A Survey on Deep Learning Technique for Video Segmentation

Wenguan Wang, Tianfei Zhou, Fatih Porikli, *Fellow IEEE*, David Crandall, Luc Van Gool

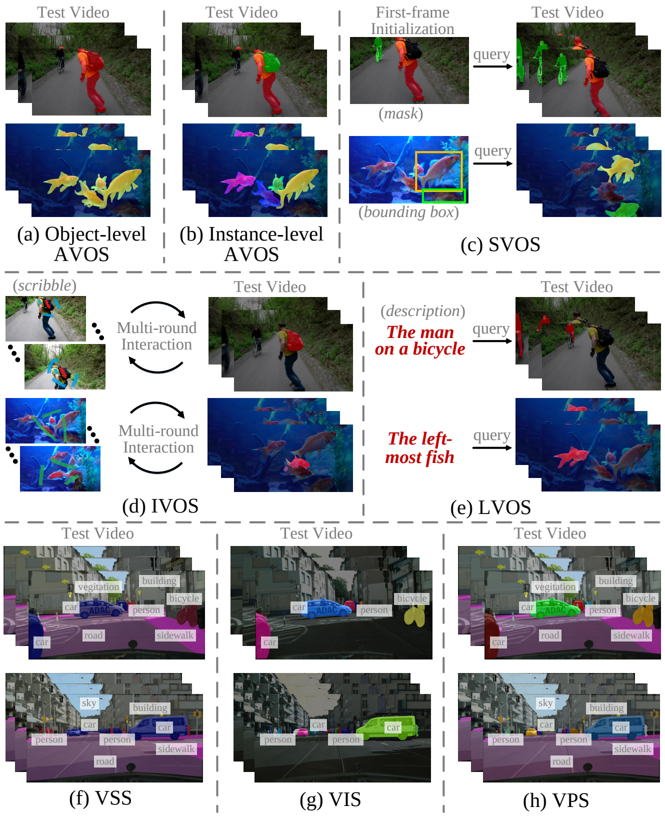
Abstract—Video segmentation, *i.e.*, partitioning video frames into multiple segments or objects, plays a critical role in a broad range of practical applications, *e.g.*, visual effect assistance in movie, scene understanding in autonomous driving, and virtual background creation in video conferencing, to name a few. Recently, due to the renaissance of connectionism in computer vision, there has been an influx of numerous deep learning based approaches that have been dedicated to video segmentation and delivered compelling performance. In this survey, we comprehensively review two basic lines of research in this area, *i.e.*, generic object segmentation (of unknown categories) in videos and video semantic segmentation, by introducing their respective task settings, background concepts, perceived need, development history, and main challenges. We also provide a detailed overview of representative literature on both methods and datasets. Additionally, we present quantitative performance comparisons of the reviewed methods on benchmark datasets. At last, we point out a set of unsolved open issues in this field, and suggest possible opportunities for further research.

Index Terms—Video Segmentation, Video Object Segmentation, Video Semantic Segmentation, Deep Learning

1 Introduction

Video segmentation, considered as the identification of the set of objects (with some specific feature properties or semantic value) building a video scene, is a fundamental and challenging problem in machine vision. Due to its crucial role in widespread application areas (*e.g.*, autonomous driving, robotics, automated surveillance, social media, augmented reality, movie industry, video conferencing, *etc.*), it has long garnered significant attention and been actively researched in computer vision and graphics communities.

Video segmentation has been addressed in the past using various traditional computer vision and machine learning techniques, including hand-craft features (*e.g.*, color, histogram statistics, optical flow, *etc.*), heuristic prior knowledge (*e.g.*, visual attention mechanism [1], motion boundary [2], *etc.*), low/mid-level visual representations (*e.g.*, super-voxel [3], trajectory [4], object proposal [5], *etc.*), and classical machine learning models (*e.g.*, clustering algorithms [6], graph models [7], random walk [8], support vector machine [9], random decision forest [10], markov random field [11], conditional random field [12], *etc.*). Recently, with the prosperity of deep neural networks and in particular the development of Fully Convolutional Network (FCN) [13], remarkable advances in video segmentation has been achieved. These deep learning based video segmentation algorithms surpassed other old-established approaches by a large margin in terms of accuracy and sometimes even efficiency, and continued to improve the state-of-the-art.



(a) Object-level AVOS (b) Instance-level AVOS (c) SVOS (d) IVOS (e) LVOS (f) VSS (g) VIS (h) VPS

Fig. 1. Video segmentation tasks reviewed in this survey: (a) object-level automatic video object segmentation (object-level AVOS), (b) instance-level automatic video object segmentation (instance-level AVOS), (c) semi-automatic video object segmentation (SVOS), (d) interactive video object segmentation (IVOS), (e) language-guided video object segmentation (LVOS), (f) video semantic segmentation (VSS), (g) video instance segmentation (VIS), and (h) video panoptic segmentation (VPS).

With the rapid advance of this field, there is a huge body of new literature being produced. However, existing surveys are mostly outdated (published before the modern deep learning era) [14], [15], and often with a narrow view, *i.e.*, focusing on fore-/back-ground video segmentation only [16], [17]. There is a lack of unifying works and state-of-the-art

- W. Wang, T. Zhou, and L. Van Gool are with ETH Zurich, Switzerland. (Email: {wenguan.wang, ztfei.debug}@gmail.com, vangool@vision.ee.ethz.ch)
- F. Porikli is with the School of Computer Science, Australian National University. (email: fatih.porikli@anu.edu.au)
- D. Crandall is with the Luddy School of Informatics, Computing, and Engineering, the Indiana University. (email: djcran@indiana.edu)

reviews. The ever-changing state of the field and fast pace of technique evolution cause initiation difficult. Hence, due to the diverse video segmentation settings and fuzzy notions, making a comprehensive and in-depth survey is incredibly hard and time-consuming, but highly needed and helpful.

In response, we provide the first review that systematically introduces recent advance on video segmentation, spanning from task formulation to taxonomy, from algorithms to datasets, from unsolved issues to future research directions. It covers several crucial aspects, including task categories (*i.e.*, fore-/back-ground separation *vs* semantic segmentation), processing modes (*i.e.*, automatic, semi-automatic, and interactive), and learning paradigms (*i.e.*, supervised, unsupervised, and weakly supervised), as well as clarifies confusing terminology (*e.g.*, background substraction, motion segmentation, *etc.*). We hope that this survey can provide a comprehensive review for researchers of interest, and foster research on the open issues raised.

1.1 Scope

This survey mainly focuses on recent progress in two major branches of video segmentation, namely video object segmentation (Fig. 1(a-e)) and video semantic segmentation (Fig. 1(f-h)), which can be further divided into eight subfields. Though we restrict our focus to deep learning based video segmentation solutions, there are still hundreds of works published in this rapidly growing field, making it impractical (and fortunately unnecessary) to review all of them. Instead, we select influential papers published in prestigious journals and conferences. Hence, some non-deep learning video segmentation models and relevant literature in other areas, such as video object detection and visual tracking, will be included to give necessary background.

1.2 Organization

The paper proceeds as follows. §2 gives some brief background on taxonomy, terminology, study history and related research areas. Representative works on deep learning algorithms and videos segmentation datasets are reviewed in §3 and §4, respectively. §5 conducts performance evaluation and analysis. Further, §6 points out a set of open questions and directions. Finally, conclusion remarks are given in §7.

2 Background

In §2.1, we formulate the task, categorize research directions, and numerate their key challenges and driven factors. §2.2 provides the reader with a brief historical background covering early works and foundations. §2.3 establishes linkages to the progress of other relevant fields.

2.1 Problem Formulation and Taxonomy

We next give a few words about problem setup and taxonomy in this survey. The categorizations of video segmentation models are also introduced from different viewpoints. Formally, let $\mathcal{X}$ and $\mathcal{Y}$ denote the input space and output segmentation space, respectively. Deep learning based video segmentation solutions generally seek to learn an *ideal* video-to-segment mapping $f^*: \mathcal{X} \mapsto \mathcal{Y}$.

2.1.1 Video Segmentation Category

According to the definitions of the output space $\mathcal{Y}$, video segmentation can be broadly categorized into two classes: video object (fore-/back-ground) segmentation and video semantic segmentation, which identify appropriate objects to satisfy the specific needs of different applications.

- **Video Fore-/Background Segmentation (Video Object Segmentation, VOS)**. VOS, as the most classic video segmentation setting, refers to the task of segmenting the dominant, general objects (of unknown categories) in video sequences. Thus $\mathcal{Y}$ is a binary, fore-/background segmentation space. VOS is typically used in some video analysis and editing related application scenarios, such as object cutting-off in movie editing, content-based video coding, and virtual background creation in video conferencing; it does not concern the exact semantic categories of the foreground.
- **Video Semantic Segmentation (VSS)**. As a direct extension of image semantic segmentation to the spatio-temporal domain, VSS aims to extract objects within predefined semantic categories (*e.g.*, car, building, road) from videos. Thus, $\mathcal{Y}$ corresponds to a multi-class, semantic parsing space. VSS serves as a perception foundation for many application fields, such as robot sensing, human-machine interaction, and autonomous driving, which require high-level understanding of the physical environment.

Remark. VOS and VSS do share some common challenges, such as fast motion, occlusion. However, due to their specific application scenarios, some challenges being concerned are different. For instance, VOS may pay more attention to the scenes with camera motion, large deformation, and appearance change, which typically appear in human created media. However, VSS instead pursues a good trade off between accuracy and latency, and focuses more on how to identify small objects, conduct model parallelization and strengthen cross-domain generalization ability, which is crucial for handling the data captured by vehicle cameras.

2.1.2 Processing Modes for Video Segmentation

VOS methods can be further classified into three types: automatic, semi-automatic, and interactive, according to how many human interventions are involved during inference.

- **Automatic Video Object Segmentation (AVOS)**. AVOS, or *unsupervised video segmentation* or *zero-shot video segmentation*, performs VOS in an automatic manner, without any manual initialization (Fig. 1(a-b)). The input space $\mathcal{X}$ refers to the video domain $\mathcal{V}$ only. AVOS is suitable for video analysis but not for video editing that requires to segment arbitrary objects or their parts flexibly; the typical application is virtual background creation in video conferencing.
- **Semi-automatic Video Object Segmentation (SVOS)**. SVOS, also known as *semi-supervised video segmentation* or *one-shot video segmentation* [18], involves limited human inspection (typically provided in the first frame) to determine the desired objects (Fig. 1(c)). SVOS addresses the limitation of AVOS techniques that lack flexibility in defining target objects, with the cost of additional human intervention. SVOS is typically applied in user-friendly setting (without specialized equipment), such as video content creation in mobile phones. Thus we have $\mathcal{X} = \mathcal{V} \times \mathcal{M}$, where $\mathcal{V}$ indicates the video space and $\mathcal{M}$ refers to human intervention. The

most typical form of human intervention considered in SVOS is first-frame object mask. In such case SVOS is also termed as *pixel-wise tracking* or *mask propagation*. Some other forms include bounding-box and scribble [8]. From this perspective, **language-guided video object segmentation (LVOS)** is a sub-branch of SVOS, where human intervention is given as linguistic descriptions about the desired objects, enabling efficient human-computer interaction (Fig. 1(e)). One of the core challenges in SVOS is how to fully utilize the target information from limited human intervention.

- **Interactive Video Object Segmentation (IVOS)**. SVOS models are designed to operate automatically once the target has been identified, while systems for IVOS incorporate user guidance throughout the analysis process (Fig. 1(d)). Thus IVOS can obtain high-quality segments and fit well for computer-generated imagery and video post-production, while requiring tedious human supervision. IVOS also draws attention from graphics community, termed as *video cutout*. The input space $\mathcal{X}$ for IVOS shall be $\mathcal{V}\times\mathcal{S}$, where $\mathcal{S}$ typically refers to human scribbling. Key challenges include: 1) allow users to easily specify segmentation constraints; 2) incorporate human specified constraints into the segmentation algorithm; 3) give quick response to the constraints.

VSS methods typically work in the automatic mode (Fig. 1(f-h)), *i.e.*, $\mathcal{X}\equiv\mathcal{V}$; only a few early methods address the semi-automatic setting, called as *label propagation* [19].

Remark. The terms "unsupervised" and "semi-supervised" are conventionally used in the field of VOS, to determine the amount of human interactions involved during inference. But they are easily confused with "unsupervised learning" and "semi-supervised learning". Thus we call on our community to replace these two fuzzy terms as "automatic" and "semi-automatic" resp., for unambiguous task definition.

2.1.3 *Learning Paradigms for Video Segmentation*

According to the learning strategy to approximate f^*, current deep learning based video segmentation models can be grouped into three categories, namely supervised, unsupervised, and weakly supervised learning based.

