) combines SiamRPN (Li et al. 2018a) and a light segmentation module for SVOS and VOT. In order to achieve rich cross-frame correlations, the method implements the correlation module with depth-wise convolution rather than vanilla convolution operations. **Discussion**: Benefiting from its efficient correlation computation and segmentation, SiamMask achieves the competitive balance between SVOS accuracy and efficiency. However, SiamMask suffers a similar problem to FAVOS: the less optimal ROI segmentation.

SAT (*State-Aware Tracker*, Chen et al. 2020) combines SiamFC++ (Xu et al. 2020) and a saliency detection module for SVOS. Unlike FAVOS and SiamMask, which perform object tracking and segmentation in order, the method executes both modules in parallel. Then, the resulting correlation and saliency maps are fused for final outputs. In SAT, saliency maps are computed within the ROIs generated by a dynamic strategy. The strategy switches between the efficient or accurate approaches based on the previous frame outputs. **Discussion**: Unlike DTN (another dynamic method), SAT can switch approaches more efficiently since no extra modules (e.g., optical flow) are involved. In addition, SAT computes a global representation for each target object, improving the global consistency of the segmentation results throughout the sequence.

4.2.3 Summary

This section reviews the feature matching-based VOS method, which resorts to pixel-/ROI-level feature correspondence between frames to segment objects. Based on the discriminative knowledge learned offline, this method can predict robust results without online fine-tuning, resulting in a better balance of VOS efficiency and accuracy. So far, most top-performing SVOS methods have been developed based on feature matching, as shown in

Tables 16, 17, and 19. In addition, the reliable correspondence also brings high-quality results to UVOS methods, as shown in Table 20.

Most existing feature matching-based VOS methods are implemented with **pixel-level matching**, which measures the dense correspondence between frames. The earlier practices (e.g., PLM, (Yoon et al. 2017), RGMP (Oh et al. 2018), and AGSS-VOS (Lin et al. 2019)) achieve this **implicitly** via learnable fully-connected or CNN-based layers. To suppress ambiguous regions, the above methods take the previous frame masks into account to constrain the matching area. Therefore, they only apply to SVOS.

Unlike implicit matching, the **explicit scheme** achieves dense correspondence by directly measuring the pair-wise similarities between target and reference frames' pixels. Therefore, no extra modules are required for the correspondence, and the backbone networks can focus more on the discriminative representation. The earlier methods (PML (Chen et al. 2018b), VideoMatch (Hu et al. 2018c)) only utilise the correspondence to propagate labels between frames. With a CNN-based decoder, the subsequent methods (FEELVOS (Voigtlaender et al. 2019), RANet (Wang et al. 2019d)) combine the correspondence with the fine-grained features in the target frame, further improving the segmentation performance. By considering more past frames as the reference, STM (Oh et al. 2019) starts a new branch on top of pixel-level matching: the **memory-based method**, which significantly enhances the VOS robustness against object changes, occlusion, and fast motion. Most of the current SVOS practices are inspired by STM, and they improve the segmentation performance mainly in the following aspects: (1) temporal correspondence between memory frames (EGMN, (Lu et al. 2020a)); (2) matching scheme (KMN, (Seong et al. 2020), LCM (Hu et al. 2021), RMNet (Xie et al. 2021)); (3) memory management (AFB-URR (Liang et al. 2020), SwiftNet (Wang et al. 2021a)); (4) encoder architecture (SwiftNet (Wang et al. 2021a), STCN (Cheng et al. 2021)); (5) similarity metrics (STCN (Cheng et al. 2021)); (6) multi-scale matching (HMMN (Seong et al. 2021)). Although achieving great improvement in VOS performance, several challenges remain in the memory-based VOS, e.g., a better balance between memory management, VOS accuracy, and VOS efficiency. Besides the memory-based methods, there are still current practices exploring different ways to improve pixel-level matching-based VOS. For example, TVOS (Zhang et al. 2020), CFBI/CFBI+ (Yang et al. (2020, 2021b)), and AOT (Yang et al. (2021a)).

Pixel-level matching also applies to UVOS, whose main idea is to locate the frequently appearing objects via the cross-correlation between frames. For example, AD-Net (Yang et al. 2019b), COSNet (Lu et al. 2019, 2020b), and EGMN (Lu et al. 2020a) perform UVOS based on the cross-correlation. The main difference between them is the selection of the reference frames. From Table 20, it is observed that the matching-based UVOS methods outperform other methods, illustrating the effectiveness of cross-correlation on UVOS.

The discussed variants of pixel-level matching consider super-pixels (CCCN, (Li et al. 2017b)) or capsule components (CapsuleVOS, (Duarte et al. 2019)) as the basic entities for matching. Instead of regular grids, the variants encode features for the regions with similar appearance or semantic information, enriching the diversity of the matching-based VOS. Although their segmentation performance is limited, these variants still make a good exploration for future research.

The **ROI-level matching** scheme measures the ROI-level correspondence between frames. In general, there are two approaches to generate ROIs from the target frame: **Detection-based approach** and **Correlation-based approach**. The earlier detection-based methods (DyeNet (Li and Change Loy 2018), PReMVOS (Luiten et al. 2018), DMM-Net (Zeng et al. 2019a)) utilise off-the-shelf networks (e.g., Mask R-CNN) to generate proposals (bounding boxes or coarse masks) for the target objects. To accelerate the ROI

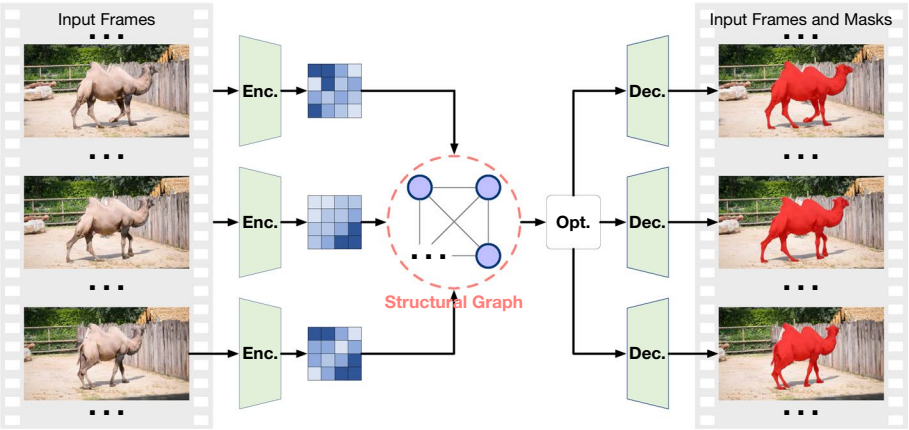

Fig. 13 Diagram of graph optimisation-based VOS methods, which apply to both SVOS and UVOS. Specifically, the methods organise the input features (blue maps) into a structural graph, encoding the comprehensive global context. Then, the context is optimised to generate the final segmentation results. Best viewed in colour

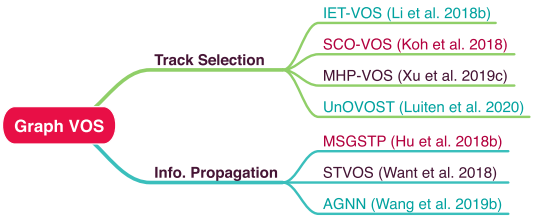

Fig. 14 Development roadmap of the discussed graph optimisation-based methods. Note that the blue, black, and red words indicate the methods performing UVOS, SVOS, and both, respectively. Best viewed in colour

generation and suppress ambiguous backgrounds, DTN (Zhang et al. 2019) predicts object proposals from the previous frame results instead of resorting extra networks. In contrast, the correlation-based methods (e.g., FAVOS (Cheng et al. 2018), SiamMask (Wang et al. 2019a), and SAT (Chen et al. 2020)) generate ROIs based on the correlations between the first frame annotations and the target frame. Their differences mainly lie in the matching entities and the approach for the target object localisation. Since the correlation-based methods do not require any extra networks for ROI generation, they perform VOS much faster than the detection-based ones. However, the segmentation performance of this method is limited because the fine-grained details are ignored in the ROI-based segmentation module.

4.3 Graph optimisation-based methods

Until 2021, most existing VOS methods implement spatial techniques based on online fine-tuning and feature matching. However, for each frame to segment, both techniques only establish the correspondence between the frame and the limited number of other frames in the same video, which is insufficient to handle the objects experiencing dramatic changes. To address the issue, the graph optimisation-based technique was developed, in which additional video frames from the same video are involved and organised as a graph structure. Based on the analysis and optimisation for the graph, this technique can achieve more

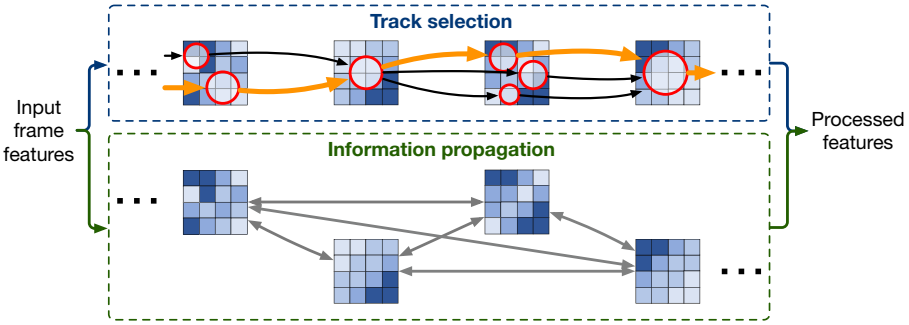

Fig. 15 Diagram of two techniques for graph node organisation and analysis. In the track selection-based methods, super-pixels or object proposals are considered as nodes (red circles), which are connected with those from other frames only. The connected nodes form a set of tracks throughout the sequence. This technique aims to retrieve an optimal track (an example for the optimal track is highlighted in bold orange arrows) from the formed tracks (orange and black arrows). The final results come from the nodes belonging to the optimal track. As for the ones utilising information propagation, more varied connections are supported (even the nodes within the same frame). This technique aims to establish a high-level and comprehensive context by exchanging information between nodes (grey lines with two-way arrows). The final results come from both the spatial features and context information. Best viewed in colour

Table 11 Summary of the discussed graph optimisation-based VOS methods

Methods	Node entities	Connection modes
IET-VOS	Seed pixels	Nodes from I_{t-1} and I_{t+1}
SCO-VOS	Object masks	Nodes from all other frames
MHP-VOS	Bounding boxes	Nodes from I_{t-1} and I_{t+1}
UnOVSVT	Object tracklets	Temporally neighbouring and visually similar nodes
MSGSTP	Super-pixels	Nodes from $[I_{t-15}, I_{t+15}]$
STVOS	Super-pixels	Nodes from I_{t-1} and I_{t+1}
AGNN	Video frames	Nodes from $\{..., I_{t-2N}, I_{t-N}, I_t, I_{t+N}, I_{t+2N}, ...\}$, where $N = T/5$, T is the video length.

The last item indicates which nodes should be connected with for each node in the t^{th} frame I_t

comprehensive and high-level dependencies between frames, further facilitating the high-quality segmentation results, as shown in Fig. 13.

In general, there are two types of approaches to organise and analyse the graph nodes: **track selection** and **information propagation**, as shown in Fig. 14. Given a set of input frames, the former (Sect. 4.3.1) retrieves the optimal node track throughout the video, from which the target object masks can be derived. In contrast, the latter (Sect. 4.3.2) focuses more on propagating the foreground and background information between frames, achieving a global context to facilitate the segmentation. The difference between them is shown in Fig. 15. Section 4.3.3 summarises the discussed methods.

4.3.1 Track selection-based method

This method performs VOS by resolving an optimal node track from a set of continuous frames. Specifically, the method first encodes input frames. Then, as shown in Fig. 15, nodes are abstracted from the encoded features and then connected frame by frame. Finally, the optimal node track is resolved from the graph and decoded to the object masks. This section introduces the representative track selection-based works, which are briefly summarised in Table 11.

IET-VOS (*Instance Embedding Transfer to UVOS*, Li et al. 2018b) is one of the earliest VOS methods based on track selection. With the pixel-level instance embeddings, the method considers the locally stable and globally diverse pixels as the seed pixels, over which a set of tracks are built throughout the video. The track with the highest accumulated objectness and motion scores are selected as the optimal track to derive the object masks. **Discussion**: Due to the accumulation-based strategy, the method can effectively localise the frequently reappearing objects from the input video.

SCO-VOS (*Sequential Clique Optimisation for VOS*, Koh et al. 2018) builds a k-partite graph over object masks (predicted by FCIS,). The method derives the optimal mask track by minimising an energy function defined upon the graph nodes. **Discussion**: Unlike IET-VOS, SCO-VOS considers object-level features, which makes the segmentation more robust against noises, but extra computations and more elaborate optimisation are required.

MHP-VOS (*Multiple Hypotheses Propagation for VOS*, Xu et al. 2019c) builds tree structures to cover all the possible tracks of object proposals throughout the sequence. For each target object, the method derives one track with the highest spatial-temporal consistency. **Discussion**: Compared with other graphs, tree structure applies better to SVOS since it generates fewer hypotheses for earlier frames and more for later frames, consistent with the confidence decay of the propagated masks with time.

UnOVOST (*Unsupervised Offline Video Object Segmentation and Tracking*, Luiten et al. 2020) implements a similar approach to MHP-VOS. Differently, the method considers tracklets instead of object proposals as nodes, where each tracklet is a track of spatiotemporally consistent object proposals. **Discussion**: Compared with MHP-VOS, fewer nodes are involved in UnOVOST, mitigating the efforts for track selection. In addition, the method supports multi-object UVOS due to the integration of instance-level segmentation module.

4.3.2 Information propagation-based methods

Information propagation is another technique to implement graph optimisation-based VOS. This technique builds the comprehensive context of target objects and backgrounds by propagating semantic information between nodes. The resulting context provides more discriminative features are achieved for robust segmentation. Table 11 briefly summarises the discussed methods.

MSGSTP (*Motion Saliency-Guided Spatio-Temporal Propagation*, Hu et al. 2018b) builds intra-frame, inter-frame, and long-range connections between super-pixels. Along with the connections, the method propagates the initial motion saliency to achieve the output object masks. **Discussion**: Although MSGSTP is not a deep learning (DL)-based method, the idea of graph optimisation-based information propagation is enlightening and worth mentioning. In addition, the method can achieve competitive results with other DL-based ones, as shown in Table 20, which provides further evidence of the effectiveness of the proposed graph optimisation-based idea.

STVOS (*Super-Trajectories for VOS*, Wang et al. 2018) performs SVOS by propagating the annotated labels along super-trajectories, which link sets of pixels from continuous frames. Unlike MSGSTP, which only determines super-pixels based on spatial features, STVOS also considers motion clues when grouping pixels, enabling reliable temporal consistency. **Discussion**: STVOS is also a non-DL method (MSGSTP and STVOS are the only two non-DL methods in this review). However, the segmentation results of STVOS demonstrate that point trajectory, as an earlier VOS technique, still works well if designed elaborately. With the development of deep feature descriptors, it is believed that the trajectory-based method could be further improved.