- **Supervised Learning based Methods.** Modern video segmentation models are typically learned in a fully supervised manner, requiring N input training samples and their desired outputs $y_n := f^*(x_n)$, where $\{(x_n, y_n)\}_n \subset\mathcal{X}\times\mathcal{Y}$. The standard way of evaluating learning outcomes follows an *empirical* risk/loss minimization formulation [20][1] :

$$\tilde{f} \in \mathop{\arg\min}_{f\in\mathcal{F}} \frac{1}{N}\sum\nolimits_{n} \varepsilon(f(x_n), z(x_n)), \tag{1}$$

where $\mathcal{F}$ denotes the hypothesis (solution) space, and $\varepsilon : \mathcal{X}\times\mathcal{Y}\mapsto\mathbb{R}$ is an error function that evaluates the estimate $f(x_n)$ against video segmentation related prior knowledge $z(x_n)\in\mathcal{Z}$. To make $\tilde{f}$ a good approximation of f^*, current supervised video segmentation methods directly use the desired output y_n, *i.e.*, $z(x_n){:=}f^*(x_n)$, as the prior knowledge, with the price of vast amounts of well-annotated data.

- **Unsupervised Learning based Methods.** When only samples $\{x_n\}_n\subset\mathcal{X}$ are given, the problem of approximating f^* is known as unsupervised learning. The prior knowledge $\mathcal{Z}$, in such setting, is typically built upon certain intrinsic properties of video data (*e.g.*, cross-frame consistency).

- **Weakly-Supervised Learning based Methods.** In this case, $\mathcal{Z}$ is typically a more easily annotated domain, such as tag/bounding box/scribble-level supervision. Then f^* is approximated using a finite number of samples from $\mathcal{X}\times\mathcal{Z}$.

Remark. So far, deep supervised learning based methods are dominant in the filed of video segmentation. However, exploring the task in an unsupervised or weakly supervised setting is more appealing, not only because it alleviates the annotation burden of $\mathcal{Y}$, but also because it inspires an in-depth understanding of the nature of task by exploring $\mathcal{Z}$.

2.2 History and Terminology

The history of segmentation of digital images using computers could be traced back 50 years [21]. In 1965, Roberts designed an operator for edge detection [21], representing the first step toward decomposing an image into its constitutional components. Since then, a large number of algorithms for image segmentation have been proposed, and naturally extended to the video domain. The field of video segmentation has evolved very quickly and undergone great change.

Earlier attempts focus on **video over-segmentation**, *i.e.*, partitioning a video into space-time homogeneous, perceptually distinct regions. Typical ones include hierarchical video segmentation [7], temporal superpixel [22], and supervoxels [3], based on the discontinuity and similarity of pixel intensities in a particular location, *i.e.*, separating pixels according to abrupt changes in intensity or grouping pixels with similar intensity together. They are instructive for early stage video preprocessing, but cannot solve the problem of object-level pattern modeling, as they do not provide any principled approach to flatten the hierarchical video decomposition into a binary segmentation [2], [23].

To extract foreground objects from video sequences, **background substraction** techniques emerged since the late 70s [24], and became popular following the work of [25]. They assume that the background is known a priori, and that the camera is stationary [26], [27] or undergoes a predictable, parametric 2D [28] or 3D motion with 3D parallax[29], [30]. These geometry-based methods fit well for specific application scenarios such as surveillance systems [23], [27], but they are sensitive to model selection (2D or 3D), and cannot handle non-rigid camera movements. Another group of video segmentation solutions is introduced to tackle the task of **motion segmentation**, *i.e.*, finding objects in motion. Background substraction can also be viewed as a specific case of motion segmentation. However, most motion segmentation models are built upon motion analysis [31], [32], factorization [33], and/or statistical [34] techniques, that comprehensively model the characteristics of moving scenes without prior knowledge of camera motion. Among the big family of motion segmentation algorithms, **trajectory segmentation** attained particular attention [4], [35]–[38]. Trajectories are generated through tracking points over multiple frames and have the advantage of representing long-term motion patterns, serving as an informative cue for segmentation. Though impressive, motion based methods heavily rely on the accuracy of optical flow estimation and suffer difficulties when different parts of an object exhibit

[1] We omit the regularization term for brevity.

heterogeneous motions. To overcome these limitations, the task of extracting generic objects from unconstrained video sequences, *i.e.*, AVOS, drew more and more research interest [39]. Several methods [5], [40]–[42] explored object hypotheses, *i.e.*, object proposal [43], as middle-level object representations. They generated a large number of object candidates in every frame and cast the task of segmenting video objects as an object region selection problem. The main drawbacks of the proposal based algorithms are their high computational cost [17] associated with proposal generation and complicated object inference schemes. Some others explored certain heuristic hypotheses, such as visual attention [1] and motion boundary [2], but they easily fail in the scenarios where the heuristic assumptions do not hold.

As argued earlier, an alternative to above unattended solutions is incorporating human marked initialization, *i.e.*, SVOS. Older SVOS methods often rely on optical flow [44]–[47] and share a similar spirit with *object tracking* [48], [49]. In addition, some pioneer IVOS methods were proposed to address high-quality video segmentation under extensive human guidance, including rotoscoping [50], [51], scribble [8], [52]–[55], contour [56] and point [57]. Significant engineering is typically needed to allow IVOS systems operate at interactive speeds. In short, SVOS and IVOS pay for the improved flexibility and accuracy: they are unfeasible at large scale, due to the human-in-the-loop nature.

In the pre deep learning era, much fewer efforts [12], [58], [59] were made towards tackling VSS, due to greater complexity of the problem. They typically relied on supervised classifiers such as SVM and video over-segmentation techniques. There were also some weakly supervised solutions that learn VSS from tagged videos [60]–[62].

Overall, traditional approaches for video segmentation, though giving interesting results, are constrained by hand-crafted features and heavy engineering. Only until recently, the great advance in deep learning brought the performance of video segmentation to a new level. Representative deep learning based solutions will be reviewed in §3.

2.3 Related Research Area

There are several research fields closely related to video segmentation, which are briefly described in the following.

• **Visual Tracking.** To infer the location of a target object over time, current tracking methods usually consider the situation where the target is determined by a bounding box in the first frame [63]. However, in more general tracking scenario, in particular the cases studied in early tracking methods, diverse object representations are explored [64], including centroid, skeleton, and contour. Some video segmentation techniques, such as background substraction, are also merged into older trackers [65], [66]. Hence, visual tracking and video segmentation encounter some common challenges (*e.g.*, object/camera motion, appearance change, occlusion, *etc.*), fostering their mutual collaboration.

• **Image Semantic Segmentation.** The success of end-to-end image semantic segmentation [67] stimulates the rapid development of VSS. Rather than directly applying image semantic segmentation techniques frame by frame, recent VSS systems explore temporal continuity to increase both the accuracy and efficiency. Still, image semantic segmentation techniques will continue to serve as the foundation for the advance of its counterpart in the video domain.

• **Video Object Detection.** To generalize object detection in the video domain [68], video object detectors incorporate temporal cues over the box- or feature- level. There are many key technical steps/challenges, such as object proposal generation, temporal information aggregation, and cross-frame object association, that are shared between video object detection and (instance-level) video segmentation.

• **Video Salient Object Detection.** As a crucial ability of human visual system, visual attention mechanism allows human to quickly allocate attention to important parts of visual scenes. To interpret such cognitive behavior, video salient object detection algorithms output a probability map per frame – highlighting most attention-grabbing objects in dynamic scenes [69]. In the pre deep learning era, video salient object detection draw more inspirations from classic visual attention theories in cognitive science, while AVOS is mainly driven by the practical requirements of video analysis. Thus dynamic saliency was often explored as a biologically-inspired feature in early AVOS systems. But in the deep learning era, the difference between the two fields become marginal. However, based on human eye fixation records, some recent studies [70], [71] suggested to reconsider the notion of "primary objects" in AVOS and emphasize stimulus-driven attention-shifts (*i.e.*, visually salient object(s) may dynamically change), demonstrating the two fields are both closely relative and certainly independent.

3 Deep Learning based Video Segmentation

This section provides a detailed overview of representative, deep learning algorithms in the two sub-areas of video segmentation (§2.1.1), namely video object segmentation (VOS, §3.1) and video semantic segmentation (VSS, §3.2).

3.1 Deep Learning based VOS Models

VOS aims to extract generic, foreground objects from video sequences with no concern of semantic category recognition. Based on how much human intervention involved in inference, VOS models can be divided into three classes (§2.1.2): automatic VOS (AVOS, §3.1.1), semi-automatic VOS (SVOS, §3.1.2), and interactive VOS (IVOS, §3.1.3). Moreover, although language-guided video object segmentation (LVOS) falls in the broader category of SVOS, LVOS methods are reviewed alone (§3.1.4), due to the multi-modal task setup.

3.1.1 Automatic Video Object Segmentation (AVOS)

Instead of using heuristic priors and hand-crafted features to automatically execute VOS, modern AVOS methods learn video generic object patterns in a data-driven fashion. Next, landmark efforts are grouped based on their key techniques.

• **Deep Learning Module based Methods.** In 2015, Fragkiadaki *et al.* [114] made an early effort that learns a multi-layer perceptron to rank proposal segments and infer foreground objects. In 2016, Tsai *et al.* [115] proposed a joint optimization framework for AVOS and optical flow estimation with a naïve use of deep features from a pre-trained classification network. Later methods [76], [78] learn FCNs to predict initial, pixel-level foreground estimates from frame

TABLE 1
Summary of essential characteristics for reviewed AVOS methods. Instance: instance- or object-level segmentation; **Flow**: whether use optical flow; **E2E**: whether end-to-end trainable. See §3.1.1 for more detailed descriptions.

Year	Method	Pub.	Core Architecture	Backbone	Instance	Flow	E2E	Training Dataset
2017	FSEG [72]	CVPR	Two-Stream FCN	ResNet101	Object	✓	✓	ImageNet VID [73] + DAVIS$_{16}$ [17]
	SFL [74]	ICCV	Two-Stream FCN	ResNet101	Object	✓	✓	DAVIS$_{16}$ [17]
	LVO [75]	ICCV	Two-Stream FCN	VGGNet	Object	✓	✓	DAVIS$_{16}$ [17]
	LMP [76]	ICCV	FCN	VGGNet	Object	✓	✓	FT3D [77]
	Li *et al.* [78]	ICCV	FCN	VGGNet	Object	✓	✓	Youtube-Objects [79]
2018	Li *et al.* [80]	CVPR	FCN	ResNet101	Object	✓		DAVIS$_{16}$ [17]
	FGRNE [81]	CVPR	FCN+RNN	VGGNet	Object		✓	SegTrackV2 [82]+DAVIS$_{16}$ [17]+FBMS [38]
	Li *et al.* [83]	ECCV	FCN	ResNet101	Object	✓		DAVIS$_{16}$ [17]
	PDB [84]	ECCV	RNN	ResNet101	Object		✓	DAVIS$_{16}$ [17]
	MOT [85]	ICRA	Two-Stream FCN	ResNet101	Object	✓	✓	DAVIS$_{16}$ [17]
2019	RVOS [86]	CVPR	RNN	ResNet101	Instance		✓	DAVIS$_{17}$ [87]/YouTube-VIS [88]
	COSNet [89]	CVPR	Siamese FCN + Co-attention	ResNet101	Object		✓	MSRA10K [90]+DUT [91]+DAVIS$_{16}$ [17]
	UMOD [92]	CVPR	Adversarial Network	VGGNet	Object	✓		SegTrackV2 [82]+DAVIS$_{16}$ [17]+FBMS [38]
	AGS [93]	CVPR	FCN	ResNet101	Object		✓	SegTrackV2 [82]+DAVIS$_{16}$ [17]+DUT [91] + PASCAL-S [94]
	AGNN [95]	ICCV	FCN + GNN	ResNet101	Object		✓	MSRA10K [90]+DUT [91]+DAVIS$_{16}$ [17]
	MGA [96]	ICCV	Two-Stream FCN	ResNet101	Object	✓	✓	DUTS [97]+DAVIS$_{16}$ [17]+FBMS [38]
	AnDiff [98]	ICCV	Siamese FCN + Co-attention	ResNet101	Object		✓	DAVIS$_{16}$ [17]
	LSMO [99]	IJCV	Two-Stream FCN	VGGNet	Object	✓	✓	FT3D [77]+DAVIS$_{16}$ [17]
2020	MATNet [100]	AAAI	Two-Stream FCN	ResNet101	Object	✓	✓	Youtube-VOS [101]+DAVIS$_{16}$ [17]
	PyramidCSA [102]	AAAI	Siamese FCN + Co-attention	MobileNetV3	Object		✓	DUTS [97]+DAVIS$_{16}$ [17]+DAVSOD [70]
	MuG [103]	CVPR	FCN	ResNet50	Object		✓	OxUvA [104]
	EGMN [105]	ECCV	FCN + Episodic Memory	ResNet50	Object		✓	MSRA10K [90]+DUT [91]+DAVIS$_{16}$ [17]
	WCSNet [106]	ECCV	Siamese FCN	EfficientNetV2	Object		✓	SALICON [107]+PASCAL VOC 2012 [108] + DUTS [97] + DAVIS$_{16}$ [17]
	DFNet [109]	ECCV	Siamese FCN	ResNet101	Object		✓	MSRA10K [90] + DUT [91] + DAVIS$_{16}$ [17]
2021	F2Net [110]	AAAI	Siamese FCN	ResNet101	Object		✓	MSRA10K [90]+DAVIS$_{16}$ [17]
	Zhou *et al.* [111]	CVPR	Siamese FCN	ResNet101	Instance			DAVIS$_{17}$ [87]/YouTube-VIS [88]
	RTNet [112]	CVPR	Two-Stream FCN	ResNet101	Object	✓	✓	DUTS [97]+DAVIS$_{16}$ [17]
	DyStab [113]	CVPR	Adversarial Network	ResNet101	Object	✓	✓	SegTrackV2 [82]+DAVIS$_{16}$ [17]+FBMS [38]

images [69], [78] or optical flow fields [76], while several post-processing steps are still needed. Basically, these primitive solutions largely rely on traditional AVOS techniques; the learning ability of neural networks is under-explored.

• **Pixel Instance Embedding based Methods.** A group of AVOS models is developed to make use of stronger deep learning descriptors [80], [83] – instance embeddings – learned from image instance segmentation data [116]. They first generate pixel-wise instance embeddings; the similarity between the embeddings indicates whether the pixels belong to a same object instance. Then, some representative embeddings are selected and clustered into foreground and background. The labels from these sampled embeddings are finally propagated to the rest ones. The clustering and label propagation can be achieved without video specific supervision. Though using fewer annotations, these methods suffer from a fragmented and sophisticated pipeline.

• **End-to-end Methods with Short-term Information Encoding.** End-to-end model designs became the mainstream in this field. For example, convolutional recurrent neural networks (RNNs) were used to learn spatial and temporal visual patterns jointly [84], [95]. Another big family is built upon two-stream networks [72], [74], [75], [81], [96], [99], [100], wherein two parallel streams are built to extract features from raw image and optical flow, which are further fused for segmentation prediction. Two-stream methods make explicit use of appearance and motion cues, with the cost of optical flow computation and heavy weights. These end-to-end methods achieve improvements in accuracy and show well the advantages of applying neural networks to this task, while considering local content within very limited time span. They only stack appearance and/or motion information from a few successive frames as input, ignoring relations among distant frames. Though RNNs are usually adopted, the internal hidden memory brings the inherent limitation in modeling long-term data dependency [117].

• **End-to-end Methods with Long-term Context Encoding.** Current top-leading AVOS models were built upon the usage of global context over long time span. In a seminal work [89], Lu *et al*. proposed a Siamese architecture based AVOS model, which extracts features for arbitrary frame pairs and captures cross-frame context through calculating pixel-wise feature correlations. During inference, for each test frame, context from several other frames (within the same video) can be aggregated to locate the primary objects. A contemporary work [98] exploited a similar idea but solely considered the first frame as reference for other test frames. Since put forward, [89] has led to several follow-up works [106], [112], which mainly focus on making more comprehensive use of information from multiple frames [173], [174], encoding spatial context [110], or incorporating temporal consistency to improve representation power and computation efficiency [102], [109].

• **Un-/Weakly-Supervised Learning based AVOS Methods.** Only a handful of methods learn to perform AVOS from unlabeled or weakly labeled data. In [93], static image salient object segmentation data and dynamic eye fixation data, which are more easily acquired compared with video segmentation data, are used to learn video generic object patterns. Hence, in [103], visual patterns are learned through exploring several intrinsic properties of video data at multiple granularities, *i.e.*, intra-frame saliency, short-term visual coherence, long-range semantic correspondence, and video-level discriminativeness. In [92], an adversarial contextual model is developed to segment moving objects without any manual annotation, achieved by minimizing the mutual information between the motions of an object and its context. This method is further enhanced in [113], by adopting a bootstrapping strategy and enforcing temporal consistency.

• **Instance-Level AVOS Methods.** Instance-level AVOS, also referred as *multi-object unsupervised video segmentation*, was introduced with the launch of DAVIS$_{19}$ challenge [175]. This task setting is more challenging as it requires to not only separate the foreground objects from the background, but also discriminate different object instances. To tackle this task, current solutions typically work in a two-down

TABLE 2
Summary of essential characteristics for reviewed SVOS methods. Flow: whether use optical flow; **E2E**: whether end-to-end trainable. See §3.1.2 for more detailed descriptions.

Year	Method	Pub.	Core Architecture	Backbone	Flow	E2E	Technical Feature	Training Dataset
2017	OSVOS [18]	CVPR	FCN	VGGNet			Online Fine-tuning	DAVIS$_{16}$ [17]
	MaskTrack [118]	CVPR	FCN	VGGNet	✓		Propagation-based	ECSSD [119]+MSRA10K [90]+PASCAL-S [94]+DAVIS$_{16}$ [17]
	CTN [120]	CVPR	FCN	VGGNet	✓		Propagation-based	PASCAL VOC 2012 [108]
	VPN [121]	CVPR	Bilateral Network	VGGNet			Propagation-based	DAVIS$_{16}$ [17]
	PLM [122]	CVPR	Siamese FCN	VGGNet			Matching-based	DAVIS$_{16}$ [17]
	OnAVOS [123]	BMVC	FCN	ResNet38			Online Fine-tuning	PASCAL VOC 2012 [108] + COCO [124] + DAVIS [17]
	Lucid [125]	IJCV	Two-Stream FCN	VGGNet	✓		Propagation-based	DAVIS$_{16}$ [17]
2018	CINM [126]	CVPR	Spatio-temporal MRF	VGGNet	✓		Propagation-based	DAVIS$_{17}$ [87]
	FAVOS [127]	CVPR	FCN	ResNet101			Propagation-based	DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	RGMP [128]	CVPR	Siamese FCN	ResNet101			Propagation-based	PASCAL VOC 2012 [108] + ECSSD [119] + MSRA10K [90] + DAVIS$_{17}$ [87]
	OSMN [129]	CVPR	FCN + Meta Learning	VGGNet			Online Fine-tuning	ImageNet VID [73] + DAVIS$_{16}$ [17]
	MONet [130]	CVPR	FCN	VGGNet	✓		Online Fine-tuning	PASCAL VOC 2012 [108] + DAVIS$_{16}$ [17]
	CRN [131]	CVPR	FCN + Active Contour	ResNet101	✓		Propagation-based	PASCAL VOC 2012 [108] + DAVIS$_{16}$ [17]
	RCAL [132]	CVPR	FCN + RL	DenseNet56			Propagation-based	MSRA10K [90] + PASCAL-S + SOD +ECSSD [119] + DAVIS$_{16}$ [17]
	OSVOS-S [133]	PAMI	FCN	VGGNet			Online Fine-tuning	DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	Videomatch [134]	ECCV	Siamese FCN	ResNet101		✓	Matching-based	DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	Dyenet [135]	ECCV	Re-ID	ResNet101			Propagation-based	DAVIS$_{17}$ [87]
	LSE [136]	ECCV	FCN	ResNet101			Propagation-based	PASCAL VOC 2012 [108]
	Vondrick *et al.* [137]	ECCV	Siamese FCN	ResNet18			Unsupervised Learning	Kinetics [138]
2019	MVOS [139]	PAMI	Siamese FCN + Meta Learning	ResNet101			Online Fine-tuning	PASCAL VOC 2012 [108] + DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	FEELVOS [140]	CVPR	FCN	Xception-65		✓	Matching-based	COCO [124] + DAVIS$_{17}$ [87]+YouTube-VOS [101]
	MHP-VOS [141]	CVPR	Graph Optimization	ResNet-101			Propagation-based	COCO [124] + DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	AGSS [142]	CVPR	FCN	ResNet-101	✓	✓	Propagation-based	DAVIS$_{17}$ [87]/YouTube-VOS [101]
	AGAME [143]	CVPR	FCN	ResNet-101		✓	Propagation-based	MSRA10K [90]+PASCAL VOC 2012 [108]+DAVIS$_{17}$ [87]/YouTube-VOS [101]
	SiamMask [144]	CVPR	Siamese FCN	ResNet-50			Box-Initialization	DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]/YouTube-VOS [101]
	RVOS [86]	CVPR	RNN	ResNet101		✓	Propagation-based	DAVIS$_{17}$ [87]/YouTube-VIS [88]
	BubbleNet [145]	CVPR	-	ResNet50			-	DAVIS$_{17}$ [87]
	RANet [146]	ICCV	Siamese FCN	ResNet101			Matching-based	MSRA10K [90]+ECSSD [119]+HKU-IS [147]+DAVIS$_{16}$ [17]/DAVIS$_{17}$ [87]
	DMM-Net [148]	ICCV	Mask R-CNN	ResNet50		✓	Differentiable Matching	DAVIS$_{17}$ [87]/YouTube-VOS [101]
	DTN [149]	ICCV	FCN	ResNet50	✓		Propagation-based	COCO [124] + PASCAL VOC 2012 [108] + DAVIS$_{16}$/DAVIS$_{17}$ [87]
	STM [150]	ICCV	Memory Network	ResNet50			Matching-based	PASCAL VOC 2012 [108] + COCO [124] + ECSSD [119] + DAVIS$_{17}$ [87]/YouTube-VOS [101]
	TimeCycle [93]	ECCV	Siamese FCN	ResNet50			Unsupervised Learning	VLOG [151]
2020	e-OSVOS [152]	NeurIPS	Mask R-CNN + Meta Learning	ResNet101			Online Fine-tuning	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	Liang *et al.* [153]	NeurIPS	Memory Network	ResNet50			Matching-based	PASCAL VOC 2012 [108] + COCO [124] + ECSSD [119] + DAVIS$_{17}$ [87]/YouTube-VOS [101]
	Fasttan [154]	CVPR	Faster R-CNN	ResNet50 + I3D		✓	Propagation-based	COCO [124] + DAVIS$_{17}$ [87]
	Fasttmu [155]	CVPR	FCN + RL	ResNet101			Box-Initialization	PASCAL VOC 2012 [108] + DAVIS$_{17}$ [87]
	SAT [156]	CVPR	FCN + RL	ResNet50			Propagation-based	COCO [124] + DAVIS$_{17}$ [87] + YouTube-VOS [101]
	FRTM-VOS [157]	CVPR	FCN	ResNet101			Matching-based	DAVIS$_{17}$ [87]/YouTube-VOS [101]
	TVOS [158]	CVPR	FCN	ResNet50		✓	Matching-based	DAVIS$_{17}$ [87]/YouTube-VOS [101]
	MuG [103]	CVPR	Siamese FCN	ResNet50			Unsupervised Learning	OxUvA [104]
	MAST [159]	CVPR	Memory Network	ResNet18			Unsupervised Learning	OxUvA [104]+YouTube-VOS [101]
	GCNet [160]	ECCV	Memory Network	ResNet50			Matching-based	MSRA10K [90] + ECSSD [119]+HKU-IS [147]+ DAVIS$_{17}$ [87]/YouTube-VOS [101]
	KMN [161]	ECCV	Memory Network	ResNet50			Matching-based	PASCAL VOC 2012 [108] + COCO [124] + ECSSD [119] + DAVIS$_{17}$ [87]/YouTube-VOS [101]
	CFBI [162]	ECCV	FCN	ResNet101			Matching-based	COCO [124] + DAVIS$_{17}$ [87]/YouTube-VOS [101]
	LWL [163]	ECCV	Siamese FCN + Meta Learning	ResNet50		✓	Matching-based	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	MSN [164]	ECCV	Memory Network	ResNet50			Matching-based	DAVIS$_{17}$ [87]/YouTube-VOS [101]
	EGMN [105]	ECCV	Memory Network	ResNet50			Matching-based	MSRA10K [90]+COCO [124] +DAVIS$_{17}$ [87]+YouTube-VOS [101]
	STM-Cycle [165]	NeurIPS	Memory Network	ResNet50			Matching-based	DAVIS$_{17}$ [87]+YouTube-VOS [101]
2021	QMA [166]	AAAI	Memory Network	ResNet50			Box-Initialization	DUT [91] + HKU-IS [147] + MSRA10K [90] + YouTube-VOS [101]
	SwiftNet [167]	CVPR	Memory Network	ResNet50			Matching-based	COCO [124]+DAVIS$_{17}$ [87]/YouTube-VOS [101]
	G-FRTM [168]	CVPR	FCN + RL	ResNet101			Matching-based	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	SST [169]	CVPR	Transformer	ResNet101		✓	Matching-based	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	Ge *et al.* [170]	CVPR	Siamese FCN	ResNet101		✓	Matching-based	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	LCM [171]	CVPR	Memory Network	ResNet50			Matching-based	PASCAL VOC 2012 [108] + COCO [124] + ECSSD [119] + DAVIS$_{17}$ [87]/YouTube-VOS [101]
	RMNet [172]	CVPR	Memory Network	ResNet50	✓		Matching-based	PASCAL VOC 2012 [108] + COCO [124] + ECSSD [119] + DAVIS$_{17}$ [87]/YouTube-VOS [101]

fashion, *i.e.*, generating object candidates for each frames, and associating instances over different frames. In the early attempt [86], Ventura *et al.* delivered a recurrent network based model that consists of a spatial LSTM for per-frame instance discovering and a temporal LSTM for cross-frame instance association. This method enjoys the elegant model design, while its representation ability is too weak to numerate all the object instances and to capture the complex interactions between instances over the temporal domain. Thus later methods [71], [174], [176] strengthen the two-step pipeline through: **i)** employing image instance segmentation models (*e.g.*, Mask R-CNN [177]) to detect an exhaustive set of object candidates, and **ii)** leveraging object tracking or re-identification techniques for instance association, supplemented by some manually defined rules. Fore-/background AVOS techniques [89], [173] are also used to filter out nonsalient candidates [71], [174]. Hence, most recent methods [111] decompose the task into proposal generation and SVOS: object candidates are first generated, and corresponding tracklets are then obtained using advanced SVOS techniques. Overall, current instance-level AVOS models follow the classic tracking-by-detection paradigm, involving several ad-hoc designs. There still remains considerable room for further improvement in accuracy and efficiency.