AGNN (*Attentive Graph Neural Network*, Wang et al. 2019b) builds a fully-connected graph upon a subset of the input sequence. During inference, the method iteratively performs information propagation, achieving a global context for all the involved nodes. **Discussion**: Unlike the above methods, AGNN considers video frames as nodes, where more comprehensive features are encoded to build high-level dependencies. However, the efficiency of information propagation and summarisation decreases as the number of involved nodes increases, limiting further performance improvement.

4.3.3 Summary

In this section, several representative graph optimisation-based methods are introduced. Unlike online fine-tuning and feature matching, this technique considers more frames to establish extensive correspondence throughout the sequence. Therefore, the VOS methods based on graph optimisation can better localise frequently appearing objects from raw videos, achieving competitive results in UVOS (as shown in Table 20).

There are mainly two schemes to organise and analyse graph data in VOS methods: (1) track selection and (2) information propagation. The **track selection-based** scheme organises nodes into a set of tracks or trees, from which the optimal track is retrieved to generate object masks. The earlier method (IET-VOS, (Li et al. 2018b)) only embeds pixel-level features to build tracks, which are vulnerable to background noises. Therefore, the subsequent ones (SCO-VOS (Koh et al. 2018), MHP-VOS (Xu et al. 2019c), and UnOVOST (Luiten et al. 2020)) consider object proposals/tracklets for robust track generation and selection. However, these methods require extra networks (e.g., Mask R-CNN or FCIS) to generate object proposals, increasing the network parameters and computation costs.

Unlike track selection, the **information propagation-based** scheme assists VOS via iterative information propagation among nodes. The earlier methods (MSGSTP (Hu et al. 2018b), STVOS (Wang et al. 2018)) mainly propagate the label-related information (e.g., motion saliency or annotations), making the segmentation results sensitive to the connections between nodes. The recent work (AGNN, (Wang et al. 2019b)) mitigates this issue due to the consideration of deep features and frame-level information propagation. To sum up, graph optimisation brings more extensive correspondence to VOS methods, enabling high-quality segmentation results (especially in UVOS). However, such good results are achieved at the cost of segmentation efficiency due to graph optimisation's high computation complexity (whether the track selection- or information propagation-based schemes). As a result, the current graph optimisation-based approaches are unsuitable for real-time and resource-limited VOS tasks.

Table 12 Summary of the discussed optical flow-based VOS methods. (t-1, t) measures the flow maps from I_{t-1} to I_t. "Computation and Usage" indicate how the listed methods compute and use the flow maps

Methods	Flow maps	Computation and usage
MP-Net	(t-1, t)	LDOF; Generate initial object masks
FusionSeg	(t-1, t)	CNN module; Fused with appearance (when predicting)
SegFlow	(t-1, t)	CNN module; Fused with appearance (when upsampling)
CINM	(t-2, t), (t-1, t), (t, t+1), (t, t+2)	FlowNet 2.0; Build temporal dependencies among pixels
MBN-VOS	(t-1, t)	FlowNet 2.0; Background suppression
MoNet	(t-1, t); (t, t+1)	FlowNet 2.0; Feature wrapping and background suppression

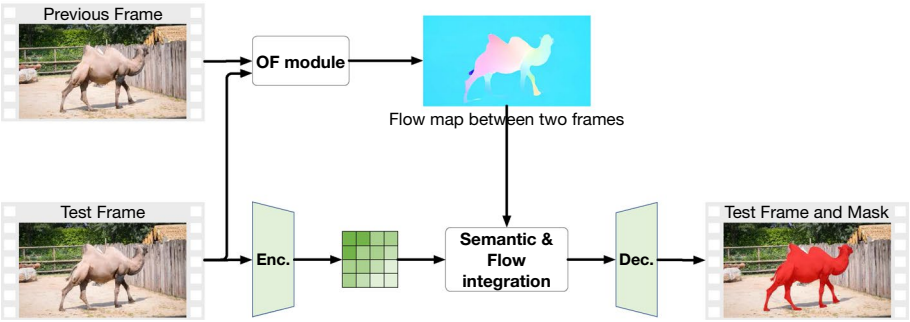

Fig. 16 Diagram of the optical flow-based VOS methods, which apply to both SVOS and UVOS. During inference, the methods compute the optical flow from continuous frames and integrate the flow map with semantic features for object mask generation. When the target object and background have separate motion patterns, their difference can be easily derived, providing helpful clues for segmentation. The map shown here is generated by FlowNet 2.0 (Ilg et al. 2017), the most frequently used approach for flow computation in VOS

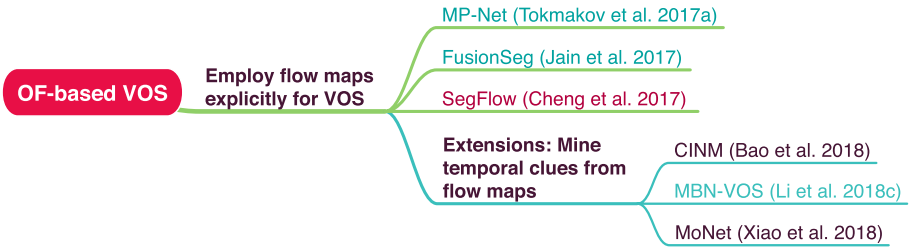

Fig. 17 Development roadmap of the discussed optical flow-based VOS methods. With the estimated flow maps, the earlier methods explicitly feed them into segmentation networks, while recent methods further mine temporal clues from them. Note that the blue, black, and red words indicate the methods for UVOS, SVOS, and both, respectively. Best viewed in colour

4.4 Optical flow-based methods

The previous Sects. 4.1, 4.2, and 4.3 introduce the frequently used spatial techniques for VOS methods. These techniques. These techniques are beneficial to address several

challenges in VOS, such as occlusion, out of view, and fast motion. However, only considering spatial features cannot generate high-quality results when segmenting the objects with appearance change, scale variation, or complex background. Therefore, several techniques focusing on continuous video frames (i.e., optical flow, mask propagation, and long-term temporal modelling) have been proposed to address these challenges.

Optical flow has been a widely used technique in VOS due to the pixel-level motion patterns. The technique assumes that the target object and backgrounds have different movement patterns. Therefore, integrating optical flow into VOS can provide segmentation networks with reasonable priors, as shown in Fig. 16. The earlier methods integrate the estimated flow maps explicitly into their segmentation networks. To further encode the temporal clues implicit in flow maps, some recent methods employ optical flow to build the short-term correspondence between frames. The representative works are depicted in Fig. 17. In this section, the earlier works are discussed first in Sect. 4.4.1, followed by the recent extensions (Sect. 4.4.2). Section 4.4.3 summarises the discussed methods. The properties of these methods are briefly listed in Table 12.

4.4.1 Early optical flow-based methods

MP-Net (*Motion Pattern Network*, Tokmakov et al. 2017a) is one of the earliest deep methods using optical flow. LDOF (Large Displacement Optical Flow, Brox and Malik 2010a) is used to generate flow maps between frames, which are refined by deep networks and then integrated with objectness scores for the final results. **Discussion**: Due to the comprehensive motion patterns and complementary information integration, MP-Net can better segment the moving object from complex scenes than traditional motion-based VOS methods.

FusionSeg (Jain et al. 2017) implements a parallel mechanism to integrate motion patterns and semantic information. Unlike MP-Net, which post-process the motion-based results with objectness, FusionSeg encodes the motion-based and appearance-based segmentation results with two separate branches and combines them later. **Discussion**: Therefore, the combination module can be trained to better utilise both information and generate more accurate results.

SegFlow (Cheng et al. 2017) is also a parallel network integrating motion and appearance features. Unlike FusionSeg, SegFlow trains two branches for different goals (for segmentation and optical flow, respectively). Therefore, the method cannot combine both branches directly. Instead, the combination is achieved during the upsampling stage. **Discussion**: Compared with MP-Net and FusionSeg, SegFlow builds a more tight bridge between motion and appearance features, resulting in an end-to-end trainable architecture for both UVOS and SVOS.

4.4.2 Extensions

Optical flow brings pixel-level location and shape priors to the target object, enabling VOS methods to produce competitive results on several benchmark datasets. However, limited by the lack of training data and the challenges such as moving background and camera shaking, the quality of the estimated flow maps are not always good enough to provide such valuable priors. Therefore, several methods have been proposed to mine more confident temporal information from the flow maps.

CINM (*Cnn IN Mrf*, Bao et al. 2018) performs SVOS by minimising an MRF model:

$$E(\mathbf{x}) = \sum_{i \in \mathcal{S}} E_u(x_i) + \sum_{(i,j) \in \mathcal{N}_T} E_t(x_i, x_j) + \sum_{c \in \mathcal{S}} E_s(\mathbf{x}_c), \tag{7}$$

where E_u, E_t, and E_s are the energies for likelihood maximisation, temporal and spatial dependencies. x denotes the initial labels predicted by OSVOS, $\mathcal{V}$ defines all the pixels in a video sequence. The local connection $\mathcal{N}_T$ is established by optical flow. $\mathbf{x}_c$ is an auxiliary mask in the c^{th} frame, initialised with x and then refined by a DeepLabv2-based module. **Discussion**: Unlike the above methods, CINM considers flow maps as a constraint term in MRF instead of using them explicitly for mask generation. Although achieving competitive results, the computation cost is enormous in CINM since it integrates several deep networks (OSVOS, FlowNet, and DeepLab).

MBN-VOS (Motion-based Bilateral Network for UVOS, Li et al. 2018c) implements a Bilateral Network (Jampani et al. 2016), which generates generate background motion patterns based on objectness and optical flow. The resulting patterns can serve as priors to reduce the negative effect of static background objects on the final results. **Discussion**: Unlike CINM, optical flow in MBN-VOS is mainly used for background suppression, further improving the segmentation performance, especially when background objects look similar to the foreground objects.

MoNet (Xiao et al. 2018) implements a similar approach to MBN-VOS for background suppression. However differently, MoNet generates background motion patterns from the flow branch only. In addition, both forward and backward optical flows are estimated to wrap the adjacent frame features. **Discussion**: With the bi-directional flow maps and the wrapped features, MoNet can effectively encode the spatial-temporal correspondence between frames, improving VOS performance.

4.4.3 Summary

This section discusses the representative VOS methods based on optical flow. These methods assume that the target object and backgrounds have different motion patterns. Therefore, the generated flow maps can reasonably estimate the shape and location priors of the target object. The earlier methods (MP-Net (Tokmakov et al. 2017a), FusionSeg (Jain et al. 2017), and SegFlow (Cheng et al. 2017)) explicitly integrate the flow maps with spatial features, to generate object masks. The main difference between them is how to encode and combine different types of features. However, the estimated flows are not always reliable due to the lack of training data and challenging sequences (e.g., dynamic background). Therefore, several methods have been recently proposed to exploit optical flow more sufficiently while avoiding the above risks. As discussed above, CINM (Bao et al. 2018) implicitly constraints the VOS results with optical flow-based temporal dependencies. MBN-VOS (Li et al. 2018c) and MoNet (Xiao et al. 2018) suppress the background regions by recognising the background motion patterns.

Although achieving high-quality results on many challenging sequences, optical flow has been rarely employed in recent VOS systems since: (1) In some cases, the object and background flow maps are not discriminative (e.g., static object); (2) Extra deep networks (e.g., FlowNet) are required during inference (as shown in Tables 3, 4 and 5). Therefore, this technique has been gradually replaced by mask propagation, which will be discussed in the next section.

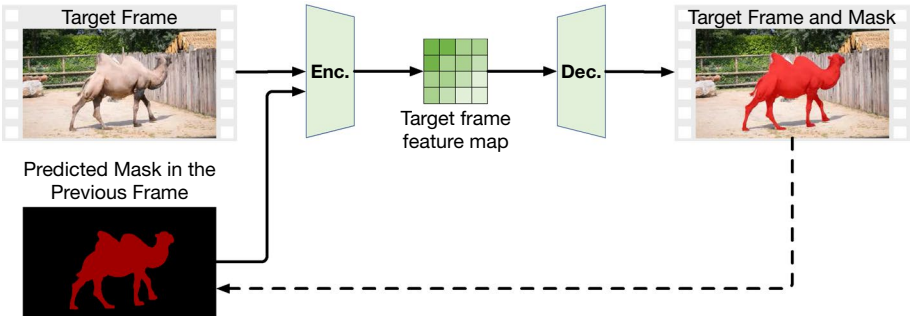

Fig. 18 Diagram of the mask propagation-based VOS method, which applies to SVOS only. During inference, the segmentation network takes the target frame and the previous frame mask as input and predicts object masks for the target frame. After this, the predicted masks are propagated to serve the subsequent frame segmentation (the dashed line with an arrow)

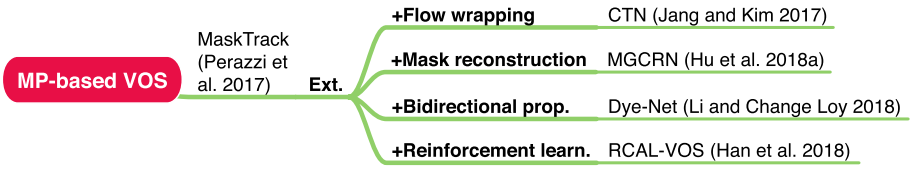

Fig. 19 Development roadmap of the mask propagation-based VOS. MaskTrack is the first method using mask propagation in SVOS. After that, several methods are proposed on top of MaskTrack for further improvement. Their main modifications to MaskTrack are highlighted by bold words with the prefix "+"

Table 13 Summary of the discussed methods based on mask propagation

Methods	Directions	Mask refinement	Focused regions
MaskTrack	Forward	–	Whole target frame
CTN	Forward	OF-based wrapping	Bounding box of the refined mask
MGCRN	Forward	OF-based contour evolution	Whole target frame
DyeNet	Bi-direction	OF-based wrapping	Bounding box of the refined mask
RCAL-VOS	Forward	RL-based relocation	Bounding boxes containing object and surrounding context, generated from the propagated mask

The first item indicates the directions of mask propagation. The second item shows the techniques used for mask refinement (OF: Optical Flow, RL: Reinforcement Learning). The final item indicates the target frame regions focused explicitly by segmentation networks. Note that "Whole target frame" means the propagated masks implicitly guide the segmentation

4.5 Mask propagation-based methods

This section discusses the representative methods based on mask propagation. This method assumes the target objects move smoothly throughout the input sequence. Therefore, the masks predicted in the previous frame can well estimate the location and shape of the target objects. With such estimates, the methods can focus more on the regions where the target objects most probably appear. Unlike optical flow, mask propagation applies better to

the sequences with dynamic backgrounds. Tables 3, 4, and 5 show that most existing methods are built based on this technique.