3.1.2 Semi-automatic Video Object Segmentation (SVOS)

Deep learning based SVOS methods mainly focus on the first-frame *mask* propagation setting. They are categorized by their utilization of the test-time provided object masks.

• **Online Fine-tuning based Methods.** Following the one-shot principle, this family of methods [18], [130], [133], [139] online trains a segmentation model separately on each given object mask. Fine-tuning methods essentially exploit the transfer learning capabilities of neural networks and often follow a two-step training procedure: i) *offline pre-training*: learn general segmentation features from images and video sequences, and ii) *online fine-tuning*: learn target-specific representations from test-time supervision. The idea of fine-tuning was first introduced in [18], where only the initial image-mask pair is used for one-shot, online training a merely appearance-based FCN model. Then, in [123], more pixel samples in the unlabeled frames are mined as online training samples for better adapting to further changes over time. As [18], [123] have no notion of individual objects, [133] further incorporates instance segmentation models (*e.g.*, Mask R-CNN [177]) during inference. While elegant through their simplicity, fine-tuning methods face several shortcomings: i) pre-training is fixed and not optimized for the subsequent fine-tuning, ii) the hyperparameters of online fine-tuning are often excessively hand-crafted and fail to generalize between test cases, iii) the common existing fine-tuning setups suffer from high test runtimes (up to 1000 training iterations per segmented object online [18]). The root cause is that, these approaches choose to encode all the target-related cues (*i.e.*, appearance, mask) into network parameters implicitly. Towards efficient and automated fine-tuning, some recent methods [129], [139], [152] turn to meta learning techniques, *i.e.*, optimize the fine-tuning policies [139], [152] (*e.g.*, generic model initialization, learning rates, *etc.*) or even directly modify network weights [129].

• **Propagation based Methods.** Two lines of recent research efforts, built upon mask propagation and template matching techniques respectively, are devoted to refraining from the online optimization and delivering compact, end-to-end SVOS solutions. In particular, propagation methods use the previous frame mask to infer the current frame mask. With the emergence of fine-tuning methods [18], several propagation methods were proposed [118], [120], [121]. For example, Jampani *et al.* [121] propose a bilateral network for long-range video-adaptive mask propagation. Perazzi *et al.* [118] approach SVOS by employing a modified FCN, where the previous frame mask is considered as an additional channel besides the input frame. This approach is further reinforced in the follow-up methods through conducting optical flow guided mask alignment [130], performing heavy first-frame data augmentation [125], or making multi-step segmentation refinement [131]. Some others use a re-identification module to retrieve missing objects after prolonged occlusions [135], design a reinforcement learning agent to conduct segmentation in a conditional decision-making process [132], or embed mask propagation into the inference of a spatiotemporal MRF model to improve temporal coherency [126]. Some researcher propose to learn location-aware embeddings to improve the discriminative ability of features [136], or directly learn the mask propagation through a sequence-to-sequence model [101]. In addition, advanced tracking techniques are also exploited in several propagation methods [127], [141], [154], [156]. Propagation methods are found to easily suffer from error accumulation due to drifts and occlusions during mask propagation. Conditioning propagation on the initial frame-mask pair [128], [142], [149] seems a feasible solution to this. Although target-specific mask is explicitly encoded into the segmentation network, which makes up for the deficiencies of fine-tuning methods to a certain extent, propagation methods still learn object appearance by hidden network weights. Such implicit target-appearance modeling strategy hurts the flexibility and adaptivity. The only exception may be [143], which learns a generative model of target and background feature distributions to facilitate mask propagation.

• **Matching based Methods.** Matching based methods might be the most promising solution. They first construct an embedding space to memorize the initial object embeddings, then classify each pixel's label according to their similarities to the target object in the embedding space. Thus the initial object appearance is explicitly modeled, and no fine-tuning is necessary during inference. The earliest effort in this direction can be tracked back to [122]. Inspired by the advance in visual tracking [187], Yoon *et al.* [122] proposed a Siamese network to perform pixel-level matching between the first frame and upcoming frames. Later, [127] proposed to learn an embedding space from the first-frame supervision and pose VOS as a task of pixel retrieval: pixels are simply their respective nearest neighbors in the learned embedding space. The idea of [122] is also explored in [134], while it computes two matching maps for each upcoming frame, with respect to the foreground and background annotated in the first frame. In [140], pixel-level similarities are computed from the first frame and from the previous frame are both used as a guide to segment succeeding frames. Later, many matching based solutions were proposed [146], [150], [188], among which [150] significantly advances the development of matching based solutions. In [150], Oh *et al*. propose a space-time memory (STM) model to explicitly store previously computed segmentation information in an external memory, which facilitates learning the evolution of objects over time and allows for comprehensive use of past segmentation clues even over long period of time. Almost all current top-leading SVOS solutions [158], [162] are built upon STM architecture; they improve the target adaption ability [105], [157], [163], incorporate local temporal continuity [161], [171], [172], explore instance-aware cues [170], or develop more efficient memory designs [153], [160], [164], [167]. Recently, [169] introduced a Transformer [189] based model, which performs matching-like computation through attending over a history of multiple frames. In general, matching based solutions enjoy the advantage of flexible and differentiable model design as well as long-term correspondence modeling. On the other hand, feature matching relies on a powerful and generic feature embedding, which may limit its performance in challenging scenarios.

It is also worth mentioning that, as an effective technique for target-specific model learning, online learning is applied by many propagation [101], [118], [126], [131], [141] and matching [122], [135], [146] methods to boost performance.

• **Box-Initialization based Methods.** As pixel-wise object annotations are time-consuming or even impractical to acquire in realistic scenes, some efforts were made to consider the situation where the first-frame annotation is provided in the form of bounding box. In [144], Siamese trackers are augmented with a mask prediction branch. In [155], reinforcement learning is introduced to make decisions for target updating and matching. Later, in [166], an outside memory is utilized to build a stronger Siamese track-segmenter.

• **Un-/Weakly-Supervised Learning based Methods.** To alleviate the demand for large-scale, pixel-wise annotated training samples, several un-/weakly-supervised learning based SVOS solutions were recently developed. They typically utilize self-supervised learning techniques that impose the learned features to capture certain constraints on local coherence, such as cross-frame color consistency [137] and temporal cycle-correspondence [93]. The method proposed in [103] is both applicable to AVOS and SVOS settings. In [159], the memory-augmented network architecture is also used to better leverage the long-range correspondence among distant frames as the self-supervision signal.

• **Other Specific Methods.** There are also some efforts made specific contributions that deserve a separate look. In [148], Zeng *et al*. extract mask proposals per frame and formulate the matching between object templates and proposals in a *differentiable* manner. Instead of only using the first frame annotation, [145] learns to select *the best frame* from

TABLE 3
Summary of essential characteristics for reviewed IVOS methods. See §3.1.3 for more detailed descriptions.

Year	Method	Pub.	Core Architecture	Backbone	Technical Feature	Training Dataset
2017	Benard *et al.* [178]	-	FCN	ResNet101	Interaction-Propagation	PASCAL VOC 2012 [108]
2018	BFVOS [179]	CVPR	FCN	ResNet101	Pixel-wise Retrieval	DAVIS$_{16}$ [17]
2019	IVS [180]	CVPR	FCN	ResNet50	Interaction-Propagation	DAVIS$_{17}$ [87]+YouTube-VOS [101]
2020	MANet [181]	CVPR	Siamese FCN	ResNet101	Interaction-Propagation	DAVIS$_{17}$ [87]
	ATNet [182]	ECCV	FCN	SE-ResNet50	Interaction-Propagation	SBD + DAVIS$_{17}$ [87]+YouTube-VOS [101]
	ScribbleBox [183]	ECCV	GCN	ResNet50	Interaction-Propagation	COCO [124] + ImageNet VID [73] + YouTube-VOS [101]
2021	IVOS-W [184]	CVPR	FCN + RL	ResNet50	Keyframe Selection	DAVIS$_{17}$ [87]
	GIS [185]	CVPR	FCN	SE-ResNet50	Interaction-Propagation	DAVIS$_{17}$ [87]+YouTube-VOS [101]
	MiVOS [186]	CVPR	Memory Network	ResNet50	Interaction-Propagation	BL30K [186]+DAVIS$_{17}$ [87] + YouTube-VOS [101]

TABLE 4
Summary of essential characteristics for reviewed LVOS methods. See §3.1.4 for more detailed descriptions.

Year	Method	Pub.	Visual + Language Encoder	Technical Feature	Training Dataset
2018	Gavrilyuk *et al.* [190]	CVPR	I3D + CNN	Dynamic Convolution	A2D Sentences [190]
	Khoreva *et al.* [191]	ACCV	CNN + CNN	Cross-modal Attention	DAVIS$_{17}$-RVOS [190]
2019	Wang *et al.* [192]	ICCV	I3D + CNN	Cross-modal Attention	A2D Sentences [190]
2020	Wang *et al.* [193]	AAAI	I3D + GRU	Dynamic Convolution	A2D Sentences [190]
	Ning *et al.* [194]	IJCAI	I3D + LSTM	Dynamic Convolution	A2D Sentences [190]
	McIntosh *et al.* [195]	CVPR	I3D + CNN	Capsule Routing	A2D Sentences [190]
	URVOS [196]	ECCV	CNN + MLP	Cross-modal Attention	Refer-YouTube-VOS [196]
2021	Hui *et al.* [197]	CVPR	I3D + GRU	Cross-modal Attention	A2D Sentences [190]
	CMSANet [198]	PAMI	CNN + Word embed.	Cross-modal Attention	A2D Sentences [190]

the whole video for user interaction, so as to boost mask propagation. In [165], Li *et al*. introduce a forward-backward data flow based cycle consistency mechanism to improve both traditional SVOS training and offline inference protocols, through mitigating the error propagation problem. To accelerate processing speed, a dynamic network [168] is proposed to selectively allocate computation source for each frame, according to the similarity to the previous frame.

3.1.3 Interactive Video Object Segmentation (IVOS)

AVOS, without any human involvement, loses flexibility in segmenting arbitrary objects of user interest. SVOS additionally considers first-frame annotations, but easily fails in challenging scenes, without involving human feedback. Moreover, the first-frame annotations are typically detailed masks, which are tedious to acquire: 79 seconds per instance on the coarse polygon annotations of COCO [124], significantly more for higher quality. Thus performing VOS in the interactive setting gained increasing attention. Unlike traditional models [51], [52], [55] requiring extensive and professional user intervention, recent deep learning based IVOS solutions usually work under a multi-round, scribble-supervision setting, so as to minimize the user's effort. In this scenario [199], the user draws scribbles on a selected frame and an algorithm computes the segmentation maps for all video frames in a batch process. To refine the results, user intervention and segmentation are repeated. This *round-based interaction* [180] is useful for consumer-level applications and rapid prototyping for professional usage, where the efficiency is the main concern. One can control the quality of segmentation according to the time budget, as more rounds of interactions will provide better results.

• **Interaction-Propagation based Methods.** The majority of current studies [185], [186] follow an *interaction-propagation* scheme. In the primary attempt [178], IVOS is achieved by a simple combination of two separate modules: an interactive image segmentation model [200] for producing and refining segmentation under user supervision; and a SVOS model [18] for propagating masks from the user-annotated frames to the rest ones. Later, [180] proposed a more compact solution, which also has two modules for interaction and propagation respectively. However, the two modules are internally connected through intermediate feature exchanging, and also externally connected, *i.e.*, each of them conditions on the other's output. In [182], a similar model design is also adopted, while the propagation part is equipped with two specific transfer modules for addressing local mask tracking (over adjacent frames) and global propagation (among distant frames), respectively. Though effective, [178], [182] have to start a new feed-forward computation in each interaction round, making them inefficient when rounds grow up. A more efficient solution was developed in [181]. The critical idea is to build a common encoder for discriminative pixel embedding learning, upon which two small network branches are added for interactive segmentation and mask propagation, respectively. Thus the model extracts pixel embeddings for all frames only once (in the first round). In the following rounds, the feed-forward computation is only made within the two shallow branches.