Fig. 18 shows the diagram of the mask propagation-based VOS method. In the MaskTrack proposed by Perazzi et al. (2017), mask propagation is first proposed for deep learning-based SVOS (Sect. 4.5.1), where the masks predicted in the previous frame are directly fed to the segmentation network for mask generation. To further improve the segmentation performance, recent methods incorporate other valuable techniques to enhance the confidence of the propagated masks, e.g., optical flow, bidirectional propagation, and reinforcement learning, enabling the segmentation network to focus on more plausible regions. These methods are introduced in Sect. 4.5.2. Section 4.5.3 summarises the discussed methods. Table 13 briefly summarises the related methods. The development roadmap of the discussed methods is shown in Fig. 19.

4.5.1 MaskTrack

MaskTrack, proposed by Perazzi et al. (2017), is the first SVOS method based on mask propagation. As shown in Fig. 18, the segmentation network takes a tensor with four channels (the current frame + previous frame mask) as input and predicts object masks for each target frame. **Discussion**: Although achieving competitive results, MaskTrack cannot handle the objects experiencing abrupt changes or occlusion since the previous frame masks fail to provide the correct estimations. Therefore, the majority of its extension works focus on how to improve the confidence of such estimations.

4.5.2 Extensions

CTN (*Convolutional Trident Network*, Jang and Kim 2017) adapts the previous mask to the current frame. For each pixel $\mathbf{p} = [x, y]^{\mathrm{T}}$, its label $H^t(\mathbf{p})$ comes from the previous frame labels and flow map between frames:

$$H^t(\mathbf{p}) = S^{t-1}(x + u_b^t(\mathbf{p}), y + v_b^t(\mathbf{p})), \tag{8}$$

where S^{t-1} is the previous frame mask, $[u_b^t(\mathbf{p}), v_b^t(\mathbf{p})]$ is the flow vector at pixel $\mathbf{p}$. **Discussion**: Unlike MaskTrack, CTN refines the foreground and background masks separately. With the mask adaptation by optical flow, future changes can be better handled.

MGCRN (*Motion-Guided Cascaded Refine Network*, Hu et al. 2018a) also refines the previous mask with optical flow. Unlike CTN, MGCRN utilises active contours (Chan and Vese 2001) to iteratively estimate the object contours from the flow map, where the initial contours come from the previous mask. **Discussion**: Due to properties of active contours, the estimation can focus mainly on the regions neighbouring to the target object from the previous frame, suppressing the distraction from background objects.

DyeNet (Li and Change Loy 2018) has been discussed in Sect. 4.2.2 due to its ROI-based matching. Besides, DyeNet implements bi-directional mask propagation. To this end, DyeNet first predicts a set of high-confident masks from the input sequence and then propagates them bi-directionally to the remaining frames. **Discussion**: Unlike other methods, DyeNet performs better on the sequences with dramatic occlusions or deformations since, in some cases, the mask propagated inversely might better estimate the shape and location of the target object.

RCAL-VOS (*Reinforcement Cutting-Agent Learning for VOS*, Han et al. 2018) implements two deep reinforcement learning-based networks (CPN: Cutting-Policy Network;

CEN: Cutting-Execution Network), which are trained to predict a set of actions to refine the propagated object proposals (e.g., moving up, down, left or right; shrink or expand; ratio changes; stop). **Discussion**: Unlike other methods, optical flow is not required for mask refinement. Also, the method demonstrates that the contrast information around target object is beneficial for high-quality results.

4.5.3 Summary

This section discusses several representative methods based on mask propagation. This method originates from MaskTrack (Perazzi et al. 2017). With the underlying shape and location priors, the segmentation network can be guided to focus more on the object region and derive high-quality results. However, there are still some factors affecting the VOS performance: (1) drastic deformation; (2) abrupt motion or occlusions. The previous frame mask generally fails to estimate the potential shape or location of the target object when these factors occur. Therefore several extension works are proposed to improve the confidence of the propagated mask.

CTN (Hu et al. 2018a) and MGCRN (Hu et al. 2018a) improve the propagated mask with optical flow, providing the motion clues to adapt to the changes between frames. Although MaskTrack also considers optical flow, more straightforward approaches are utilised in the extensions to refine the propagated masks. However, these methods still cannot handle occlusions and abrupt motion well. To address the issue, the bi-directional mask propagation (DyeNet, Li and Change Loy 2018) is proposed. Some occluded-then-reappear objects can contribute to the VOS via inverse mask propagation. RCAL-VOS (Han et al. 2018) implements a deep reinforcement learning-based method to generate potential target regions. Without the deep networks for optical flow, RCAL-VOS learns a policy to automatically adapt the propagated mask to the target frame.

From Tables 3, 4 and 5, it is observed that most existing methods consider mask propagation for VOS. Since their implementations about the propagation are similar, only five representative methods are discussed here. Mask propagation contributes a lot to the earlier methods because it can provide implicit location and shape priors. In the recent SVOS methods based on feature matching, the propagated mask has been the essential data indicating the probabilities that a reference point belongs to the target object or background.

4.6 Long-term temporal propagation-based methods

The previous two sections discuss the VOS methods based on optical flow and mask propagation, which only focus on the short-term temporal correspondence and generate implicit object location and shape priors for target objects. Relying less on visual features enables these methods to handle the sequences with visually ambiguous backgrounds. However, only short-term temporal clues are insufficient for high-quality results when segmenting the sequences with dynamic backgrounds or heavy occlusions. To address the issues, methods based on long-term temporal information are proposed. These methods accumulate the spatial-temporal clues from relative longer video clips, implicitly encoding the dynamic properties related to the target objects and background, such as the changes in appearance, scale, location and shape. Therefore, segmentation networks can adapt to the changes and achieve better results.

Fig. 20 shows the diagram of this method. According to the techniques for spatial-temporal information extraction and utilisation, the discussed methods can be mainly

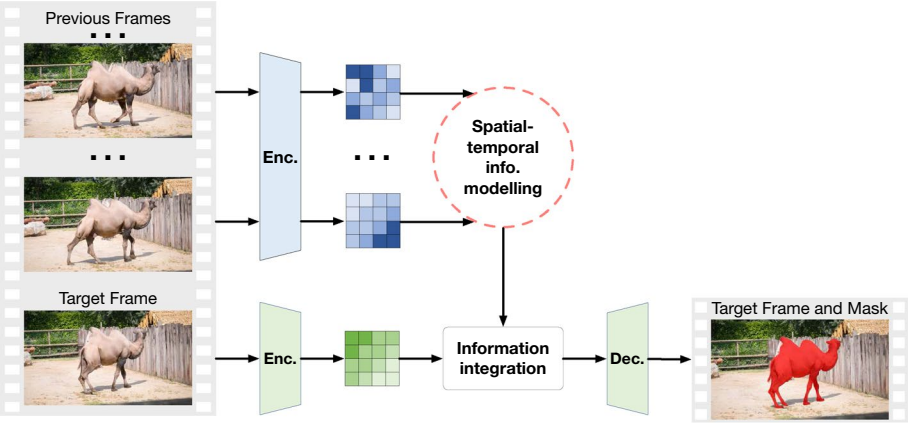

Fig. 20 Diagram of VOS methods based on long-term temporal propagation, which apply to both SVOS and UVOS. For each target frame, the encoded features and the features propagated from a set of past frames are integrated for mask generation. Note that in some cases, the encoders for the target and past frames are different, and thus they are highlighted in different colours. Best viewed in colour

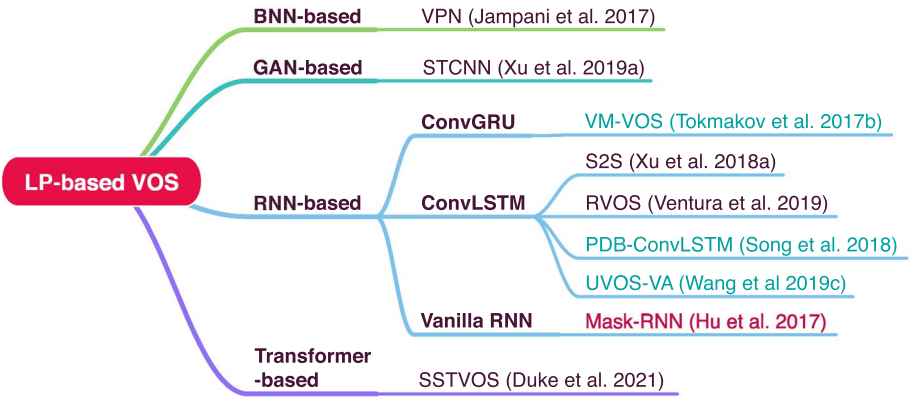

Fig. 21 Development tracks of VOS methods based on long-term temporal propagation (abbreviated as 'LP-based VOS'). Based on the techniques for spatial-temporal information extraction, the listed methods are grouped into three categories: BNN-based method, GAN-based method and RNN-based method. Note that the methods marked in blue and black are UVOS and SVOS methods, respectively, those in red are the methods for both UVOS and SVOS. Best viewed in colour

categorised into three groups: (1) Bilateral Neural Network (BNN)-based methods (Sect. 4.6.1); (2) Generative Adversarial Network (GAN)-based methods (Sect. 4.6.2); (3) Recurrent Neural Network (RNN)-based methods (Sect. 4.6.3); and (4) Transforms-based methods (Sect. 4.6.4). Section 4.6.5 summarises the discussed methods, whose development roadmap is shown in Fig. 21. Table 14 illustrates a brief comparison between these methods.

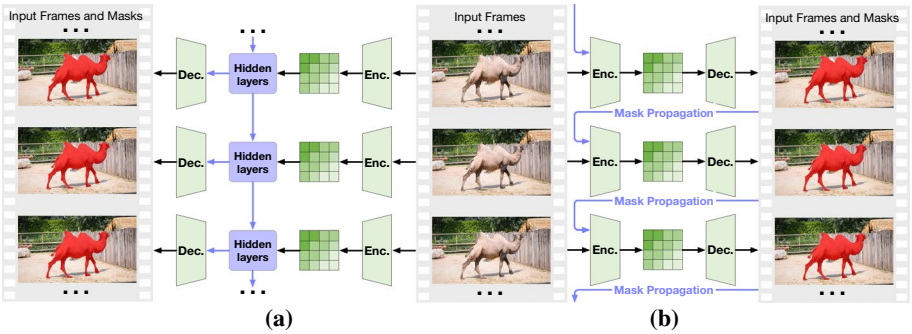

Fig. 22 Difference between the VOS methods based on (a) typical RNNs and (b) recurrently connected networks. In former methods, the hidden layers integrate current frame features and spatial-temporal information propagated from the past frames. The resulting hidden state is then decoded to predict the final results. The latter methods bridge the current and past segmentation via mask propagation. Each of them involves one copy of the segmentation network. The losses are calculated and accumulated during training to update the network parameters. Unlike the methods discussed in Sect. 4.5, which do not accumulate losses across frames, the illustrated methods can better handle error propagation

4.6.1 BNN-based method

VPN (Video Propagation Networks, Jampani et al. 2017) is one of the earliest deep methods based on long-term temporal propagation. The propagation is achieved by a BNN (Bilateral Neural Network), where input frames and masks are converted into bilateral space and then filtered for spatial-temporal propagation throughout the input space. **Discussion**: Due to the dense connections in BNN, VPN can achieve more comprehensive information propagation than traditional VOS methods. However, BNN only takes raw frames as input, limiting higher-level information propagation and further performance improvement.

4.6.2 GAN-based method

Besides BNN, GAN (Generative Adversarial Network, Goodfellow et al. 2014) can also be used for VOS. STCNN (SpatioTemporal CNN, Xu et al. (2019a)) implements a frame generation module, which predicts the target frame according to four previous frames. The module is trained in an adversarial manner to implicitly learn spatial-temporal information propagation and accumulation. **Discussion**: The competitive results demonstrate the effectiveness of the GAN-based technique on VOS. Future research in the longer-term analysis is encouraged since STCNN only considers four frames for accumulation.

4.6.3 RNN-based methods

Many related works have been proposed to solve VOS problems that take advantage of recurrent neural networks (RNNs) in spatial-temporal information modelling. Until now, there have been three types of RNN-based VOS methods: ConvGRU (Ballas et al. 2016), ConvLSTM (Shi et al. 2015b), and the recurrently connected networks. The first two RNNs are the convolutional versions of GRU and LSTM, while the last one is built by recurrently concatenating segmentation networks and trained with BPTT (Back Propagation Through

Time, Werbos 1990). Fig. 22 demonstrates the difference between the VOS methods based on typical RNNs and recurrently connected networks.

(1) ConvGRU-based method. VM-VOS (*Visual Memory for UVOS*, Tokmakov et al. 2017b) implements a bidirectional ConvGRU module. Unlike the aforementioned methods, VM-VOS accumulates and propagates the appearance and motion information (optical flow) in both forward and backward directions. **Discussion**: Since both spatial and temporal features are considered, VM-VOS can encode more comprehensive contexts to handle the sequences with dramatic deformation and occlusions.

(2) ConvLSTM-based methods. S2S (*Sequence-to-Sequence*, Xu et al. 2018a) performs SVOS with a simple but effective ConvLSTM model, whose architecture is similar to the one shown in Fig. 22 (a). Given the first frame annotation, S2S encode features for the annotated objects and propagate them throughout the input sequence. **Discussion**: S2S is a baseline method trained on YouTube-VOS (Xu et al. 2018b). The competitive results show the large-scale and long-range video datasets can activate the VOS methods based on long-term spatial-temporal modelling.

RVOS (*Recurrent network for VOS*, Ventura et al. 2019) extends S2S by propagating information in both spatial and temporal domains. Unlike S2S, RVOS considers each object as an entity, which accumulates the spatial-temporal features from the inter-frame and intra-frame objects. **Discussion**: With the object-level feature propagation, RVOS can encode more comprehensive spatial-temporal dependencies, therefore handling the sequences with complex scenes and multiple target objects.

PDB (*Pyramid Dilated Bidirectional ConvLSTM*, Song et al. 2018) implements a ConvLSTM with PDC (Pyramid Dilated Convolution) and bidirectional propagation. Unlike other methods, the method applies PDC in both backbone and ConvLSTM modules for comprehensive spatial features, which are propagated in directions with different dilated rates. **Discussion**: Due to the multi-scale and bidirectional feature propagation, PDB encodes deeper spatial-temporal relationships between frames, deriving competitive results on challenging sequences.

AGS (*Attention Guided-Segmentation*, Wang et al. 2019c) performs UVOS with a coarse-to-fine mechanism, where the coarse attention maps are predicted from a ConvLSTM-based module and then refined to generate the final results. **Discussion**: Unlike the above methods, AGS demonstrates that even the datasets with coarse annotations (generated with human eye-tracking records) enable the UVOS systems to achieve competitive results. Such a conclusion is further validated in the extended work (Wang et al. 2020) via systematic analysis. Since less cost is required for coarse annotations, more sequences can be considered in future works for further improvement.

(3) Recurrently connected network. MaskRNN (Hu et al. 2017) implements the RNN architecture by connecting the copies of the segmentation network recurrently. As shown in Fig. 22 (b), the connection is established via mask propagation. **Discussion**: The main difference between MaskRNN and mask propagation-based methods is that MaskRNN can accumulate losses across frames during training (via back propagation through time, Werbos 1990). Therefore, the network can handle the sequential inference better with less error propagation.

4.6.4 Transformer-based method

Transformers have been widely explored in recent computer vision tasks (Carion et al. 2020; Wang et al. 2021c). Since transformers can model long-range dependencies, they

Table 14 Summary of the discussed methods based on long-term temporal propagation

Methods	# Frames	Directions	Spatial-temporal information
VPN	9 / t-1	Forward	Bilateral filtering response
STCNN	4 / 4	Forward	Intermediate outputs of the frame generation branch
VM-VOS	NG / T-1	Bi-direction	Concatenation of forward and backward hidden states
S2S	< 10 / T-1	Forward	Forward hidden states
RVOS	4 / t-1	Forward	Forward hidden states (integrating cross-frame and in-frame sequential clues)
PDB	4 / T-1	Bi-direction	Sum of forward and backward hidden states
AGS	< 2 / t-1	Forward	Forward hidden states
MaskRNN	7 / t-1	Forward	Binary object masks
SSTVOS	NG / 3	Non-direction	Spatial-temporal attention maps

"# Frames": Number of the involved frames during training / inference. T: video length. t: target frame index. NG: not given. "Directions": Directions of temporal propagation. "Spatial-temporal info.": Data representing the long-term spatial-temporal information

are naturally suitable for video-related applications. *SSTVOS* (*Sparse Spatiotemporal Transformers for VOS*, Duke et al. 2021) is one of the earliest SVOS approaches based on transformers. The method takes the frame to segment and several past frames as inputs and encodes the spatial-temporal attention and affinities over them for mask generation. To improve the segmentation efficiency, SSTVOS implements a sparse scheme to compute attention maps. **Discussion**: Unlike other methods in this section, SSTVOS processes input frames in parallel instead of sequential, mitigating drifting errors to some extent. However, limited by computation resources, SSTVOS only considers three past frames during inference, failing to sufficiently utilise long-range dependencies over frames.

4.6.5 Summary

This section discusses the VOS methods based on long-term temporal propagation, which accumulate the spatial-temporal features within a period, implicitly encoding the trends of objects and context. There are mainly four types of techniques to achieve this. **BNN (Bilateral Neural Network)** is one of the earliest techniques. VPN, (Jampani et al. 2017) utilises BNN to accumulate spatial-temporal features from past frames. Besides BNN, **GAN (Generative Adversarial Network)-based method** (STCNN, Xu et al. 2019a) GAN also performs well on several VOS benchmarks. The long-term information is embedded implicitly from a frame generation branch. The success of ConvGRU and ConvLSTM in spatial-temporal feature embedding gave rise to the **RNN-based VOS methods**. The earlier methods (VM-VOS, (Tokmakov et al. 2017b), S2S (Xu et al. 2018a)) build standard pipelines to perform VOS with long-term information propagation. PDB (Song et al. 2018) achieves further improvement by incorporating multi-scale feature embedding and bidirectional propagation. To facilitate multiple object segmentation, RVOS (Ventura et al. 2019) propagates both temporal correspondences between frames and spatial relationships between objects. AGS (Wang et al. 2020) implements a coarse-to-fine strategy and validates that even the datasets with coarse annotations can also facilitate learning long-term temporal correspondence. All of the above approaches require a specific module for long-term propagation, which typically runs slowly due to the time-consuming information accumulation.

Table 15 Segmentation accuracy and efficiency of the representative SVOS methods on the DAVIS-2016 validation set

Methods	S. techs			T. techs			Frames	Resolutions	$\mathcal{J}\&\mathcal{F}\uparrow$	FPS ↑
	O	M	G	O	P	L				
OSVOS	✓						1	480 × 854	80.2	0.38
MSKTrack	✓			✓	✓		1, t-1	480 × 854	77.6	0.29
OSMN					✓		1, t-1	480 × 854	73.3	1.87
RGMP		✓		✓		✓	1, t-1	480 × 854	81.8	12.4
SiamMask		✓			✓		1, t-1	480 × 854	69.8	79.6
A-GAME					✓		1, t-1	480 × 854	81.9	12.6
RVOS						✓	[1, t-1, 1]	240 × 427	72.3	193.7
RANet		✓			✓		1, t-1	480 × 854	87.1	41.3
STM		✓			✓		[1, t-1, 5]	480 × 854	89.4	11.9

S. techs: Spatial techniques, O: Online fine-tuning, M: Matching, G: Graph; T. techs: Temporal techniques, O: Optical flow; P: Mask propagation; L: Long-term temporal propagation. Frames: indices of the involved frames, where [b, e, i] indicated the frames sampled from I_b to I_e, with an interval i. FPS: frames segmented per second. Resolutions: resolutions of input video frames

Therefore, several methods (MaskRNN (Hu et al. 2017)) propagate the predicted masks only to the subsequent frames.

From Tables 16, 17, 19, and 20, it is observed that long-term propagation is rarely used in recent methods and it does not bring better results than other methods, even on the long-term VOS benchmark (YouTube-VOS, Table 19). Therefore, the long-term propagation does not contribute a lot to the VOS as the expectation, which might be explained by Table 14. The table shows that the discussed methods accumulate spatial-temporal information from all previous frames during inference. However, only a limited number of frames are used during training due to the limitation of the computation cost. Therefore, the existing methods are still largely trained by short-term frames, which prevents them from achieving desirable results in some cases. Although the transformer-based method (Duke et al. 2021) significantly improves the VOS performance, it still relies heavily on short-term temporal dependencies over frames. Future research is encouraged to learn long-term information propagation with limited resources.

5 Experimental results and discussion

The previous section discussed the current VOS methods according to their techniques exploiting spatial and temporal features. To better understand how these techniques perform and make a fair comparison, we test some representative VOS methods with the same experimental settings. Both quantitative and qualitative results, alongside the theoretical analysis, help to draw the conclusion on these methods. In addition, the segmentation scores of all the reviewed methods on several benchmark datasets are tabulated to further support the conclusion. Last, we outline possible research trends in this field.

Table 16 Segmentation performance of the reviewed SVOS methods on the DAVIS dataset (including 2016 val set, 2017 val set and 2017 test-dev set) (part 1/2)

Methods	S. techs			T. techs			2016 val set			2017 val set			2017 test-dev set			Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
OSVOS (Caelles et al. 2017)	✓						80.2	79.8	80.6	60.3	56.6	63.9	50.9	47.0	54.8	*D16*
MaskTrack (Perazzi et al. 2017)	✓			✓	✓		77.6	79.7	75.4	–	–	–	–	–	–	**E, M, So**, *D16*
VPN (Jampani et al. 2017)						✓	73.7	75.0	72.4	–	–	–	–	–	–	*D16*
CTN (Jang and Kim 2017)				✓	✓		73.5	75.5	71.4	–	–	–	–	–	–	**P**
OnAVOS (Voigtlaender and Leibe 2017)	✓				✓		85.0	85.7	84.2	65.4	61.6	69.1	52.8	49.9	55.7	**P**, *D16*
PLM (Yoon et al. 2017)	✓	✓			✓		66.0	70.0	62.0	–	–	–	–	–	–	*D16*
SegFlow (Cheng et al. 2017)	✓			✓			76.1	76.1	76.0	–	–	–	–	–	–	*D16*
MaskRNN (Hu et al. 2017)	✓			✓	✓	✓	81.4	80.4	82.3	–	60.5	–	–	–	–	*D16, D17*
OSVOS-S (Maninis et al. 2018)	✓						86.5	85.6	87.5	68.0	64.7	71.3	57.5	52.9	62.1	*D16*
CINM (Bao et al. 2018)	✓			✓			–	84.2	–	70.6	67.2	74.0	67.5	64.5	70.5	**P**, *D17*
PML (Chen et al. 2018b)		✓					78.4	77.4	79.3	–	–	–	–	–	–	*D16*
FAVOS (Cheng et al. 2018)		✓					81.0	82.4	79.5	58.2	54.6	61.8	43.6	42.9	44.2	*D16*
RCAL-VOS (Han et al. 2018)	✓			✓			83.8	83.9	83.6	–	–	–	–	–	–	**E, M, P, So**, *D16*
MGCRN (Hu et al. 2018a)	✓			✓			85.1	84.4	85.7	–	–	–	–	–	–	**P**, *D16*
RGMP (Oh et al. 2018)		✓		✓		✓	81.8	81.5	82.0	66.7	64.8	68.6	52.8	51.3	54.4	**E, M, P**, *D17*
MoNet (Xiao et al. 2018)	✓			✓			84.8	84.7	84.8	–	–	–	–	–	–	**P**, *D16*
OSMN (Yang et al. 2018)	✓			✓			–	84.2	–	54.8	52.5	57.1	41.3	37.7	44.9	**C**, *D17*
LSE-VOS (Ci et al. 2018)	✓				✓		81.5	82.9	80.1	–	–	–	–	–	–	**P**
V-Match (Hu et al. 2018c)		✓			✓		–	81.0	–	62.4*	56.5	68.2*	–	–	–	*D16, D17*
Dye-Net (Li and Change Loy 2018)	✓	✓		✓	✓	✓	–	86.2	–	69.1*	67.3*	71.0*	68.2	65.8	70.5	*D17*

Table 16 (continued)

Methods	S. techs			T. techs			2016 val set			2017 val set			2017 test-dev set			Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
SCO-VOS (Koh et al. 2018)			✓				78.3	79.6	77.0	67.7	66.5	68.8	–	–	–	**P**
MSGSTP (Hu et al. 2018b)			✓		✓		79.7	81.0	78.3	–	–	–	–	–	–	Non-DL method
S2S (Xu et al. 2018a)	✓			✓			–	79.1	–	–	–	–	–	–	–	*YV*
STVOS (Wang et al. 2018)			✓	✓			68.0	68.9	67.0	–	–	–	–	–	–	Non-DL method
PReMVOS (Luiten et al. 2018)	✓	✓		✓	✓		86.8	84.9	88.6	77.8	73.9	81.8	71.6	67.5	75.8	**C, Ma**, *D16, D17*
LucidTracker (Khoreva et al. 2019)	✓			✓	✓		85.7	86.6	84.8	–	–	–	66.6	63.4	69.9	*D16*

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. $\mathcal{J}$, $\mathcal{F}$: mean Jaccard-index and F-measure in Eq. 1. $\mathcal{J}\&\mathcal{F}$ is the average of $\mathcal{J}$ and $\mathcal{F}$. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based. The scores with '*' indicate they come from the non-original works using the original code

Table 17 Segmentation performance of the reviewed SVOS methods on the DAVIS dataset (including 2016 val set, 2017 val set and 2017 test-dev set) (part 2/2)

Methods	S. techs			T. techs			2016 val set			2017 val set			2017 test-dev set			Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
FEELVOS (Voigtlaender et al. 2019)		✓			✓		–	79.1	–	71.5	69.1	74.0	57.8	55.1	60.4	**C**, *D17, YV*
SiamMask (Wang et al. 2019a)		✓			✓		69.8	71.7	67.8	56.4	54.3	58.5	43.2	40.6	45.8	**C**, *IV, YV*
A-GAME (Johnander et al. 2019)					✓		81.9	81.5	82.2	71.0	68.5	73.6	52.3	49.2	55.3	**M, P**, *D17, YV*
STCNN (Xu et al. 2019a)	✓					✓	83.8	83.8	83.8	61.7	58.7	64.6	–	–	–	**M, P**, *D16*
MHPVOS (Xu et al. 2019c)	✓		✓		✓		88.6	87.6	89.5	76.1	73.4	78.9	69.5	66.4	72.7	**C**, *D17*
RVOS (Ventura et al. 2019)						✓	–	–	–	60.6	57.5	63.6	50.3	47.9	52.6	*D17, YV*
AGSS-VOS (Lin et al. 2019)		✓		✓	✓	✓	–	–	–	67.4	64.9	69.9	57.2	54.8	59.7	*YV*
RANet (Wang et al. 2019d)		✓			✓		87.1	86.6	87.6	65.7	63.2	68.2	55.4	53.4	57.3	**E, H, M, I, So**, *D16, D17*
DTN (Zhang et al. 2019)		✓		✓	✓		83.6	83.7	83.5	67.4	64.2	70.6	–	–	–	**C, P**, *D17*
DMM-Net (Zeng et al. 2019a)	✓	✓				✓	–	–	–	70.7	68.1	73.3	–	–	–	**C**, *D17, YV*
CapsVOS (Duarte et al. 2019)		✓				✓	–	–	–	–	–	–	51.3	47.4	55.2	*D17, YV*
STM (Oh et al. 2019)		✓			✓		89.4	88.7	90.1	81.7	79.2	84.3	72.2	69.3	75.2	**C, E, P, M**, *D17, YV*
TVOS (Zhang et al. 2020)		✓			✓		–	–	–	72.3	69.9	74.7	63.1	58.8	67.4	*D17, YV*
SAT(Chen et al. 2020)		✓			✓		83.1	82.6	83.6	72.3	68.6	76.0	–	–	–	**C**, *D17, YV*
FRTM(Xie et al. 2021)	✓						83.5	–	–	76.7	–	–	–	–	–	*D17*
LWL (Bhat et al. 2020)	✓						–	–	–	74.3	72.2	76.3	–	–	–	*D17*
EGMN (Lu et al. 2020a)		✓	✓		✓		–	–	–	82.8	80.2	85.2	–	–	–	**C, M**, *D17, YV*

Table 17 (continued)