• **Other Methods.** Chen *et al*. [179] propose a pixel embedding learning based model, applicable to both SVOS and IVOS. With a similar idea of [127], IVOS is formulated as a pixel-wise retrieval problem, *i.e.*, transferring labels to each pixel according to its nearest reference pixel. This model supports different kinds of user input, such as masks, clicks and scribbles, and can provide immediate feedback after user interaction. In [183], an interactive annotation tool is proposed for VOS. The annotation has two phases: annotating objects with tracked boxes, and labeling masks inside these tracks. Box tracks are annotated efficiently by approximating the trajectory using a parametric curve with a small number of control points which the annotator can interactively correct. Segmentation masks are corrected via scribbles which are propagated through time. In [184], a reinforcement learning framework is exploited to automatically determine the most valuable frame for interaction.

TABLE 5
Summary of essential characteristics for reviewed VSS methods. Flow: whether use optical flow. See §3.2 for more detailed descriptions.

Year	Method	Pub.	Seg. Level	Core Architecture	Backbone	Flow	Technical Feature	Training Dataset
2016	Clockwork [201]	ECCV	Semantic	FCN	VGGNet	✓	Faster Segmentation	Cityscapes [202]/YouTube-Objects [79]
	FSO [203]	CVPR	Semantic	FCN + Dense CRF	VGGNet	✓	Temporal Feature Aggregation	Cityscapes [202]/CamVid [204]
	JFS [205]	ECCV	Semantic	FCN	VGGNet	✓	Temporal Feature Aggregation	KITTI MOTS [206]
2017	BANet [207]	CVPR	Semantic	FCN + LSTM	VGGNet		Keyframe Selection	CamVid [204]/KITTI
	PEARL [208]	ICCV	Semantic	FCN	VGGNet/ResNet101	✓	Flow-guided Feature Aggregation	Cityscapes [202]/CamVid [204]
	NetWarp [209]	ICCV	Semantic	Siamese FCN	ResNet101	✓	Flow-guided Feature Aggregation	Cityscapes [202]/CamVid [204]
	DFF [210]	ICCV	Semantic	FCN	ResNet50/101		Flow-guided Feature Aggregation	Cityscapes [202]
	Saleh *et al.* [211]	ICCV	Semantic	Two-Stream FCN	VGGNet	✓	Weakly-Supervised Learning	Cityscapes [202]/CamVid [204]
2018	GRFP [212]	CVPR	Semantic	FCN + GRU	VGGNet	✓	Temporal Feature Aggregation	Cityscapes [202]/CamVid [204]
	LVS [213]	CVPR	Semantic	FCN	ResNet101		Keyframe Selection	Cityscapes [202]/CamVid [204]
	DVSN [214]	CVPR	Semantic	FCN+RL	ResNet101	✓	Keyframe Selection	Cityscapes [202]
	EUVS [215]	ECCV	Semantic	Bayesian CNN	-	✓	Flow-guided Feature Aggregation	CamVid [204]
	GCRF [216]	CVPR	Semantic	FCN+CRF	ResNet101	✓	Gaussian CRF	CamVid [204]
2019	Accel [217]	CVPR	Semantic	FCN	ResNet101	✓	Keyframe Selection	KITTI
	Zhu *et al.* [218]	CVPR	Semantic	FCN	WideResNet38		Weakly-Supervised Learning	Cityscapes [202]/CamVid [204]
	MOTS [206]	CVPR	Instance	Mask R-CNN	ResNet101		Tracking by Detection	KITTI MOTS [206] /MOTSChallenge [206]
	MaskTrack R-CNN [88]	ICCV	Instance	Mask R-CNN	ResNet50-FPN		Tracking by Detection	YouTube-VIS [88]
2020	EFC [219]	AAAI	Semantic	FCN	ResNet50	✓	Temporal Feature Aggregation	Cityscapes [202]/CamVid [204]
	TDNet [220]	CVPR	Semantic	Memory Network	ResNet50		Attention-based Feature Aggregation	Cityscapes [202]/CamVid [204]/NYUDv2 [221]
	MaskProp [222]	CVPR	Instance	Mask R-CNN	ResNeXt-101-64x4d + FPN		Instance Feature Propagation	YouTube-VIS [88]
	VPS [223]	CVPR	Panoptic	Mask R-CNN	ResNet50-FPN		Spatio-Temporal Feature Alignment	VIPER-VPS [223]/Cityscapes-VPS [223]
	MOTSNet [224]	CVPR	Instance	Mask R-CNN	ResNet50-FPN		Unsupervised Learning	KITTI MOTS [206] /BDD100K [225]
	MVAE [226]	CVPR	Instance	Mask R-CNN+VAE	ResNet101		Variational Inference	KITTI MOTS [206] /YouTube-VIS [88]
	ETC [227]	ECCV	Semantic	FCN + KD	ResNet18/MobileNetV2/HRNet18	✓	Knowledge Distillation	Cityscapes [202]/CamVid [204]
	Sipmask [228]	ECCV	Instance	FCOS	ResNet50/101		Single-Stage Segmentation	YouTube-VIS [88]
	STEm-Seg [229]	ECCV	Instance	FCN	ResNet101		Spatio-Temporal Embedding Learning	DAVIS$_{17}$ [87]/YouTube-VIS [88]/KITTI-MOTS [206]
	Naive-Student [230]	ECCV	Semantic	FCN+KD	Xception-71/WideResNet38		Semi-Supervised Learning	Cityscapes [202]
2021	CompFeat [231]	AAAI	Instance	Mask R-CNN	ResNet50		Spatio-Temporal Feature Alignment	YouTube-VIS [88]
	TraDeS [232]	CVPR	Instance	Siamese FCN	Bi-directional GRU		Tracking by Detection	MOT/nuScenes/KITTI MOTS [206] /YouTube-VIS [88]
	SG-Net [233]	CVPR	Instance	FCOS	ResNet50/101-FPN		Single-Stage Segmentation	YouTube-VIS [88]
	VisTR [234]	CVPR	Instance	Transformer	ResNet50		Transformer-based Segmentation	YouTube-VIS [88]
	Propose-Reduce [235]	CVPR	Instance	Mask R-CNN	ResNet50/101/ResMeXt-101		Propose and Reduce	DAVIS$_{17}$ [87]/YouTube-VIS [88]
	Lukas *et al.* [236]	CVPR	Semantic	FCN	ResNet101		Semi-Supervised Learning	Cityscapes [202]
	SiamTrack [237]	CVPR	Panoptic	Siamese FCN	ResNet50-FPN		Supervised Contrastive Learning	VIPER-VPS [223]/Cityscapes-VPS [223]
	ViP-DeepLab [238]	CVPR	Panoptic	FCN	ResNet101		Depth-Aware Panoptic Segmentation	Cityscapes-VPS [223]
	fIRN [239]	CVPR	Instance	Mask R-CNN	ResNet50	✓	Weakly-Supervised Learning	YouTube-VIS [88]/Cityscapes [202]
	Fu *et al.* [240]	CVPR	Instance	SOLO	ResNet50		Semi-Supervised Learning	YouTube-VIS [88]/Cityscapes [202]

3.1.4 Language-guided Video Object Segmentation (LVOS)

LVOS is an emerging area, dating back to 2018 [190], [191]. Although there have already exist some efforts [241]–[243] in the intersection of language and video understanding, none of them addresses pixel-level video-language reasoning. Most research efforts in LVOS are made around the theme of efficient alignment between visual and linguistic modalities. According to the multi-modal information fusion strategy, existing models can be divided into three groups.

• **Dynamic Convolution based Methods.** The first initiate was proposed in [190] that applies dynamic networks [244] for visual-language relation modeling. Specifically, convolve filters, dynamically generated from linguistic query, are used to adaptively transform visual features into desired segments. In the same line of work, [193], [194] incorporate spatial context into filter generation. However, as indicated by [192], linguistic variation of input description may greatly impact sentence representation and subsequently make dynamic filters unstable, causing inaccurate segmentation. For example, "car in blue is parked on the grass" and "blue car standing on the grass" have the same meaning but different generated filters, leading to poor performance.

• **Capsule Routing based Methods.** In [195], a visual-textual capsule routing model was developed for LVOS. Both video and textual inputs are encoded through capsules [245], which are considered effective in modeling visual/textual entities. Then, dynamic routing is applied over the video and text capsules for visual-textual information integration.

• **Attention based Methods.** Neural attention mechanism is also widely adopted in the filed of LVOS [191], [196], [198], [246], due to its advantage in modeling global visual/textual context. In [192], two modules, called vision guided language attention and language guided vision attention, were developed to capture visual-textual correlations. In [197], two different attentions are learned to ground spatial- and temporal-content relevant linguistic cues to static and dynamic visual embeddings, respectively. In [246], multi-step, cross-modality attention based reasoning is conducted.

3.2 Deep Learning based VSS Models

Extending the success of deep learning based image semantic segmentation techniques to the video domain has become one of the research focus in computer vision recently. To achieve this, the most straightforward strategy is the naïve application of an image semantic segmentation model in a frame-by-frame manner. But this strategy completely ignores temporal continuity and coherence cues provided in videos. To make better use of temporal information, research efforts in this field are mainly made along two lines.

• **Efforts towards More Accurate Segmentation.** A major stream of methods exploits cross-frame relations to boost the prediction accuracy. They typically first apply the very same segmentation algorithms to each frame independently. Then they add extra modules on top, *e.g.*, optical flow-guided feature aggregation [208], [209], [215], and sequential network based temporal information propagation [212], to gather multi-frame context and get better results. For example, in some pioneer work [203], [205], after performing static semantic segmentation for each frame individually, optical flow [205] or dense 3D CRF [203] based post processing is applied for gaining temporally consistent segments. Later, [216] proposes to jointly learn CNN based per-frame segmentation and CRF based spatio-temporal reasoning in an end-to-end manner. In [209], features wrapped from previous frames with optical flow are combined with the current frame features for prediction. These methods require additional feature aggregation modules, which increase the computational costs during the inference phase. Recently,

[219] proposes to only incorporate flow-guided temporal consistency into the training phase, without bringing any extra inference cost. But its processing speed is still bounded to the adopted per-frame semantic segmentation algorithms, as all features must be recomputed at each frame. For these methods, the utility in time-sensitive application areas, such as mobile and autonomous driving, is limited.

• **Efforts towards Faster Segmentation.** Yet another complementary line of work focuses on leveraging temporal information to accelerate computation. They typically approximate the expensive per-frame forward pass with cheaper alternatives, *i.e.*, reusing the features in neighbouring frames. In [201], parts of segmentation networks are adaptively executed across frames, thus reducing the computation cost. Later methods use keyframes to avoid processing of each frame, and then propagate the outputs or the feature maps to other non-key frames. For example, [210] employs optical flow to warp the feature map between the keyframe and non-key frames. Adaptive keyframe selection is later exploited in [207], [213], [214], further enhanced by adaptive feature propagation [213]. In [217], Jain *et al.* use a large, strong model to predict the keyframe and use a compact one in non-key frames. Keyframe based methods have different computational loads between keyframes and non-keyframes, which causes high maximum latency and unbalanced occupation of computation resources that may decrease system efficiency [220]. Additionally, the spatial misalignment of other frames with respect to the keyframes is challenging to compensate for and often leads to different quantity results between keyframes and other frames. In [227], a temporal consistency guided knowledge distillation technique is proposed to train a compact network, which is applied to all frames. In [220], several weight-sharing sub-networks are distributed over sequential frames, whose extracted shallow features are composed for final segmentation. This trend of methods indeed speeds up inference, but still with the cost of reduced accuracy.

• **Video Instance Segmentation (VIS) Methods.** In 2019, Yang *et al*. extended image instance segmentation to the video domain [88], which requires simultaneous detection, segmentation and tracking of instances in videos. This task is also known as *multi-object tracking and segmentation* (MOTS) [206]. Based on the patterns of generating instance sequences, existing frameworks can be roughly categorized into four paradigms: i) *track-detect*, ii) *clip-match*, iii) *propose-reduce*, iv) *segment-as-a-whole*. Track-detect methods detect and segment instances for each individual frame, followed by obtaining instance sequences with frame-by-frame tracking [88], [206], [224], [226], [228], [231]–[233], [249]. For example, in [88], [206], Mask R-CNN [177] is adapted for VIS/MOTS by adding a tracking branch for cross-frame instance association. Clip-match methods divide an entire video into multiple short overlapped clips, and obtain VIS results for each clip through mask propagation [222] or spatial-temporal embedding [229]. Final instance sequences are generated by matching and merging neighboring clips. Both of the two paradigms need two independent steps to generate a complete sequence. They both generate multiple incomplete sequences (*i.e.*, frames or clips) from a video, and merge (or complete) them by tracking/matching at the second stage. Intuitively, these paradigms are vulnerable to error accumulation in the process of merging sequences, especially when occlusion or fast motion exists. To address these limitations, propose-reduce paradigm is proposed in [235]. It first samples several key frames and obtains instance sequences by propagating the instance segmentation results from each key frame to the entire video. Then, the redundant sequence proposals of the same instances are removed. This paradigm not only discards the step of merging incomplete sequences, but also achieves robust results considering multiple key frames. However, these three types of methods still need complex heuristic rules to associate instances and/or multiple steps to generate instance sequences. Segment-as-a-whole paradigm [234] poses the task as a direct sequence prediction problem by using the Transformer [189].