Methods	S. techs			T. techs			2016 val set			2017 val set			2017 test-dev set			Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
KMN (Seong et al. 2020)		✓			✓		90.5	89.5	91.5	82.8	80.0	85.6	77.2	74.1	80.3	**C, E, P, M, S**, *D17, YV*
AFB-URR (Liang et al. 2020)		✓			✓		–	–	–	74.6	73.0	76.1	–	–	–	**C, E, P, M**, *D17*
CFBI+ (Yang et al. 2021b)		✓			✓		89.9	88.7	91.1	82.9	80.1	85.7	75.6	71.6	79.6	**C**, *D17, YV*
SSTVOS (Duke et al. 2021)		✓			✓	✓	–	–	–	82.5	79.9	85.1	–	–	–	*D17, YV*
SwiftNet (Wang et al. 2021a)		✓			✓		90.4	90.5	90.3	81.1	78.3	83.9	–	–	–	**C**, *D17, YV*
LCM (Hu et al. 2021)		✓			✓		90.7	89.9	91.4	83.5	80.5	86.5	78.1	74.4	81.8	**C, E, M**, *D17, YV*
RMNet (Xie et al. 2021)		✓		✓	✓		88.8	88.9	88.7	83.5	81.0	86.0	75.0	71.9	78.1	**C, E, P, M**, *D17, YV*
HMMN (Seong et al. 2021)		✓			✓		90.8	89.6	92.0	84.7	81.9	87.5	78.6	74.7	82.5	**C, E, P, M, S**, *D17, YV*
STCN (Cheng et al. 2021)		✓					91.6	90.8	92.5	85.4	82.2	88.6	77.8	74.3	81.3	**D, E, Hr, B, F**, *D17, YV*
AOT (Yang et al. 2021a)		✓			✓		91.1	90.1	92.1	84.9	82.3	87.5	79.6	75.9	83.3	**C, E, P, M, S**, *D17, YV*

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. $\mathcal{J}$, $\mathcal{F}$: mean Jaccard-index and F-measure in Eq. 1. $\mathcal{J}\&\mathcal{F}$ is the average of $\mathcal{J}$ and $\mathcal{F}$. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based. The scores with '*' indicate they come from the non-original works using the original code

5.1 Quantitative and qualitative results

In this section, the reviewed methods are discussed in both segmentation efficiency and accuracy. To make a fair comparison in computation efficiency, 9 representative SVOS methods are evaluated on the DAVIS-2016 dataset (Perazzi et al. 2016a), which consists of 20 video sequences and 1475 frames. The selected methods are: OSVOS (Caelles et al. 2017, online fine-tuning), MaskTrack (Perazzi et al. 2017, online fine-tuning, mask propagation), RGMP (Oh et al. 2018, matching, mask propagation, and long-term propagation), STM (Oh et al. 2019, matching, mask propagation), RANet (Wang et al. 2019d, matching, mask propagation), SiamMask (Wang et al. 2019a, matching, mask propagation), OSMN (Yang et al. 2018, variant of online fine-tuning, mask propagation), A-GAME (Johnander et al. 2019, variant of online fine-tuning, mask propagation) and RVOS (Ventura et al. 2019, long-term propagation). The evaluation is achieved with a single NVIDIA GeForce RTX 2080 Ti GPU card.

Table 15 illustrates the average accuracy and FPS of the representative methods on DAVIS-2016. In addition, several properties considered to be significant for the efficiency are listed, including the use of spatial and temporal techniques, involved frames, and the resolutions of input frames. Tables 16, 17, 19 demonstrate the quantitative results of the reviewed SVOS methods on four benchmark datasets: DAVIS-2016 validation set (Perazzi et al. 2016a), DAVIS-2017 validation set, DAVIS-2017 test-development set (Pont-Tuset et al. 2017), and YouTube-VOS test set (Xu et al. 2018b). The comparison results between the reviewed UVOS methods are shown in Table 20, where four benchmarks are considered for the performance evaluation: DAVIS-2016 validation set (Perazzi et al. 2016a), YouTube-Objects (Prest et al. 2012), SegTrack v2 (Li et al. 2013, and FBMS (Ochs et al. 2013).

The evaluation metrics in this section are $\mathcal{J}$, $\mathcal{F}$, and $\mathcal{J}\&\mathcal{F}$ introduced in Eqn 1. For the YouTube-VOS series (Xu et al. 2018b; Yang et al. 2019a; Xu et al. 2019b), since the datasets were divided into two subsets ('seen' and 'unseen'), different metrics are computed: $\mathcal{J}$-seen, $\mathcal{F}$-seen, $\mathcal{J}$-unseen and $\mathcal{F}$-unseen.

From Sect. 4.1, it is observed that online fine-tuning is a technique that fine-tunes the segmentation network with annotations during inference. Although much knowledge about the target object can be learned for accurate segmentation, an extra fine-tuning process is required. In our experiments, the number of iterations for online fine-tuning is set to 1000, which takes an average of 118 seconds for each video sequence. Therefore, the FPS values of OSVOS (Caelles et al. 2017) and MaskTrack (Perazzi et al. 2017) are the lowest two values of all listed methods. To improve the efficiency of VOS methods, online fine-tuning have been gradually replaced by its variants and the matching-based techniques.

The reference frames have a great impact on both SVOS efficiency and accuracy. As shown in Table 15, the first frame is mandatory since it contains the target object masks. Besides the first frame, the previous frame is also frequently utilised in most representative methods. The correspondence between the previous and target frames provides SVOS with short-term temporal correlations. Compared with other methods listed in Table 15, STM (Oh et al. 2019) and RVOS (Ventura et al. 2019) utilise more reference frames for segmentation. In RVOS, the network architecture is constructed based on ConvLSTM, thus long-term temporal information can be achieved to build long-range correlations between target objects and backgrounds. To perform VOS efficiently, RVOS resizes the input frames to half of the original resolution before training and inference. As discussed in Sect. 4.6.5, the recurrent module in RVOS is trained on five consecutive frames only due to the limited

Table 18 Segmentation performance of the reviewed SVOS methods on the YouTube-VOS-2018 val set

Methods	S. techs			T. techs			Overall	Seen		Unseen		Training data
	O	M	G	O	P	L		$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
OSVOS (Caelles et al. 2017)	✓						58.8*	60.5*	60.7*	59.8*	54.2*	*D16*
OnAVOS (Voigtlaender and Leibe 2017)	✓				✓		55.2*	62.7*	51.4*	60.1*	46.6*	**P**, *D16*
MaskTrack (Perazzi et al. 2017)	✓			✓	✓		53.1*	59.5*	47.9*	59.9*	45.0*	**E, M, So**, *D16*
OSMN (Yang et al. 2018)	✓			✓			51.2*	60.1*	44.0*	60.0*	40.6*	**C**, *D17*
RGMP (Oh et al. 2018)		✓		✓		✓	–	59.5*	–	45.2*	–	**E, M, P**, *D17*
S2S (Xu et al. 2018a)	✓			✓			70.0	66.9	74.1	66.8	72.3	*YV*
RVOS (Ventura et al. 2019)						✓	56.8	63.6	67.2	45.5	51.0	*D17, YV*
SiamMask (Wang et al. 2019a)		✓			✓		52.8	60.2	58.2	45.1	47.7	**C**, *IV, YV*
A-GAME (Johnander et al. 2019)					✓		–	67.8	–	60.8	–	**M, P**, *D17, YV*
CapsVOS (Duarte et al. 2019)		✓				✓	62.3	67.3	68.1	53.7	59.9	*D17, YV*
AGSS-VOS (Lin et al. 2019)		✓		✓	✓	✓	71.3	71.3	75.2	65.5	73.1	*YV*
DMM-Net (Zeng et al. 2019a)	✓	✓				✓	58.0	60.3	63.5	50.6	57.4	**C**, *D17, YV*
STM (Oh et al. 2019)		✓			✓		79.4	79.7	84.2	72.8	80.9	**C, E, P, M**, *D17, YV*
TVOS (Zhang et al. 2020)		✓			✓		67.8	67.1	69.4	63.0	71.6	*D17, YV*
SAT(Chen et al. 2020)		✓			✓		63.6	67.1	70.2	55.3	61.7	**C**, *D17, YV*

Table 18 (Continued)

Methods	S. techs			T. techs			Overall	Seen		Unseen		Training data
	O	M	G	O	P	L		$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
EGMN (Lu et al. 2020a)		✓	✓		✓		80.2	80.7	85.1	74.0	80.9	**C, M**, *D17, YV*
KMN (Seong et al. 2020)		✓			✓		81.4	81.4	85.6	75.3	83.3	**C, E, P, M, S**, *D17, YV*
FRTM (Bhat et al. 2020)	✓						72.1	72.3	76.2	65.9	74.1	*YV*
LWL (Bhat et al. 2020)	✓						80.2	78.3	82.3	75.6	84.4	*YV*
AFB-URR (Liang et al. 2020)		✓			✓		79.6	78.8	83.1	74.1	82.6	**C, E, P, M**, *YV*
CFBI+ (Yang et al. 2021b)		✓			✓		82.0	81.2	86.0	76.2	84.6	**C**, *D17, YV*
SSTVOS (Duke et al. 2021)		✓			✓	✓	81.7	81.2	–	76.0	–	*D17, YV*
SwiftNet (Wang et al. 2021a)		✓			✓		77.8	77.8	81.8	72.3	79.5	**C**, *D17, YV*
LCM (Hu et al. 2021)		✓			✓		82.0	82.2	86.7	75.7	83.4	**C, E, M**, *D17, YV*
RMNet (Xie et al. 2021)		✓		✓	✓		81.5	82.1	85.7	75.7	82.4	**C, E, P, M**, *D17, YV*
HMMN (Seong et al. 2021)		✓			✓		82.6	82.1	87.0	76.8	84.6	**C, E, P, M, S**, *D17, YV*
STCN (Cheng et al. 2021)		✓					83.0	81.9	86.5	77.9	85.7	**D, E, Hr, B, F**, *D17, YV*
AOT (Yang et al. 2021a)		✓			✓		84.1	83.7	88.5	78.1	86.1	**C, E, P, M, S**, *D17, YV*

Table 18 (Continued)

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based. The scores with '*' indicate they come from the non-original works using the original code

Table 19 Segmentation performance of the reviewed SVOS methods on the YouTube-VOS-2019 val set

Methods	S. techs			T. techs			Overall	Seen		Unseen		Training data
	O	M	G	O	P	L		$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
STM (Oh et al. 2019)		✓			✓		79.3*	79.8*	83.8*	73.0*	80.5*	**C, E, P, M**, *D17, YV*
KMN (Seong et al. 2020)		✓			✓		80.0*	80.4*	84.5*	73.8*	81.4*	**C, E, P, M, S**, *D17, YV*
CFBI+ (Yang et al. 2021b)		✓			✓		82.9	80.6	85.2	78.9	86.8	**C**, *D17, YV*
SSTVOS(Duke et al. 2021)		✓			✓	✓	81.8	80.9	–	76.6	–	*D17, YV*
HMMN (Seong et al. 2021)		✓			✓		82.5	81.7	86.1	77.3	85.0	**C, E, P, M, S**, *D17, YV*
STCN (Cheng et al. 2021)		✓					82.7	81.1	85.4	78.2	85.9	**D, E, Hr, B, F**, *D17, YV*
AOT (Yang et al. 2021a)		✓			✓		84.1	83.5	88.1	78.4	86.3	**C, E, P, M, S**, *D17, YV*

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based. The scores with '*' indicate they come from the non-original works using the original code

Table 20 Segmentation performance of the reviewed UVOS methods on DAVIS-2016 val set, YouTube-Objects (abbr. as YTBO), SegTrack v2 (abbr. as STV2), and FBMS

Methods	S. techs			T. techs			DAVIS 2016 val set			YTO	STV2	FBMS		Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{J}$	$\mathcal{J}$	$\mathcal{F}$	
CCNN (Li et al. 2017b)		✓					–	–	–	63.3	67.0	–	–	**E, M, D, H**
MP-Net (Tokmakov et al. 2017a)				✓			68.0	69.7	66.3	–	–	35.7*	77.5*	*FT3D*
FusionSeg (Jain et al. 2017)				✓			–	71.5	–	68.4	61.4	68.4	–	**P**, *ImageNet-Video*
SegFlow (Cheng et al. 2017)	✓			✓			66.1	67.4	64.7	57.1*	–	56.0*	63.4*	*DAVIS-2016*
VM-VOS (Tokmakov et al. 2017b)					✓	✓	74.0	75.9	72.1	67.5*	57.3	64.7*	77.8	*DAVIS-2016*
IET-VOS (Li et al. 2018b)			✓	✓			77.4	78.6	76.1	–	59.3	71.9*	82.8	*DAVIS-2016*
SCO-VOS (Koh et al. 2018)			✓				78.3	79.6	77.0	–	–	–	81.6	**P**
MSGSTP (Hu et al. 2018b)			✓		✓		76.3	77.6	75.0	–	70.1	60.8	–	Non-DL method
MBN-VOS (Li et al. 2018c)			✓	✓	✓		79.5	80.4	78.5	–	–	73.9*	83.2*	*DAVIS-2016*
MGCRN (Hu et al. 2018a)	✓			✓	✓		76.5	76.4	76.6	–	–	–	–	**P**, *DAVIS-2016*
PDB (Song et al. 2018)						✓	75.9	77.2	74.5	–	–	74.0	81.5	**M, D**, *DAVIS-2016*
COSNet (Lu et al. 2019)		✓					80.4	81.1	79.7	71.0	–	74.8	–	**M, D**, *DAVIS-2016*

Table 20 (Continued)

Methods	S. techs			T. techs			DAVIS 2016 val set			YTO	STV2	FBMS		Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{J}$	$\mathcal{J}$	$\mathcal{F}$	
AD-Net (Yang et al. 2019b)		✓					81.1	81.7	80.5	–	–	–	81.2	*DAVIS-2016*
AGNN (Wang et al. 2019b)			✓				79.9	80.7	79.1	70.8	–	–	–	**M, D**, *DAVIS-2016*
AGS (Wang et al. 2020)						✓	78.6	79.7	77.4	69.7	–	76.0	87.4	**P, D**, *D-2016, YTO, STV2*
EGMN (Lu et al. 2020a)		✓	✓		✓		81.9	82.5	81.2	71.4	–	–	–	**M, D**, *DAVIS-2016*

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. $\mathcal{J}$, $\mathcal{F}$: mean Jaccard-index and F-measure in Eq. 1. $\mathcal{J}\&\mathcal{F}$ is the average of $\mathcal{J}$ and $\mathcal{F}$. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based. The scores with '*' indicate they come from the non-original works using the original code

Table 21 Segmentation performance of the reviewed UVOS methods on the DAVIS-2017 unsupervised dataset

Methods	S. techs			T. techs			Val set			Test-dev set			Training data
	O	M	G	O	P	L	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	
PDB (Song et al. 2018)						✓	55.1	53.2	57.0	40.4	37.7	43.0	**M, D**, *DAVIS-2016*
RVOS (Ventura et al. 2019)						✓	41.2	36.8	45.7	22.5	17.7	27.3	*DAVIS-2017, YouTube-VOS*
AGNN (Wang et al. 2019b)			✓				61.1	58.9	63.2	–	–	–	**M, D**, *DAVIS-2016*
AGS (Wang et al. 2020)						✓	57.5	55.5	59.5	45.6	42.1	49.0	**P, D**, *DAVIS-2016, YTO, STV2*
UnOVOST (Luiten et al. 2020)		✓				✓	67.9	66.4	69.3	58.0	54.0	62.0	**C**, *YouTube-VOS*
TAODA (Zhou et al. 2021)	✓					✓	65.0	63.7	66.2	59.8	56.0	63.7	*DAVIS-2017, YouTube-VIS*

S. techs spatial techniques, *O* online fine-tuning, *M* matching, *G* graph, *T. techs* temporal techniques, *O* optical flow;*P* mask propagation, *L* long-term temporal propagation. '–': Not Given. C: COCO, D: DUTS, E: ECSSD, Hr: HRSOD, H: HKU-IS, P: PASCAL VOC, M: MSRA10K, Ma: Mapillary, S: SBD, So: SOC, I: ILSO, B: BIG, F: FS-1000, D16: DAVIS-2016, D17: DAVIS-2017, IV: ImageNet-Video, YV: YouTube-VOS. **BOLD datasets**: image-based, *ITALIC datasets*: video-based

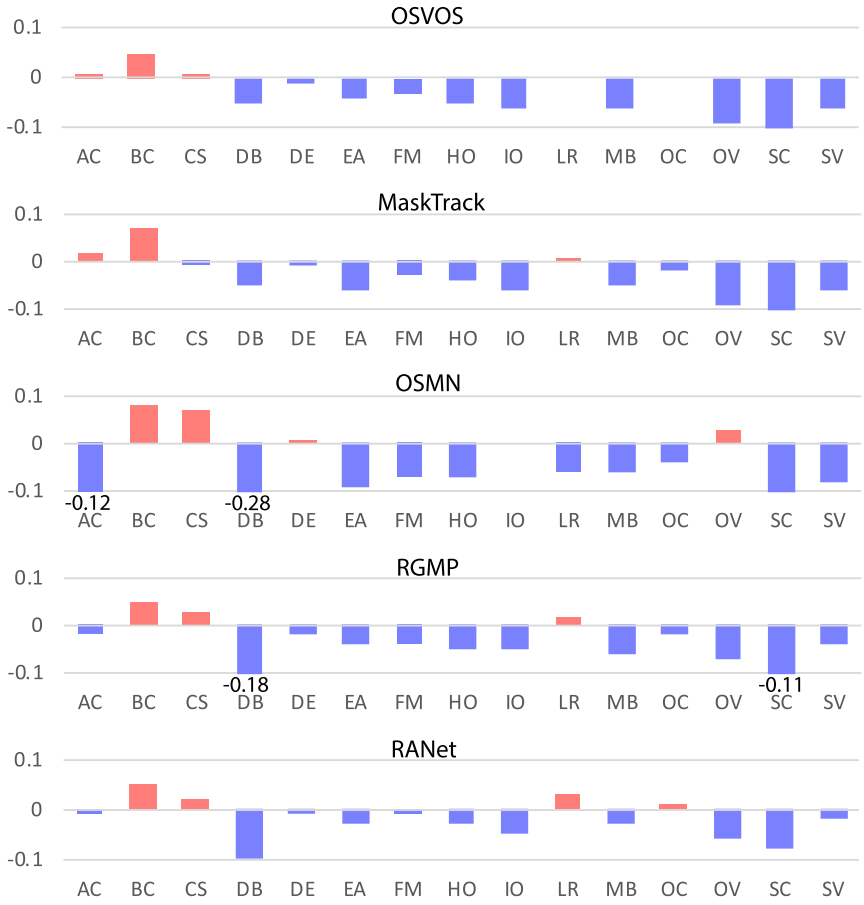

Fig. 23 The performance change due to the specific challenge attributes (part 1/2). For each attribute on the x-axis, the value measures the performance difference between evaluating the VOS methods on the dataset with and without the sequences related to the attribute. Positive values (red bars) or negative values (purple bars) indicate the performance increases or decreases when considering the specific challenging sequences. The range for the changes is set as $[-0.1, 0.1]$, the values out of bounds are marked in the corresponding bars. Note that all challenge attributes are abbreviated due to the space limitation. AC: appearance changes; BC: background clutter; CS: camera shake; DB: dynamic background; DE: non-linear deformation; EA: edge ambiguity; FM: fast-motion; HO: heterogeneous object; IO: interacting objects; LR: low resolution; MB: motion blur; OC: occlusions; OV: out-of-view; SC: shape complexity; SV: scale variation. Best viewed in colour

computation resources. Although RVOS uses all previous frames during inference, such training setting still limits the effectiveness of the recurrent module in spatial-temporal information accumulation. The quantitive results in Tables 17 and 19 show that there are still gaps between RVOS and state-of-the-art methods. Unlike RVOS, STM considers the intermediate frames as the reference, which enables the segmentation network to better adapt to the changes in target objects over time. To achieve a better trade-off between efficiency and accuracy, the intermediate frames are sampled every five frames. Compared

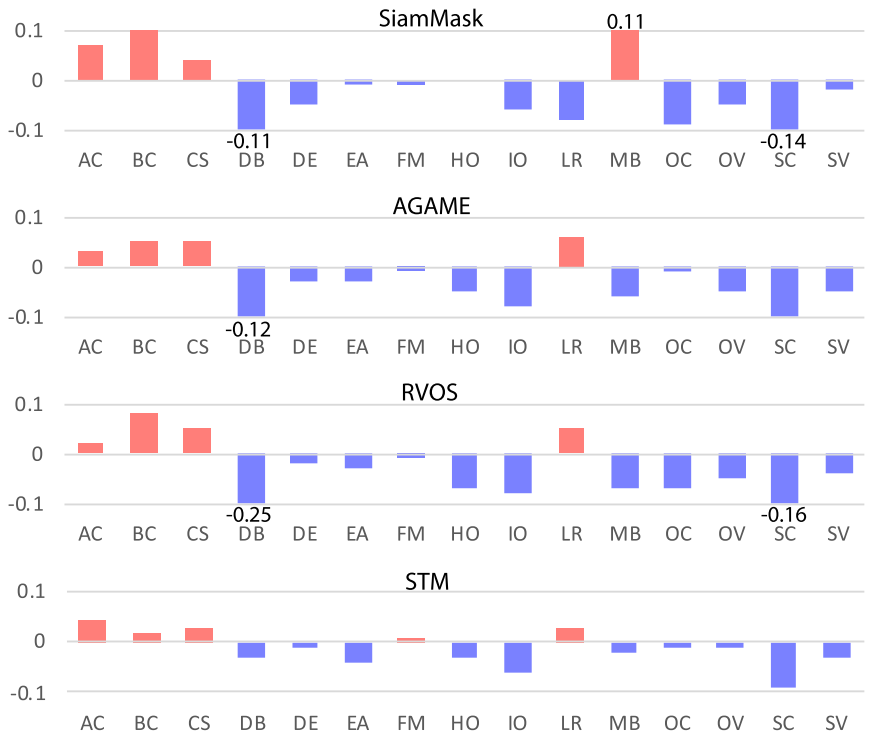

Fig. 24 The performance change due to the specific challenge attributes (part 2/2). Please refer to the caption of Fig. 23 for the definitions and abbreviations in the bar charts. Best viewed in colour

Fig. 25 Negative results due to 'dynamic background' (DB) and 'shape complexity' (SC). Top row: results predicted by RVOS (Ventura et al. 2019) on DB sequence; Bottom row: results predicted by STM (Oh et al. 2019) on SC sequence. Red masks are the ground truth masks, which are covered by the green masks (the predicted results)

with other methods listed in Table 15, STM achieves better performance on all benchmark datasets while simultaneously keeping acceptable efficiency (Table 18).

Besides the network architecture, the video datasets are also important to the segmentation performance. As shown in Tables 16, 17, and 19, there are three frequently used video datasets in SVOS: DAVIS-2016 (Perazzi et al. 2016a, 20 training sequences), DAVIS-2017 (Pont-Tuset et al. 2017, 60 training sequences) and YouTube-VOS (Xu et al. 2018b,

3400+ training sequences). Seen from the reported performance, it is clear that most of the methods trained on DAVIS-2017 and YouTube-VOS perform better than those trained on DAVIS-2016. To some extent, this reflects that more training sequences are more likely to make the VOS methods achieve high-quality results. Compared with DAVIS-2017, the methods trained on YouTube-VOS are more likely to achieve state-of-the-art performance (e.g. STM and its extensions) since YouTube-VOS has much more annotated frames and longer duration within each video sequence. For example, AFB-URR (Liang et al. 2020), as a STM-based method, cannot perform as good as other STM-based ones (e.g., EGMN, KMN, and SwiftNet) in Table 17 because YouTube-VOS is not considered during training. Besides video datasets, image datasets also play a key role in VOS training. With the pixel-level annotations in the image datasets far more than that in the video datasets, more general knowledge for object detection and segmentation can be learned by pre-training VOS methods on these datasets. Generally, the datasets for image segmentation (COCO (Lin et al. 2014), PASCAL VOC (Everingham et al. 2015), Mapillary Vistas (Neuhold et al. 2017), SBD (Hariharan et al. 2011), BIG (Cheng et al. 2020), FSS-1000 (Li et al. 2020)) and saliency detection (DUTS (Wang et al. 2017a), ECSSD (Shi et al. 2015a), HKU-IS (Li and Yu 2015), MSRA10k (Cheng et al. 2014), HRSOD (Zeng et al. 2019b), SOC (Fan et al. 2018), ILSO (Li et al. 2017a)) are considered during pre-training.

Table 20 shows the performance of UVOS methods on four datasets (DAVIS-2016 (Perazzi et al. 2016a), YouTube-Objects (Prest et al. 2012), SegTrack v2 (Li et al. 2013), and FBMS (Ochs et al. 2013)). It is obvious that DAVIS-2016 has been the primary benchmark dataset for UVOS evaluation. This is mainly because DAVIS-2016 has more video sequences with high and unified resolutions, diverse object categories, and challenges than other datasets. Since no supervision signals are required, the UVOS methods have to segment the primary object from the input sequence by themselves. Therefore, recent UVOS methods mostly learn salient object recognition via pre-training on extra image datasets (for saliency detection). The table also shows that the recent matching-based methods outperform other UVOS ones, which validates the effectiveness of dense affinity between frames in segmenting the object appearing simultaneously in multiple frames (Table 21).

To further explore the representative methods on different challenging sequences, we refer to the challenge attributes of each video in DAVIS-2016 dataset, and provide the quantitative results (mean $\mathcal{J}$ & $\mathcal{F}$) of these methods on each challenge attribute in Fig. 23 and Fig. 24. For more information about these challenge attributes, please refer to the supplemental material[4] of DAVIS-2016 dataset (Perazzi et al. 2016b).

Figures 23 and 24 demonstrate that the evaluated methods perform well on the sequences with 'background clutter' and 'camera shake'. In the sequences with these two challenges, the target objects are generally moving in a relatively still scene, where the 'background clutter' sequences contain many confusing background objects or regions, and the 'camera shake' indicates that the scenes being captured are affected by the irregular shakes. Although these challenges pose an obstacle to computing discriminative clues and motion patterns, the previous frame can provide confident estimates for the segmentation due to the relative still background and smooth movement of the target objects. Therefore, the evaluated methods (most of them consider $\mathcal{I}_{t-1}$ during inference) achieve high-quality results on the sequences with these two challenges.

[4] https://davischallenge.org/files/davis_supplementary.eps.

Among the challenges listed in Figs. 23 and 24, 'dynamic background' and 'shape complexity' appear to be the two hardest challenges to handle. Similar to 'background clutter', the target objects in the sequences with 'dynamic background' are surrounded by the ambiguous backgrounds. But on top of this, some of the background objects are moving around the scene, thus making the segmentation of target object more challenging. The challenge 'shape complexity' indicates that the target object is intricate, generally consisting of irregularly shaped parts and small details. Due to the partial loss of detailed information during feature encoding, the predicted masks are generally smoother than corresponding ground truth results, limiting the accuracy of existing VOS methods on the sequences with 'shape complexity'. To demonstrate the influence of these two challenges on the VOS performance, we provide the qualitative results of some representative methods on related sequences in Fig. 25. From the segmentation results (predicted by RVOS, Ventura et al. 2019) for the sequences with 'dynamic background', it is observed that the moving backgrounds (rising smoke) significantly reduce the quality of the predicted mask. This is because the segmentation network mistakenly estimates the motion patterns of the target object. In the bottom sequence with 'shape complexity' in Fig. 25, the target objects are a boy and a bmx bicycle. Although STM (Oh et al. 2019) successfully locates where the target object is, many small details are missing (especially the bmx bicycle parts) from the predicted masks due to its decoder cannot completely recover the detailed information.

5.2 Discussion and future research directions

In this paper, a variety of VOS methods have been reviewed in terms of their techniques, contributions and highlights. To sum up, VOS is a task of extracting pixel-level masks for target objects from each video frame while simultaneously keeping the global consistency of these objects throughout the sequence. For SVOS, the target objects generally indicate the objects annotated in the first frame. For UVOS (single-object), the target objects correspond to the salient objects, or moving objects in the video sequence. Due to its ability of producing fine-grained object masks, VOS has received great attention in the computer vision community, and facilitated many related applications. To further boost the progress in this research field, this paper summarises several factors affecting the segmentation performance of VOS methods.

The first factor is the training data. With complex scenes in real-world sequences and large number of parameters in the deep learning-based methods, the quality and quantity of annotated video data are crucial to derive high-quality segmentation results. In Sect. 3.1, the commonly used video datasets are discussed. Based on the statistics of SVOS and UVOS mentioned above, and also the features of existing VOS datasets shown in Table 1, it is observed that DAVIS-2016 (Perazzi et al. 2016a), 2017 (Pont-Tuset et al. 2017), and YouTube-VOS-2018 (Xu et al. 2018b), 2019 (Xu et al. 2019b) are the main datasets for model training, where DAVIS-2016 is used for training UVOS and SVOS methods, while the others serve SVOS methods only. To further improve their capabilities of separating objects from backgrounds, many methods train their segmentation network using a two-stage training process. At first, they utilise the datasets for static image segmentation (e.g. PASCAL VOC, Everingham et al. 2012, MSRA 10K, Cheng et al. 2014, and MS COCO, Lin et al. 2014) so as to allow their models to recognise and segment general objects. Next, the VOS datasets are employed to fine-tune the models to adapt video data.

Apart from datasets, the techniques used to preserve the spatial and temporal consistency of target objects are also important in improving the accuracy and efficiency of VOS methods. To preserve spatial consistency, the commonly employed techniques are online fine-tuning-based, matching-based and graph-based ones. From the aforementioned discussion in Sect. 4, it can be concluded that online fine-tuning-based methods focus on fine-tuning the segmentation network with annotated objects. Although accurate segmentation results can be achieved, extra time for fine-tuning is needed. On the other hand, for the sake of efficiency, these methods are gradually replaced by matching-based methods. Unlike online fine-tuning-based methods, the matching-based ones build the correspondence between target objects from different frames by comparing their features directly. Since similarity measurement is much faster than model fine-tuning, and can also achieve desirable and even better segmentation results, matching-based methods have been increasingly integrated into recent VOS works. Although graph-based methods are essentially built based on feature similarities, they establish the object correspondence between multiple frames, which is much more time consuming due to the complicated graph construction and optimisation procedures.