• **Video Panoptic Segmentation (VPS) Methods.** Very recently, Kim *et al*. extended image panoptic segmentation to the video domain [223], which aims at a holistic segmentation of all foreground instance tracklets and background regions, and assigning a semantic label to each video pixel. They adapt an image panoptic segmentation model [250] for VPS, by adding two modules for temporal feature fusion and cross-frame instance association, respectively. Later, temporal correspondence was explored in [237] through learning coarse segment-level and fine pixel-level matching. Qiao *et al*. [238] propose to learn monocular depth estimation and video panoptic segmentation jointly.

• **Semi-/Weakly-Supervised Methods.** Away from these main battlefields, some researchers made efforts to learn VSS/VIS under semi-/weakly-supervised settings. In [211], classifier heatmaps are used to learn VSS from image tags only. Some others [218], [230] leverage both labeled and unlabeled video frames for VSS model training. They propagate annotations from labeled frames to other unlabeled, neighboring frames [218], or alternatively train teacher and student networks with groundtruth annotations and iteratively generated pseudo labels [230]. In [239], motion and temporal consistency cues are leveraged to generate high-quality pseudo-labels from tag labeled videos for weakly supervised VIS learning. In [240], a semi-supervised embedding learning approach is proposed to learn VIS from pixel-wise annotated image and unlabeled videos.

4 Video Segmentation Datasets

Several datasets have been proposed for video segmentation over the past decades. We summarize the commonly used datasets in Table 6 and give detailed review below.

4.1 VOS Datasets

4.1.1 AVOS/SVOS/IVOS Datasets

• **Youtube-Objects** is a large dataset of 1,407 videos collected from 155 web videos belonging to 10 object categories (*e.g.*, dog, cat, plane, *etc*.). VOS models typically test the generalization ability on its subset [251], which has totally 126 shots with 20,647 frames, and provides coarse pixel-level fore-/back-ground annotations on every 10^{th} frames.

• **FBMS**$_{59}$ [38] consists of 59 video sequences with 13,860 frames in total. However, only 720 frames are annotated for fore-/back-ground separation. The dataset is split into 29 and 30 sequences for training and evaluation, respectively.

TABLE 6
Statistics of representative video segmentation datasets. See §4.1 and §4.2 for more detailed descriptions.

Dataset	Year	Pub.	#Video	#Train/Val/Test/Dev	Annotation	Purpose	#Class	Synthetic
Youtube-Objects [79]	2012	CVPR	1,407 (126)	-/-/-/-	Object-level AVOS, SVOS	Generic	10	
FBMS$_{59}$ [38]	2014	PAMI	59	29/30/-/-	Object-level AVOS, SVOS	Generic	-	
DAVIS$_{16}$ [17]	2016	CVPR	50	30/20/-/-	Object-level AVOS, SVOS	Generic	-	
DAVIS$_{17}$ [87]	2017	-	150	60/30/30/30	Instance-level AVOS, SVOS, IVOS	Generic	-	
YouTube-VOS [101]	2018	-	4,519	3,471/507/541/-	SVOS	Generic	94	
A2D Sentence [190]	2018	CVPR	3,782	3,017/737/-/-	RVOS	Human-centric	-	
J-HMDB Sentence [190]	2018	CVPR	928	-/-/-/-	RVOS	Human-centric	-	
DAVIS$_{17}$-RVOS [191]	2018	ACCV	90	60/30/-/-	RVOS	Generic	-	
Refer-Youtube-VOS [196]	2020	ECCV	3,975	3,471/507/-/-	RVOS	Generic	-	
CamVid [204]	2009	PRL	4	(frame: 467/100/233/-)	VSS	Urban	11	
CityScapes [202]	2016	CVPR	5,000	2,975/500/1,525	VSS	Urban	19	
NYUDv2 [221]	2012	ECCV	518	(frame: 795/654/-/-)	VSS	Indoor	40	
VSPW [247]	2021	CVPR	3,536	2,806/343/387/-	VSS	Generic	124	
YouTube-VIS [88]	2019	ICCV	3,859	2,985/421/453/-	VIS	Generic	40	
KITTI MOTS [206]	2019	CVPR	21	12/9/-/-	VIS	Urban	2	
MOTSChallenge [206]	2019	CVPR	4	-/-/-/-	VIS	Urban	1	
BDD100K [225]	2020	ECCV	100,000	7,000/1,000/2,000/-	VSS, VIS	Driving	40 (VSS), 8 (VIS)	
OVIS [248]	2021	-	901	607/140/154/-	VIS	Generic	25	
VIPER-VPS [223]	2020	CVPR	124	(frame: 134K/50K/70K/-)	VPS	Urban	23	✓
Cityscapes-VPS [223]	2020	CVPR	500	400/100/-/-	VPS	Urban	19	

• **DAVIS$_{16}$** [17] is comprised of 50 videos (30 for *train* set and 20 for *val* set) with 3,455 frames in total. For each frame, in addition to high-quality fore-/back-ground segmentation annotation, a set of attributes (*e.g.*, deformation, occlusion, motion blur, *etc.*) representing typical challenging factors in video processing, are also provided.

• **DAVIS$_{17}$** [87] contains 150 videos, *i.e.*, 60/30/30/30 videos for *train*/*val*/*test-dev*/*test-challenge* sets. Its *train* and *val* sets are extended from the respective sets in DAVIS$_{16}$. There are 10,459 frames in total. DAVIS$_{17}$ provides instance-level annotations to support SVOS. Then, DAVIS$_{18}$ challenge[252] provides scribble annotations to support IVOS. Moreover, as the original annotations of DAVIS$_{17}$ are biased towards the SVOS scenario, DAVIS$_{19}$ challenge[175] re-annotates *val* and *test-dev* sets of DAVIS$_{17}$ to support AVOS.

• **YouTube-VOS** [101] is a large-scale dataset, which is split into a *train* (3,471 videos), *val* (507 videos), and *test* (541 videos) set, in its newest 2019 version. Instance-level precise annotations are provided every five frames in a 30FPS frame rate. There are 94 object categories (*e.g.*, person, snake, *etc.*) in total, of which 26 are unseen in *train* set.

Youtube-Objects, FBMS$_{59}$ and DAVIS$_{16}$ are widely used for instance-agnostic AVOS and SVOS evaluation. DAVIS$_{17}$ is unique in comprehensive annotations for instance-level AVOS, SVOS as well as IVOS, but its scale is relatively small. YouTube-VOS is the largest one but only supports SVOS benchmarking. There also exist some other VOS datasets, such as SegTrack$_{V1}$ [48] and SegTrack$_{V2}$ [82], but they were less used recently, due to the limited scale and difficulty.

4.1.2 RVOS Datasets

• **A2D Sentence** [190] augments A2D [253] with phrases. It contains 3,782 videos, with 8 action classes performed by 7 actors. In each video, 3 to 5 frames are provided with segmentation masks. It contains 6,655 sentences describe actors and their actions. The dataset is split into 3,017/737 for `train/test`, and 28 unlabeled videos are ignored [192].

• **J-HMDB Sentence** [190] is built upon J-HMDB [254]. It is comprised of 928 short videos with 928 corresponding sentences describing 21 different action categories.

• **DAVIS$_{17}$-RVOS** [191] extends DAVIS$_{17}$ by collecting referring expressions for the annotated objects. 90 videos from `train` and `val` sets are annotated with more than 1,500 referring expressions. They provide two types of annotations, which describe the highlighted object: i) based on a entire video (*i.e.*, full-video expression) and ii) using only the first frame of the video (*i.e.*, first-frame expression).

• **Refer-Youtube-VOS** [196] includes 3,975 videos from YouTube-VOS [101], with 27,899 language descriptions of target objects. Similar to DAVIS$_{17}$-RVOS [191], both full-video and first-frame expression annotations are provided.

To date, A2D Sentence and J-HMDB Sentence are the main test-bed. However, the phrases are not produced with the aim of reference, but description, and limited to only a few object categories corresponding to the dominant 'actors' performing a salient 'action' [196]. But the new introduced DAVIS$_{17}$-RVOS and Refer-Youtube-VOS datasets show improved difficulties in both visual and linguistic modalities.

4.2 VSS Datasets

• **CamVid** [204] is composed of 4 urban scene videos with 11-class pixelwise annotations. Each video is annotated every 30 frames. The annotated frames are usually grouped into 467/100/233 for `train/val/test` [203].

• **CityScapes** [202] is a large-scale VSS dataset for street views. It has 2,975/500/1,525 snippets for `train/val/test`, captured at 17FPS. Each snippet contains 30 frames, and only the 20th frame is densely labelled with 19 semantic classes. 20,000 coarsely annotated frames are also provided.

• **NYUDv2** [221] contains 518 indoor RGB-D videos with high-quality ground-truths (every 10th video frame is labeled). There are 795 training frames and 654 testing frames being rectified and annotated with 40-class semantic labels.

• **VSPW**[247] is a recently proposed large-scale VSS dataset. It addresses video scene parsing in the wild by considering diverse scenarios. It consists of 3,536 videos, and provides pixel-level annotations for 124 categories at 15FPS. The `train/val/test` sets contain 2,806/343/387 videos with 198,244/24,502/28,887 frames, respectively.

• **YouTube-VIS** [88] is built upon YouTube-VOS [101] with instance-level annotations. Its newest 2021 version has 3,859 videos (2,985/421/453 for `train/val/test`) with 40 semantic categories. It provides 232K high-quality annotations for 8,171 unique video instances.

TABLE 7

Quantitative object-level AVOS results on the `val` set of DAVIS$_{16}$ [17] (§5.1.2) in terms of region similarity $\mathcal{J}$, boundary accuracy $\mathcal{F}$ and time stability $\mathcal{T}$. We also report the recall and the decay performance over time for both $\mathcal{J}$ and $\mathcal{F}$. (FPS denotes *frames per second*. The three best scores are marked in **red**, **blue**, and **green**, respectively. These notes also apply to the other tables.)

Year	Pub.	Method	$\mathcal{J}$ Mean↑	$\mathcal{J}$ Recall↑	$\mathcal{J}$ Decay↓	$\mathcal{F}$ Mean↑	$\mathcal{F}$ Recall↑	$\mathcal{F}$ Decay↓	$\mathcal{T}$ Mean ↓	FPS↑
2017	ICCV	SFL [74]	67.4	81.4	6.2	66.7	77.1	5.1	28.2	3.3
	ICCV	LVO [75]	75.9	89.1	0.0	72.1	83.4	1.3	26.5	-
	CVPR	LMP [76]	70.0	85.0	1.3	65.9	79.2	2.5	57.2	-
	CVPR	FSEG [72]	70.7	83.0	1.5	65.3	73.8	1.8	32.8	-
2018	ECCV	PDB [84]	77.2	93.1	0.9	74.5	84.4	-0.2	29.1	20
	ECCV	MOT [85]	77.2	87.8	5.0	77.4	84.4	3.3	27.9	-
2019	IJCV	LSMO [99]	78.2	91.1	4.1	75.9	84.7	3.5	21.2	-
	CVPR	AGS [71]	79.7	89.1	1.9	77.4	85.8	0.0	26.7	1.7
	CVPR	COSNet [89]	80.5	93.1	4.4	79.4	89.5	5.0	18.4	2.2
	ICCV	AGNN [95]	81.3	93.1	0.0	79.7	88.5	5.1	33.7	1.9
	ICCV	AnDiff [98]	81.7	90.9	2.2	80.5	85.1	0.6	21.4	2.8
2020	AAAI	Gu *et al.* [102]	78.1	90.1	-	78.5	88.2	-	-	110
	AAAI	MATNet [100]	82.4	94.5	5.5	80.7	90.2	4.5	21.6	1.3
	ECCV	EGMN [105]	82.5	94.3	4.2	81.2	90.3	5.6	19.8	-
	ECCV	DFNet [109]	83.4	-	-	81.8	-	-	15.9	3.6
2021	AAAI	F2Net [110]	83.1	95.7	0.0	84.4	92.3	0.8	20.9	10
	CVPR	RTNet [112]	85.6	96.1	-	84.7	93.8	-	-	-

• **KITTI MOTS** [206] extends the 21 training sequences of KITTI tracking dataset [255] with VIS annotations – 12 for training and 9 for validation, respectively. The dataset contains 8,008 frames with a resolution of 375 × 1242, 26,899 annotated cars and 11,420 annotated pedestrians.

• **MOTSChallenge** [206] annotates 4 of 7 training sequences of MOTChallenge$_{2017}$ [256]. It has 2,862 frames with 26,894 annotated pedestrians and presents many occlusion cases.

• **BDD100K** [225] is a large-scale dataset with 100K driving videos (40 seconds and 30FPS each) and supports various tasks, including VSS and VIS. For VSS, 7,000/1,000/2,000 frames are densely labelled with 40 semantic classes for `train/val/test`. For VIS, 90 videos with 8 semantic categories are annotated by 129K instance masks – 60 training videos, 10 validation videos, and 20 testing videos.

• **OVIS** [248] is a new challenging VIS dataset, where object occlusions usually occur. It has 901 videos and 296K high-quality instance masks for 25 semantic categories. It is split into 607 training, 140 validation and 154 test videos.

• **VIPER-VPS** [223] re-organizes VIPER [257] into the video panoptic format. VIPER, extracted from the GTA-V game engine, has annotations of semantic and instance segmentations for 10 thing and 13 stuff classes on 254K frames of ego-centric driving scenes at 1080 × 1920 resolution.

• **Cityscapes-VPS** [223] is built upon CityScapes [202]. Dense panoptic annotations for 8 thing and 11 stuff classes for 500 snippets in Cityscapes `val` set are provided every five frames and temporally consistent instance ids to the thing objects are also given, leading to 3000 annotated frames in total. These videos are split into 400/100 for `train/val`.

CamVid, CityScapes, NYUDv2, and VSPW are built for VSS benchmarking. YouTube-VIS, OVIS, KITTI MOTS, and MOTSChallenge are VIS datasets, but the diversity of the last two are limited. BDD100K has both VSS and VIS annotations. VIPER-VPS and Cityscapes-VPS are aware of VPS evaluation, but VIPER-VPS is a synthesized dataset.

5 Performance Comparison

Next we tabulate the performance of several of previously discussed algorithms. The scores are gathered from the original articles. For each of the reviewed fields: object-level AVOS, instance-level AVOS, SVOS, IVOS, LVOS, VSS, VIS, and VPS, the most widely used dataset is selected for performance benchmarking[2]. It is essential to remark the heterogeneity of the papers in the field when reporting results. Although most of them try to evaluate their methods in standard datasets and provide enough information to reproduce their results, many others fail to do so. This makes it hard or even impossible to fairly compare methods.

5.1 Object-level AVOS Performance Benchmarking

5.1.1 Evaluation Metrics

Presently, three metrics are frequently used [17] to measure how object-level AVOS methods perform on this task:

• **Region Jaccard** $\mathcal{J}$ is calculated by the intersection-over-union (IoU) between the segmentation results $\hat{Y} \in \{0,1\}^{w\times h}$ and the ground-truth $Y \in \{0,1\}^{w\times h}$:

$$\mathcal{J} = \frac{\hat{Y} \cap Y}{\hat{Y} \cup Y}, \tag{2}$$

which computes the number of pixels of the intersection between $\hat{Y}$ and Y, and divides it by the size of the union.

• **Boundary Accuracy** $\mathcal{F}$ is the harmonic mean of the boundary precision P_c and recall R_c. The value of $\mathcal{F}$ reflects how well the segment contours $c(\hat{Y})$ match the ground-truth contours $c(Y)$. Usually, the value of P_c and R_c can be computed via bipartite graph matching [258], then the boundary accuracy $\mathcal{F}$ can be computed as:

$$\mathcal{F} = \frac{2\mathrm{P}_c\mathrm{R}_c}{\mathrm{P}_c + \mathrm{R}_c}. \tag{3}$$

• **Temporal Stability** $\mathcal{T}$ is informative of the stability of segments. It is computed as the pixel-level cost of matching two successive segmentation boundaries. The match is achieved by minimizing the shape context descriptor [259] distances between matched points while preserving the order in which the points are present in the boundary polygon. Note that $\mathcal{T}$ will compensate motion and small deformations, but not penalize the inaccuracies of the contours [17].

[2] For the methods reporting multiple scores due to the use of different training settings or backbone models, we adopt the best one.

TABLE 8
Quantitative instance-level AVOS results on the `val` set of DAVIS$_{17}$ [87] (§5.2.2) in terms of region similarity $\mathcal{J}$ and boundary accuracy $\mathcal{F}$.

Year	Pub.	Method	$\mathcal{J}\&\mathcal{F}$ Mean$\uparrow$	$\mathcal{J}$ Mean$\uparrow$	$\mathcal{J}$ Recall$\uparrow$	$\mathcal{J}$ Decay$\downarrow$	$\mathcal{F}$ Mean$\uparrow$	$\mathcal{F}$ Recall$\uparrow$	$\mathcal{F}$ Decay$\downarrow$	FPS$\uparrow$
2018	ECCV	PDB [84]	55.1	53.2	58.9	4.9	57.0	60.2	6.8	0.7
2019	CVPR	RVOS [86]	41.2	36.8	40.2	0.5	45.7	46.4	1.7	14.3
	CVPR	AGS [71]	57.5	55.5	61.6	7.0	59.5	62.8	9.0	1.1
	ICCV	AGNN [174]	61.1	58.9	65.7	11.7	63.2	67.1	14.3	-
2020	ECCV	STEm-Seg [229]	64.7	61.5	70.4	-4.0	67.8	75.5	1.2	-
	WACV	UnOVOST [176]	67.9	66.4	76.4	-0.2	69.3	76.9	0.0	1.0
2021	CVPR	Zhou *et al.* [111]	65.0	63.7	71.9	6.9	66.2	73.1	9.4	9.1

5.1.2 Results

We select DAVIS$_{16}$ [17], the most widely used dataset in AVOS, for performance benchmarking. Table 7 presents the results of those reviewed AVOS methods DAVIS$_{16}$ `val` set. Current winner solution, RTNet [112], reaches 85.6 region similarity $\mathcal{J}$, significantly outperforming the early deep learning based methods, such as SFL [74], proposed in 2017.

5.2 Instance-level AVOS Performance Benchmarking

5.2.1 Evaluation Metrics

In instance-level AVOS setting, region Jaccard $\mathcal{J}$, boundary accuracy $\mathcal{F}$, and $\mathcal{J}\&\mathcal{F}$ – the mean of $\mathcal{J}$ and $\mathcal{F}$ – are used for evaluation. During evaluation [175], each of the annotated object tricklets will be matched with one of predicted tricklets according to $\mathcal{J}\&\mathcal{F}$, using bipartite graph matching. For a certain criterion, the final score will be computed between each ground-truth object and its optimal assignment.

5.2.2 Results

Regarding instance-level AVOS, we take into account DAVIS$_{17}$ [87] in which the vast majority of methods are evaluated. From Table 8 we can find that UnOVOST [176] is the top scorer, with 67.9 $\mathcal{J}$ at the time of this writing.

5.3 SVOS Performance Benchmarking

5.3.1 Evaluation Metrics

Region Jaccard $\mathcal{J}$, boundary accuracy $\mathcal{F}$, and $\mathcal{J}\&\mathcal{F}$ are also widely adopted for SVOS performance evaluation [199].

5.3.2 Results

DAVIS$_{17}$ [87] is also one of the most important SVOS dataset. Table 9 shows the results of recent SVOS methods on DAVIS$_{17}$ `val` set. In this case, all the top-leading solutions, such as EGMN [105], LCM [171], and RMNet [172], are built upon the memory augmented architecture – STM [150].

5.4 IVOS Performance Benchmarking

5.4.1 Evaluation Metrics

Area under the curve (AUC) and Jaccard at 60 seconds ($\mathcal{J}$@60s) are two widely used IVOS evaluation criteria [199].

- **AUC** is designed to measure the overall accuracy of the evaluation. It is computed over the plot Time *vs* Jaccard. Each sample in the plot is computed considering the average time and the average Jaccard for a certain interaction.
- **$\mathcal{J}$@60** measures the accuracy with a limited time budget (60 seconds). It is achieved by interpolating the Time *vs* Jaccard plot at 60 seconds. This evaluates which quality an IVOS method can obtain in 60 seconds.

TABLE 9
Quantitative SVOS results on the `val` set of DAVIS$_{17}$ [87] (§5.3.2) in terms of region similarity $\mathcal{J}$ and boundary accuracy $\mathcal{F}$.

Year	Pub.	Method	$\mathcal{J}\&\mathcal{F}$ Mean$\uparrow$	$\mathcal{J}$ Mean$\uparrow$	$\mathcal{F}$ Mean$\uparrow$	FPS$\uparrow$
2017	BMVC	OnAVOS [123]	67.9	64.5	70.5	0.08
	CVPR	OSVOS [18]	60.3	56.7	63.9	0.22
2018	CVPR	CINM [126]	67.5	64.5	70.5	0.01
	CVPR	FAVOS [127]	58.2	54.6	61.8	0.56
	CVPR	RGMP [128]	66.7	64.8	68.6	7.7
	CVPR	OSMN [129]	54.8	52.5	57.1	7.7
	PAMI	OSVOS-S [133]	68.0	64.7	71.3	0.22
	ECCV	Videomatch [134]	61.4	-	-	0.38
	ECCV	Dyenet [135]	69.1	67.3	71.0	2.4
2019	PAMI	MVOS [139]	59.2	56.3	62.1	1.5
	CVPR	FEELVOS [140]	71.5	69.1	74.0	2.2
	CVPR	MHP-VOS [141]	75.3	71.8	78.8	0.01
	CVPR	AGSS [142]	67.4	64.9	69.9	10
	CVPR	AGAME [143]	70.0	67.2	72.7	14
	CVPR	SiamMask [144]	56.4	64.3	58.5	35
	CVPR	RVOS [86]	60.6	57.5	63.6	0.56
	ICCV	RANet [146]	65.7	63.2	68.2	30
	ICCV	DMM-Net [148]	70.7	68.1	73.3	0.37
	ICCV	DTN [149]	67.4	64.2	70.6	14.3
	ICCV	STM [150]	81.8	79.2	84.3	6.3
2020	NeurIPS	e-OSVOS [152]	77.2	74.4	80.0	0.5
	NeurIPS	Liang *et al.* [153]	74.6	73.0	76.1	4
	CVPR	Fasttan [154]	75.9	72.3	79.4	7
	CVPR	Fasttmu [155]	70.6	69.1	72.1	11
	CVPR	SAT [156]	72.3	68.6	76.0	39
	CVPR	FRTM-VOS [157]	76.7	-	-	22
	CVPR	TVOS [158]	72.3	69.9	74.7	37
	CVPR	Sun *et al.* [155]	69.9	69.1	70.6	11.1
	ECCV	GCNet [160]	71.4	69.3	73.5	25
	ECCV	KMN [161]	76.0	74.2	77.8	8.3
	ECCV	CFBI [162]	81.9	79.3	84.5	2.2
	ECCV	LWL [163]	70.8	68.2	73.5	14
	ECCV	MSN [164]	74.1	71.4	76.8	10
	ECCV	EGMN [105]	82.8	80.0	85.2	5
2021	CVPR	SwiftNet [167]	81.1	78.3	83.9	25
	CVPR	G-FRTM [168]	76.4	-	-	18.2
	CVPR	SST [169]	82.5	79.9	85.1	-
	CVPR	Ge *et al.* [170]	82.7	80.2	85.3	6.7
	CVPR	LCM [171]	83.5	80.5	86.5	8.5
	CVPR	RMNet [172]	83.5	81.0	86.0	12

TABLE 10
Quantitative IVOS results on the `val` set of DAVIS$_{17}$ [87] (§5.4.2) in terms of AUC and $\mathcal{J}$@60.

Year	Pub.	Method	AUC $\uparrow$	$\mathcal{J}$@60 $\uparrow$
2019	CVPR	Oh *et al.* [180]	69.1	73.4
2020	CVPR	MA-Net [181]	74.9	76.1
	ECCV	Heo *et al.* [182]	77.1	79.0
2021	CVPR	GIS [185]	82.0	82.9
	CVPR	MiVOS [186]	84.9	85.4

5.4.2 Results

DAVIS$_{17}$ [87] is also widely used for IVOS performance benchmarking. Results summarized in Table 10 show that the method proposed by Cheng *et al.* [186] is the top one.

TABLE 11
Quantitative LVOS results on the `test` set of A2D Sentence [190] (§5.5.2) in terms of Precision@K, mAP and IoU.