For temporal correlation, the commonly employed techniques are optical flow-based, mask propagation-based and long-term temporal information-based ones. From Table 15, it can be seen that the early generation of VOS methods mainly utilises optical flow and mask propagation to capture the changes of target objects in the local temporal domain. To obtain reliable optical flow between consecutive frames, most of the related methods incorporate individual networks for flow estimation (e.g. FlowNet 2.0, Ilg et al. 2017) into their segmentation models. In this way, the number of parameters of VOS methods is increased further, which is a burden for lightweight segmentation. In addition, optical flow-based methods cannot handle the video sequences with 'dynamic background', because the moving background objects are also extracted. Therefore, recent VOS methods mostly utilise mask propagation to guide the segmentation for the current frame. As for long-term temporal information-based methods, several architectures for sequential analysis or prediction are employed to capture temporal guidance from continuous frames, such as RNN and GAN. Limited by the computation resource, existing methods generally train their model with a small amount of frames during a single iteration (as shown in Table 14), which restricts the performance of long-term temporal information analysis.

The last important factor is model architecture. For VOS methods, encoder-decoder is the most frequently used network architecture for building segmentation models, where encoder computes the semantic features from video frames, and decoder serves for restoring the processed feature map to the resolution of original input data. To better utilise the intra- and inter-frame information, most of existing VOS methods focus on developing an effective encoder and a module for feature analysis and integration. In a standard encoder-decoder architecture for VOS, the networks for image classification and segmentation (e.g. VGGNet, Simonyan and Zisserman 2015, ResNet, He et al. 2016 or DeepLab, Chen et al. 2015a, 2017a, b, 2018a) are implemented to compute semantic features. After integrating extra guidance from other frames, the obtained feature maps are converted to a map containing masks of target objects. On top of the standard architectures, several VOS methods choose to integrate other visual networks to further enhance feature representation and improve the segmentation results, e.g. Mask R-CNN, He et al. 2017 or FCIS, Li et al. 2017c for detecting object proposals, FlowNet (Ilg et al. 2017) for computing optical flow. These networks are independent of standard architectures, thus extra memory and time are needed to gain guidance from them. In general, the more sophisticated the architecture of VOS method is, the more accurate segmentation results can be achieved.

Based on the discussed methods and evaluation results, we present several future research directions that would be beneficial for the field of VOS:

- **Large-scale video datasets with dense annotations**: At the point of submitting this paper, there have been two main families of video datasets serving VOS tasks: DAVIS, (Perazzi et al. 2016a; Caelles et al. 2019; Pont-Tuset et al. 2017) and YouTube-VOS (Xu et al. 2018b; Yang et al. 2019a; Xu et al. 2019b), where DAVIS datasets have dense annotations, i.e. all video frames in the datasets have manually annotated object masks. In contrast, though YouTube-VOS datasets contain many more video sequences, the annotations are only provided with every 5th frame, which limits the further exploration of VOS methods into long-term temporal correlations. Therefore, a dataset with large-scale and dense annotations will allow VOS methods to achieve further performance improvement.
- **Long-term temporal information analysis**: As we have mentioned in this section, existing VOS methods based on long-term temporal information mostly train their recurrent modules or prediction modules using 4-11 continuous frames. In general, a video sequence in YouTube-VOS has over 200 frames, thus existing methods obviously cannot establish a comprehensive temporal correlation for the whole sequence. Because analysing the long-term changes of temporal information is beneficial for tackling several challenges such as occlusion, out-of-view and fast motion, it is desirable for future VOS methods to develop a more efficient training process that allows information propagation and accumulation for a longer temporal period.
- **Balance between VOS accuracy and efficiency**: The design for segmentation networks is essential to balance the VOS accuracy and efficiency. Unfortunately, it is still hard for the existing methods to achieve such a balance since the high-quality results generally call for deeper backbones or more elaborate processes, slowing down the VOS process. For example, among the discussed SVOS methods, the backbone network is the main difference between the methods obtaining the best accuracy (STCN, based on ResNet-50) and the best efficiency (SwiftNet, based on ResNet-18), as shown in Fig. 6. Therefore, a lightweight but discriminative network is required in future VOS methods to promote the VOS methods in real-time applications.
- **Multi-object UVOS**: Visual saliency and motion information are the most frequently used information in the current UVOS methods. However, such information is less effective to discriminate different object instances. Therefore, most existing UVOS methods can infer only a single object from the input video sequence, which rarely plays a role in real-world applications. To address this problem, several datasets for multi-object UVOS (Caelles et al. 2019; Yang et al. 2019a) have been proposed recently, which are already mentioned in Sect. 2. Due to the integration of instance-level segmentation modules, some recent works (Luiten et al. 2020; Zhou et al. 2021) have achieved good results on these datasets, validating the feasibility of this direction and encouraging future improvements.

6 Conclusion

In this paper, many recently proposed deep learning methods for VOS have been discussed. To highlight their contributions, these methods are categorised into six main groups: online fine-tuning-based, feature matching-based, graph-based, optical flow-based, mask

propagation-based and long-term temporal information-based methods. For each category of methods, we outline their main algorithmic contributions and summarise their advantages and disadvantages. Through the analysis of quantitative and qualitative results, we validate the contributions of network architectures and training datasets to VOS performance. Finally, we give an overview of the challenges in this field and future research trends.

References

Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder–decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell 39(12):2481–2495

Ballas N, Yao L, Pal C, Courville AC (2016) Delving deeper into convolutional networks for learning video representations. In: Proceedings of the International Conference on Learning Representations

Bao L, Wu B, Liu W (2018) Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5977–5986

Bertinetto L, Valmadre J, Henriques JF, Vedaldi A, Torr PH (2016) Fully-convolutional siamese networks for object tracking. In: Proceedings of the European Conference on Computer Vision, Springer, pp 850–865

Bhat G, Lawin FJ, Danelljan M, Robinson A, Felsberg M, Van Gool L, Timofte R (2020) Learning what to learn for video object segmentation. In: Proceedings of the European Conference on Computer Vision, Springer, pp 777–794

Brox T, Malik J (2010) Large displacement optical flow: descriptor matching in variational motion estimation. IEEE Trans Pattern Anal Mach Intell 33(3):500–513

Brox T, Malik J (2010b) Object segmentation by long term analysis of point trajectories. In: Proceedings of the European Conference on Computer Vision, Springer, pp 282–295

Caelles S, Maninis KK, Pont-Tuset J, Leal-Taixé L, Cremers D, Van Gool L (2017) One-shot video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 221–230

Caelles S, Pont-Tuset J, Perazzi F, Montes A, Maninis KK, Van Gool L (2019) The 2019 davis challenge on vos: Unsupervised multi-object segmentation. arXiv preprint arXiv:190500737

Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: Proceedings of the European Conference on Computer Vision, Springer, pp 213–229

Chan TF, Vese LA (2001) Active contours without edges. IEEE Trans Image Process 10(2):266–277

Chen LC, Papandreou G, Schroff F, Adam H (2017b) Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:170605587

Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H (2018a) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision, pp 801–818

Chen L, Shen J, Wang W, Ni B (2015) Video object segmentation via dense trajectories. IEEE Trans Multimedia 17(12):2225–2234

Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2017) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848

Cheng HK, Chung J, Tai YW, Tang CK (2020) Cascadepsp: toward class-agnostic and very high-resolution segmentation via global and local refinement. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8890–8899

Cheng HK, Tai YW, Tang CK (2021) Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In: Proceedings of the Advances in Neural Information Processing Systems

Cheng MM, Mitra NJ, Huang X, Torr PH, Hu SM (2014) Global contrast based salient region detection. IEEE Trans Pattern Anal Mach Intell 37(3):569–582

Cheng J, Tsai YH, Hung WC, Wang S, Yang MH (2018) Fast and accurate online video object segmentation via tracking parts. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7415–7424

Cheng J, Tsai YH, Wang S, Yang MH (2017) Segflow: Joint learning for video object segmentation and optical flow. In: Proceedings of the IEEE International Conference on Computer Vision, pp 686–695

Chen X, Li Z, Yuan Y, Yu G, Shen J, Qi D (2020) State-aware tracker for real-time video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 9384–9393

Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2015a) Semantic image segmentation with deep convolutional nets and fully connected crfs. In: Proceedings of the International Conference on Learning Representations

Chen Y, Pont-Tuset J, Montes A, Van Gool L (2018b) Blazingly fast video object segmentation with pixel-wise metric learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1189–1198

Chien SY, Ma SY, Chen LG (2002) Efficient moving object segmentation algorithm using background registration technique. IEEE Trans Circuits Syst Video Technol 12(7):577–586

Chockalingam P, Pradeep N, Birchfield S (2009) Adaptive fragments-based tracking of non-rigid objects using level sets. In: Proceedings of the IEEE International Conference on Computer Vision, IEEE, pp 1530–1537

Chollet F (2017) Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1251–1258

Ci H, Wang C, Wang Y (2018) Video object segmentation by learning location-sensitive embeddings. In: Proceedings of the European Conference on Computer Vision, pp 501–516

Cucchiara R, Grana C, Piccardi M, Prati A (2003) Detecting moving objects, ghosts, and shadows in video streams. IEEE Trans Pattern Anal Mach Intell 25(10):1337–1342

Culibrk D, Marques O, Socek D, Kalva H, Furht B (2007) Neural network approach to background modeling for video object segmentation. IEEE Trans Neural Netw 18(6):1614–1627

De Vries H, Strub F, Mary J, Larochelle H, Pietquin O, Courville AC (2017) Modulating early visual processing by language. In: Proceedings of the Advances in Neural Information Processing Systems, pp 6594–6604

Duarte K, Rawat YS, Shah M (2019) Capsulevos: Semi-supervised video object segmentation using capsule routing. In: Proceedings of the IEEE International Conference on Computer Vision, pp 8480–8489

Duke B, Ahmed A, Wolf C, Aarabi P, Taylor GW (2021) Sstvos: Sparse spatiotemporal transformers for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5912–5921

Endres I, Hoiem D (2010) Category independent object proposals. In: Proceedings of the European Conference on Computer Vision, Springer, pp 575–588

Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88(2):303–338

Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A (2015) The pascal visual object classes challenge: a retrospective. Int J Comput Vis 111(1):98–136

Everingham M, Van Gool L, Williams C, Winn J, Zisserman A (2012) The pascal visual object classes challenge 2012 (voc2012) results (2012). In: URL http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html

Faktor A, Irani M (2014) Video segmentation by non-local consensus voting. In: Proceedings of the British Machine Vision Conference, vol 2, p 8

Fan DP, Cheng MM, Liu JJ, Gao SH, Hou Q, Borji A (2018) Salient objects in clutter: Bringing salient object detection to the foreground. In: Proceedings of the European Conference on Computer Vision, pp 186–202

Fan Q, Zhong F, Lischinski D, Cohen-Or D, Chen B (2015) Jumpcut: non-successive mask transfer and interpolation for video cutout. ACM Trans Graph 34(6):195

Fragkiadaki K, Zhang G, Shi J (2012) Video segmentation by tracing discontinuities in a trajectory embedding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1846–1853

Garcia-Garcia A, Orts-Escolano S, Oprea S, Villena-Martinez V, Martinez-Gonzalez P, Garcia-Rodriguez J (2018) A survey on deep learning techniques for image and video semantic segmentation. Appl Soft Comput 70:41–65

Ghosh S, Das N, Das I, Maulik U (2019) Understanding deep learning techniques for image segmentation. ACM Comput Surv 52(4):1–35

Girshick R (2015) Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1440–1448

Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 580–587

Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Proceedings of the Advances in Neural Information Processing Systems, pp 2672–2680

Griffin BA, Corso JJ (2019) Bubblenets: Learning to select the guidance frame in video object segmentation by deep sorting frames. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8914–8923

Han J, Yang L, Zhang D, Chang X, Liang X (2018) Reinforcement cutting-agent learning for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 9080–9089

Hariharan B, Arbeláez P, Bourdev L, Maji S, Malik J (2011) Semantic contours from inverse detectors. In: Proceedings of the IEEE International Conference on Computer Vision, IEEE, pp 991–998

He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916

He K, Gkioxari G, Dollár P, Girshick R (2017) Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp 2961–2969

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 770–778

Hinton GE, Sabour S, Frosst N (2018) Matrix capsules with EM routing. In: Proceedings of the International Conference on Learning Representations

Hu YT, Chen HS, Hui K, Huang JB, Schwing AG (2019) Sail-vos: Semantic amodal instance level video object segmentation-a synthetic dataset and baselines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3105–3115

Hu YT, Huang JB, Schwing A (2017) Maskrnn: Instance level video object segmentation. In: Proceedings of the Advances in Neural Information Processing Systems, pp 325–334

Hu YT, Huang JB, Schwing AG (2018b) Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation. In: Proceedings of the European Conference on Computer Vision, pp 786–802

Hu YT, Huang JB, Schwing AG (2018c) Videomatch: Matching based video object segmentation. In: Proceedings of the European Conference on Computer Vision, pp 54–70

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4700–4708

Hu P, Wang G, Kong X, Kuen J, Tan YP (2018a) Motion-guided cascaded refinement network for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1400–1409

Hu L, Zhang P, Zhang B, Pan P, Xu Y, Jin R (2021) Learning position and target consistency for memory-based video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4144–4154

Ilg E, Mayer N, Saikia T, Keuper M, Dosovitskiy A, Brox T (2017) Flownet 2.0: Evolution of optical flow estimation with deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2462–2470

Jain SD, Grauman K (2014) Supervoxel-consistent foreground propagation in video. In: Proceedings of the European Conference on Computer Vision, Springer, pp 656–671

Jain SD, Xiong B, Grauman K (2017) Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 2117–2126

Jampani V, Gadde R, Gehler PV (2017) Video propagation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 451–461

Jampani V, Kiefel M, Gehler PV (2016) Learning sparse high dimensional filters: Image filtering, dense crfs and bilateral neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4452–4461

Jang WD, Kim CS (2017) Online video object segmentation via convolutional trident network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5849–5858

Johnander J, Danelljan M, Brissman E, Khan FS, Felsberg M (2019) A generative appearance model for end-to-end video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8953–8962

Khoreva A, Benenson R, Ilg E, Brox T, Schiele B (2019) Lucid data dreaming for video object segmentation. Int J Comput Vis 127(9):1175–1197

Kim C, Hwang JN (2002) Fast and automatic video object segmentation and tracking for content-based applications. IEEE Trans Circuits Syst Video Technol 12(2):122–129

Koh YJ, Lee YY, Kim CS (2018) Sequential clique optimization for video object segmentation. In: Proceedings of the European Conference on Computer Vision, Springer, pp 537–556

Krähenbühl P, Koltun V (2011) Efficient inference in fully connected crfs with gaussian edge potentials. In: Proceedings of the Advances in Neural Information Processing Systems, pp 109–117

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp 1097–1105

LaLonde R, Bagci U (2018) Capsules for object segmentation. arXiv preprint arXiv:180404241

Lee YJ, Kim J, Grauman K (2011) Key-segments for video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, IEEE, pp 1995–2002

Liang Y, Li X, Jafari N, Chen J (2020) Video object segmentation with adaptive feature bank and uncertain-region refinement. In: Proceedings of the Advances in Neural Information Processing Systems 33

Li X, Change Loy C (2018) Video object segmentation with joint re-identification and attention-aware mask propagation. In: Proceedings of the European Conference on Computer Vision, pp 90–105

Li F, Kim T, Humayun A, Tsai D, Rehg JM (2013) Video segmentation by tracking many figure-ground segments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2192–2199

Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2117–2125

Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: Common objects in context. In: Proceedings of the European Conference on Computer Vision, Springer, pp 740–755

Lin H, Qi X, Jia J (2019) Agss-vos: Attention guided single-shot video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 3949–3957

Li Y, Qi H, Dai J, Ji X, Wei Y (2017c) Fully convolutional instance-aware semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2359–2367

Li S, Seybold B, Vorobyov A, Fathi A, Huang Q, Jay Kuo CC (2018b) Instance embedding transfer to unsupervised video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6526–6535

Li S, Seybold B, Vorobyov A, Lei X, Jay Kuo CC (2018c) Unsupervised video object segmentation with motion-based bilateral networks. In: Proceedings of the European Conference on Computer Vision, pp 207–223

Liu Y, Zhang Q, Zhang D, Han J (2019) Employing deep part-object relationships for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1232–1241

Li X, Wei T, Chen YP, Tai YW, Tang CK (2020) Fss-1000: A 1000-class dataset for few-shot segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2869–2878

Li G, Xie Y, Lin L, Yu Y (2017a) Instance-level salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2386–2395

Li B, Yan J, Wu W, Zhu Z, Hu X (2018a) High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8971–8980

Li G, Yu Y (2015) Visual saliency based on multiscale deep features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5455–5463

Li J, Zheng A, Chen X, Zhou B (2017b) Primary video object segmentation via complementary cnns and neighborhood reversible flow. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1417–1425

Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3431–3440

Luiten J, Voigtlaender P, Leibe B (2018) Premvos: Proposal-generation, refinement and merging for video object segmentation. In: Proceedings of the Asian Conference on Computer Vision, pp 565–580
Luiten J, Zulfikar IE, Leibe B (2020) Unovost: Unsupervised offline video object segmentation and tracking. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 2000–2009
Lu X, Wang W, Danelljan M, Zhou T, Shen J, Van Gool L (2020a) Video object segmentation with episodic graph memory networks. In: Proceedings of the European Conference on Computer Vision, Springer, pp 661–679
Lu X, Wang W, Ma C, Shen J, Shao L, Porikli F (2019) See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3623–3632
Lu X, Wang W, Shen J, Crandall D, Luo J (2020b) Zero-shot video object segmentation with co-attention siamese networks. In: Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence
Ma T, Latecki LJ (2012) Maximum weight cliques with mutex constraints for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 670–677
Maninis KK, Caelles S, Chen Y, Pont-Tuset J, Leal-Taixé L, Cremers D, Van Gool L (2018) Video object segmentation without temporal information. IEEE Trans Pattern Anal Mach Intell 41(6):1515–1530
Martin DR, Fowlkes CC, Malik J (2004) Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans Pattern Anal Mach Intell 26(5):530–549
Neuhold G, Ollmann T, Rota Bulo S, Kontschieder P (2017) The mapillary vistas dataset for semantic understanding of street scenes. In: Proceedings of the IEEE International Conference on Computer Vision, pp 4990–4999
Noh H, Hong S, Han B (2015) Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1520–1528
Ochs P, Malik J, Brox T (2013) Segmentation of moving objects by long term video analysis. IEEE Trans Pattern Anal Mach Intell 36(6):1187–1200
Ochs P, Brox T (2012) Higher order motion models and spectral clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 614–621
Oh SW, Lee JY, Sunkavalli K, Joo Kim S (2018) Fast video object segmentation by reference-guided mask propagation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7376–7385
Oh SW, Lee JY, Xu N, Kim SJ (2019) Video object segmentation using space-time memory networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp 9226–9235
Papazoglou A, Ferrari V (2013) Fast object segmentation in unconstrained video. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1777–1784
Parmar N, Vaswani A, Uszkoreit J, Kaiser L, Shazeer N, Ku A, Tran D (2018) Image transformer. In: Proceedings of the International Conference on Machine Learning, PMLR, pp 4055–4064
Perazzi F, Khoreva A, Benenson R, Schiele B, Sorkine-Hornung A (2017) Learning video object segmentation from static images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2663–2672
Perazzi F, Pont-Tuset J, McWilliams B, Van Gool L, Gross M, Sorkine-Hornung A (2016a) A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 724–732
Perazzi F, Pont-Tuset J, McWilliams B, Van Gool L, Gross M, Sorkine-Hornung A (2016b) A benchmark dataset and evaluation methodology for video object segmentation: Supplemental material. In: URL https://davischallenge.org/files/davis_supplementary.pdf
Pont-Tuset J, Perazzi F, Caelles S, Arbeláez P, Sorkine-Hornung A, Van Gool L (2017) The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:170400675
Prest A, Leistner C, Civera J, Schmid C, Ferrari V (2012) Learning object class detectors from weakly annotated video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 3282–3289
Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. In: Proceedings of the Advances in Neural Information Processing Systems, pp 91–99
Robinson A, Lawin FJ, Danelljan M, Khan FS, Felsberg M (2020) Learning fast and robust target models for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7406–7415

Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, pp 234–241

Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2015) Imagenet large scale visual recognition challenge. Int J Comput Vis 115(3):211–252

Seong H, Hyun J, Kim E (2020) Kernelized memory network for video object segmentation. In: Proceedings of the European Conference on Computer Vision, Springer, pp 629–645

Seong H, Oh SW, Lee JY, Lee S, Lee S, Kim E (2021) Hierarchical Memory Matching Network for Video Object Segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 12889–12898

Shi J, Yan Q, Xu L, Jia J (2015) Hierarchical image saliency detection on extended cssd. IEEE Trans Pattern Anal Mach Intell 38(4):717–729

Shi X, Chen Z, Wang H, Yeung DY, Wong WK, Woo Wc (2015b) Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Proceedings of the Advances in Neural Information Processing Systems, pp 802–810

Sikora T (1997) The mpeg-4 video standard verification model. IEEE Trans Circuits Syst Video Technol 7(1):19–31

Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: Proceedings of the International Conference on Learning Representations

Song H, Wang W, Zhao S, Shen J, Lam KM (2018) Pyramid dilated deeper convlstm for video salient object detection. In: Proceedings of the European Conference on Computer Vision, pp 715–731

Tjaden H, Schwanecke U, Schömer E, Cremers D (2018) A region-based gauss-newton approach to real-time monocular multiple object tracking. IEEE Trans Pattern Anal Mach Intell 41(8):1797–1812

Tokmakov P, Alahari K, Schmid C (2017a) Learning motion patterns in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3386–3394

Tokmakov P, Alahari K, Schmid C (2017b) Learning video object segmentation with visual memory. In: Proceedings of the IEEE International Conference on Computer Vision, pp 4481–4490

Tron R, Vidal R (2007) A benchmark for the comparison of 3-d motion segmentation algorithms. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1–8

Tsai YH, Yang MH, Black MJ (2016) Video segmentation via object flow. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3899–3908

Tsai D, Flagg M, Nakazawa A, Rehg JM (2012) Motion coherent tracking using multi-label mrf optimization. Int J Comput Vis 100(2):190–202

Ventura C, Bellver M, Girbau A, Salvador A, Marques F, Giro-i Nieto X (2019) Rvos: End-to-end recurrent network for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5277–5286

Voigtlaender P, Chai Y, Schroff F, Adam H, Leibe B, Chen LC (2019) Feelvos: Fast end-to-end embedding learning for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 9481–9490

Voigtlaender P, Leibe B (2017) Online adaptation of convolutional neural networks for video object segmentation. In: Proceedings of the British Machine Vision Conference

Wang W, Shen J, Porikli F (2017) Selective video object cutout. IEEE Trans Image Process 26(12):5645–5655

Wang W, Shen J, Porikli F, Yang R (2018) Semi-supervised video object segmentation with super-trajectories. IEEE Trans Pattern Anal Mach Intell 41(4):985–998

Wang H, Jiang X, Ren H, Hu Y, Bai S (2021a) Swiftnet: Real-time video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1296–1305

Wang W, Lu X, Shen J, Crandall DJ, Shao L (2019b) Zero-shot video object segmentation via attentive graph neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp 9236–9245

Wang L, Lu H, Wang Y, Feng M, Wang D, Yin B, Ruan X (2017a) Learning to detect salient objects with image-level supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 136–145

Wang W, Shen J, Lu X, Hoi SC, Ling H (2020) Paying attention to video object pattern understanding. In: Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence

Wang W, Shen J, Porikli F (2015) Saliency-aware geodesic video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3395–3402

Wang W, Song H, Zhao S, Shen J, Zhao S, Hoi SC, Ling H (2019c) Learning unsupervised video object segmentation through visual attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3064–3074

Wang Z, Xu J, Liu L, Zhu F, Shao L (2019d) Ranet: Ranking attention network for fast video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 3978–3987
Wang Y, Xu Z, Wang X, Shen C, Cheng B, Shen H, Xia H (2021c) End-to-end video instance segmentation with transformers. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8741–8750
Wang Q, Zhang L, Bertinetto L, Hu W, Torr PH (2019a) Fast online object tracking and segmentation: A unifying approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1328–1338
Wang W, Zhou T, Porikli F, Crandall D, Van Gool L (2021b) A survey on deep learning technique for video segmentation. arXiv preprint arXiv:210701153
Werbos PJ (1990) Backpropagation through time: what it does and how to do it. Proc IEEE 78(10):1550–1560
Wu Z, Shen C, Van Den Hengel A (2019) Wider or deeper: revisiting the resnet model for visual recognition. Pattern Recogn 90:119–133
Xiao H, Feng J, Lin G, Liu Y, Zhang M (2018) Monet: Deep motion exploitation for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1140–1148
Xie H, Yao H, Zhou S, Zhang S, Sun W (2021) Efficient regional memory network for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1286–1295
Xu Y, Wang Z, Li Z, Yuan Y, Yu G (2020) SiamFC++: towards robust and accurate visual tracking with target estimation guidelines. Proc AAAI Conf Artif Intell 34:12549–12556
Xu S, Liu D, Bao L, Liu W, Zhou P (2019c) Mhp-vos: Multiple hypotheses propagation for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 314–323
Xu K, Wen L, Li G, Bo L, Huang Q (2019a) Spatiotemporal cnn for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1379–1388
Xu N, Yang L, Fan Y, Huang TS, Yang J, Shi H (2019b) The 2nd large-scale video object segmentation challenge - track 1: Video object segmentation. In: URL https://competitions.codalab.org/competitions/20127#participate-get-data
Xu N, Yang L, Fan Y, Yang J, Yue D, Liang Y, Price B, Cohen S, Huang T (2018a) Youtube-vos: Sequence-to-sequence video object segmentation. In: Proceedings of the European Conference on Computer Vision, pp 585–601
Xu N, Yang L, Fan Y, Yue D, Liang Y, Yang J, Huang T (2018b) Youtube-vos: A large-scale video object segmentation benchmark. arXiv preprint arXiv:180903327
Yang L, Fan Y, Xu N (2019a) Video instance segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 5188–5197
Yang Z, Wang Q, Bertinetto L, Hu W, Bai S, Torr PH (2019b) Anchor diffusion for unsupervised video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 931–940
Yang L, Wang Y, Xiong X, Yang J, Katsaggelos AK (2018) Efficient video object segmentation via network modulation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6499–6507
Yang Z, Wei Y, Yang Y (2020) Collaborative video object segmentation by foreground-background integration. In: Proceedings of the European Conference on Computer Vision, Springer, pp 332–348
Yang Z, Wei Y, Yang Y (2021a) Associating objects with transformers for video object segmentation. In: Proceedings of the Advances in Neural Information Processing Systems
Yang Z, Wei Y, Yang Y (2021b) Collaborative video object segmentation by multi-scale foreground-background integration. In: Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence
Yao R, Lin G, Xia S, Zhao J, Zhou Y (2020) Video object segmentation and tracking: a survey. ACM Trans Intell Syst Technol 11(4):1–47
Yilmaz A, Javed O, Shah M (2006) Object tracking: a survey. ACM Comput Surv 38(4):13
Yoon JS, Rameau F, Kim J, Lee S, Shin S, So Kweon I (2017) Pixel-level matching for video object segmentation using convolutional neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp 2167–2176
Yu F, Koltun V (2016) Multi-scale context aggregation by dilated convolutions. In: Proceedings of the International Conference on Learning Representations
Zeng X, Liao R, Gu L, Xiong Y, Fidler S, Urtasun R (2019a) Dmm-net: Differentiable mask-matching network for video object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp 3929–3938
Zeng Y, Zhang P, Zhang J, Lin Z, Lu H (2019b) Towards high-resolution salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7234–7243

Zhang D, Javed O, Shah M (2013) 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, pp 628–635

Zhang L, Lin Z, Zhang J, Lu H, He Y (2019) Fast video object segmentation via dynamic targeting network. In: Proceedings of the IEEE International Conference on Computer Vision, pp 5582–5591

Zhang Y, Wu Z, Peng H, Lin S (2020) A transductive approach for video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6949–6958

Zhong D, Chang SF (1999) An integrated approach for content-based video object segmentation and retrieval. IEEE Trans Circuits Syst Video Technol 9(8):1259–1268

Zhou D, Bousquet O, Lal TN, Weston J, Schölkopf B (2004) Learning with local and global consistency. In: Advances in Neural Information Processing Systems, pp 321–328

Zhou T, Li J, Li X, Shao L (2021) Target-Aware Object Discovery and Association for Unsupervised Video Multi-Object Segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6985–6994

Zivkovic Z, Van Der Heijden F (2006) Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recogn Lett 27(7):773–780

Authors and Affiliations

Mingqi Gao[1,2] · Feng Zheng[2] · James J. Q. Yu[2] · Caifeng Shan[3] · Guiguang Ding[4] · Jungong Han[1,5]

Mingqi Gao
mingqi.gao@warwick.ac.uk

Feng Zheng
zhengf@sustech.edu.cn

Guiguang Ding
dinggg@tsinghua.edu.cn

[1] WMG Data Science, University of Warwick, Coventry CV4 7AL, UK

[2] Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China

[3] College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China

[4] School of Software, Tsinghua University, Beijing 100084, China

[5] Department of Computer Science, Aberystwyth University, Aberystwyth SY23 3DB, UK