Year	Pub.	Method	Overlap					mAP↑	IoU		FPS↑
			P@0.5↑	P@0.6↑	P@0.7↑	P@0.8↑	P@0.9↑	0.5:0.95	Overall↑	Mean↑	
2018	CVPR	Gavrilyuk *et al.* [190]	50.0	37.6	23.1	9.4	0.4	21.5	55.1	42.6	-
2019	ICCV	ACGA [192]	55.7	45.9	31.9	16.0	2.0	27.4	60.1	49.0	9
2020	CVPR	VT-Capsule [195]	52.6	45.0	34.5	20.7	3.6	30.3	56.8	46.0	-
	AAAI	CMDy [193]	**60.7**	**52.5**	**40.5**	**23.5**	**4.5**	**33.3**	**62.3**	**53.1**	7
	IJCAI	Ning *et al.* [194]	**63.4**	**57.9**	**48.3**	**32.2**	**8.3**	**38.8**	**66.1**	**52.9**	5
2021	CVPR	Hui *et al.* [197]	**65.4**	**58.9**	**49.7**	**33.3**	**9.1**	**39.9**	**66.2**	**56.1**	-

TABLE 12
Quantitative VSS results on the `val` set of Cityscapes [202] (§5.6.2) in terms of IoU$_{\text{category}}$ and IoU$_{\text{class}}$. (Max Latency denotes *maximum per-frame time cost*.)

Year	Pub.	Method	IoU		FPS↑	Max Latency (ms)↓
			class↑	cate.↑		
2016	ECCV	Clockwork [201]	66.4	88.6	6.4	198
2017	ICCV	DFF [210]	69.2		5.6	575
	ICCV	PEARL [208]	75.4	89.2	1.3	800
	ICCV	NetWarp [209]	80.5	**91.0**	-	-
2018	CVPR	DVSN [214]	70.3	-	19.8	-
	CVPR	LVS [213]	76.8	89.8	5.8	380
	CVPR	GRFP [212]	**80.6**	**90.8**	3.9	255
2019	CVPR	Accel [217]	75.5	-	1.1	-
	CVPR	VPLR [218]	**81.4**	-	-	-
2020	CVPR	TDNet [220]	79.9	-	5.6	178
	AAAI	EFC [219]	**83.5**	**92.2**	-	-
2021	CVPR	Lukas [236]	71.2	-	-	-

5.5 LVOS Performance Benchmarking

5.5.1 Evaluation Metrics

As suggested by [190], IoU and precision are used to assert the validity of LVOS methods.

• **IoU**: *overall IoU* and *mean IoU* are both computed; for former is computed as total intersection area of all test data over the total union area, while the latter refers to average over the IoU of each test sample. The *overall IoU* is biased toward large segmented regions, while *mean IoU* treats large and small regions equally.

• **Precision**: Precision@K is computed as the percentage of testing samples whose IoU scores are higher than an overlap threshold K. Precision at five thresholds ranging from 0.5 to 0.9 as well as mean average precision (mAP) over 0.50:0.05:0.95 are reported.

5.5.2 Results

A2D Sentence [190] is arguably the most popular dataset in LVOS. Table 11 gives the results of six recent methods on A2D Sentence `test` set. It shows clear improvement trend from the first LVOS model [190] proposed in 2018, to recent complicated solution developed by Hui *et al.* [197].

5.6 VSS Performance Benchmarking

5.6.1 Evaluation Metrics

IoU matric is the most widely used metric in VSS. Moreover, in Cityscapes [202] – the gold-standard benchmark dataset in this field, two IoU scores, IoU$_{\text{category}}$ and IoU$_{\text{class}}$, defined over two semantic granularities, are reported. Here, 'category' refers to high-level semantic categories (*e.g.*, vehicle, human), while 'class' indicates more fine-grained semantic classes (*e.g.*, car, bicycle, person, rider). In total, [202] considers 19 classes, which are further grouped into 8 categories.

TABLE 13
Quantitative VIS results on the `val` set of YouTube-VIS [88] (§5.7.2) in terms of Precision@K, mAP, Recall@N and IoU.

Year	Pub.	Method	P@0.5↑	P@0.75↑	R@1↑	R@10↑	mAP↑ 0.5:0.95	FPS↑
2019	ICCV	MaskTrack R-CNN [88]	51.1	32.6	31.0	35.5	30.3	20
2020	ECCV	Sipmask [228]	53.0	33.3	33.5	38.9	32.5	24
	ECCV	STEm-Seg [229]	55.8	37.9	34.4	41.6	34.6	2
	CVPR	MaskProp [222]	-	**45.6**	-	-	**42.5**	-
2021	AAAI	CompFeat [231]	56.0	38.6	33.1	40.3	35.3	-
	CVPR	TraDeS [232]	52.6	32.8	29.1	36.6	32.6	26
	CVPR	SG-Net [233]	**57.1**	39.6	**35.9**	**43.0**	36.3	20
	CVPR	VisTR [234]	**64.0**	**45.0**	**38.3**	**44.9**	**40.1**	58
	-	Propose-Reduce [235]	**71.6**	**51.8**	**46.3**	**56.0**	**47.6**	-

5.6.2 Results

Table 12 summarizes the results of eleven VSS approaches on Cityscapes [202] `val` set. As seen, EFC [219] performs the best currently, with 83.5% in terms of IoU$_{\text{class}}$.

5.7 VIS Performance Benchmarking

5.7.1 Evaluation Metrics

As in [88], precision and recall metrics are used for VIS performance evaluation. Precision at IoU thresholds 0.5 and 0.75, as well as mean average precision (mAP) over 0.50:0.05:0.95 are reported. Recall@N is defined as the maximum recall given N segmented instances per video. These two metrics are first evaluated per category and then averaged over the category set. For the IoU metric, it is similar to region Jaccard $\mathcal{J}$ used in instance-level AVOS (§5.2.1).

5.7.2 Results

Table 13 gathers VIS results for on YouTube-VIS [88] `val` set. As seen, Transformer-based architecture, *i.e.*, VisTR [234], and redundant sequence proposal based solution, *i.e.*, Propose-Reduce [235], greatly improve the state-of-the-art.

5.8 VPS Performance Benchmarking

5.8.1 Evaluation Metrics

In [223], panoptic quality (PQ) metric used in image panoptic segmentation is modified as video panoptic quality (VPQ) to adapt to video panoptic segmentation.

• **VPQ**: Given a snippet $V^{t:t+k}$ with time window k, true positive (TP) is defined by TP$=\{(u,\hat{u})\in U\times\hat{U} : \text{IoU}(u,\hat{u})>0.5\}$ where U and $\hat{U}$ are the set of the ground-truth and predicted tubes, respectively. False Positives (FP) and False Nega- tives (FN) are defined accordingly. After accumulating the TP$_c$, FP$_c$, and FN$_c$ on all the clips with window size k and class c, VPQ is defined as:

$$\text{VPQ}^k = \frac{1}{N_{\text{class}}}\sum_c \frac{\sum_{(u,\hat{u})\in \text{TP}_c} \text{IoU}(u,\hat{u})}{|\text{TP}_c| + \frac{1}{2}|\text{FP}_c| + \frac{1}{2}|\text{FN}_c|}. \quad (4)$$

TABLE 14

Quantitative VPS results on the `test` set of Cityscapes-VPS [223] (§5.8.2) in term of VPQ. Each cell shows VPQ^k / VPQ^k-Thing / VPQ^k-Stuff.

Dataset	Year	Pub.	Method	Temporal window size				VPQ↑	FPS↑
				$k=0$↑	$k=5$↑	$k=10$↑	$k=15$↑		
Cityscapes-VPS `test`	2020	CVPR	FuseTrack [223]	64.2 / 59.0 / 67.7	57.9 / 46.5 / 65.1	54.8 / 41.1 / 63.4	52.6 / 36.5 / 62.9	57.4 / 45.8 / 64.8	1.3
	2021	CVPR	SiamTrack [237]	63.8 / 59.4 / 66.6	58.2 / 47.2 / 65.9	56.0 / 43.2 / 64.4	54.7 / 40.2 / 63.2	57.8 / 47.5 / 65.0	4.5
		CVPR	ViP-DeepLab [238]	68.9 / 61.6 / 73.5	62.9 / 51.0 / 70.5	59.9 / 46.0 / 68.8	58.2 / 42.1 / 68.4	62.5 / 50.2 / 70.3	-

When $k=1$, VPQ^1 is equivalent to PQ. For evaluation, VPQ^k is reported over different window sizes $k \in \{0,5,10,15\}$ and the final VPQ is given as VPQ$=\frac{1}{4}\sum_{k\in\{0,5,10,15\}}$ VPQ^k.

5.8.2 Results

Cityscapes-VPS [223] is chosen for testing VPS methods. As shown in Table 14, ViP-DeepLab [238] is the top one.

5.9 Summary

Based on the results, various conclusions can be drawn. The most important of them is related to reproducibility. Across different video segmentation areas, many methods do not describe the setup for the experimentation or do not provide the source code for implementation. Some of them even do not release segmentation masks. This makes fair comparison impossible and significantly hurts reproducibility.

Another important fact discovered thanks to this study is the lack of information about execution time and memory use. Many methods, particularly in the fields of AVOS, LVOS, and VPS, do not report execution time. Almost no paper reports memory use. This void is due to the fact that many methods focus only on accuracy without any concern about running time efficiency or memory requirements. However, in many application scenarios, such as mobile devices and self-driving cars, computational power and memory are typically limited. As benchmark datasets and challenges serve as a main driven factor behind the fast evolution of segmentation techniques, we encourage organizers of future video segmentation datasets or challenges to give this kind of metrics necessary importance in benchmarking.

Finally, performance on some extensively studied video segmentation datasets, such as DAVIS$_{16}$ [17] in AVOS, DAVIS$_{17}$ [87] in SVOS, A2D Sentence [190] in LVOS, tend to reach saturation. Though some new datasets are proposed recently and claim huge space for performance improvement, the dataset collectors just gather more and more challenging samples, without figuring out which exact challenges have been solved and which have not been.

6 Future Research Directions

Based on the reviewed research, we list several future research directions that would be interesting to pursue.

- **Long-Term Video Segmentation**: Long-term video segmentation is much closer to practical applications, such as video editing. However, as the sequences in existing datasets often span several seconds, the performance of VOS models over long video sequences (*e.g.*, at the minute level) are still unexamined. Bringing VOS in the long-term setting will unlock new research lines, and put forward higher demand of the re-detection capability of VOS models.
- **Open World Video Segmentation**: Despite the obvious dynamic and open nature of the world, current VSS algorithms are typically developed in a closed-world paradigm, where all categories are known as a prior. These algorithms are often brittle once exposed to the realistic complexity of the open world, where they are unable to efficiently adapt and robustly generalize to unseen categories. For example, practical deployments of VSS systems in robotics, self-driving cars and surveillance cannot afford to have complete knowledge on what classes to expect at inference time, while being trained in-house. This calls for much smarter video segmentation systems, with a strong capability to identify unknown categories/instances in their environments.
- **Cooperation across Different Video Segmentation Directions**: As the two main branches of video segmentation, VOS and VSS face many similar challenges, such as object occlusion, deformation, and fast motion. However, developments of these two closely related fields are relatively independent. Moreover, different VOS settings are also quite related, while there are no precedents for modeling these tasks in a unified framework. Thus we call for closer collaboration across different video segmentation sub-fields.
- **Annotation Efficient Video Segmentation Solutions**: Though great advance has been achieved in many videos segmentation tasks, current top-leading algorithms are built upon fully supervised deep learning techniques, requiring a huge amount of annotated data. Though semi-supervised, weakly supervised and unsupervised alternatives were explored in some literature, annotation efficient solutions receive far less attention and typically show weak performance, compared with the fully supervised ones. As the high temporal correlations in video data can provide additional cues for supervision, exploring existing annotation efficient techniques in static semantic segmentation in the area of video segmentation is an appealing direction.
- **Learning Dense Spatiotemporal Features from Unlabeled Videos**: Recent years have witnessed the compelling success of unsupervised visual representation learning – with massive unlabeled data, meaningful visual representations can be learned and greatly improve the performance on down-stream tasks, such as image classification, even with limited labeled samples. This sheds light on potential directions for learning dense spatiotemporal features from unlabeled videos, or noisy web videos. The learned representations can be used to better initialize video segmentation networks and facilitate annotation efficient learning.
- **Adaptive Computation**: It is widely recognized that there exist high correlations among video frames. Though such data redundancy and continuity are exploited to reduce the computation cost in VSS, almost all current video segmentation models are fixed feed-forward structures or work alternatively between heavy and light-weight modes. We

expect more flexible segmentation model designs towards more efficient and adaptive computation [260], which allows network architecture change on-the-fly – selectively activating part of the network in an input-dependent fashion.

• **Neural Architecture Search**: Video segmentation models are typically built upon hand-designed architectures, which may be suboptimal for capturing the nature of video data and limit the best possible performance. Using neural architecture search techniques to automate the design of video segmentation networks will be a promising direction.

7 CONCLUSION

As far as we know, this is the first survey to comprehensively review recent progress in video segmentation, covering several sub-fields: AVOS, SVOS, IVOS, LVOS, VSS, VIS, and VPS. We provided the reader with the necessary background knowledge and summarized more than 150 deep learning models according to various criteria, including task settings, technique contributions, and learning strategies. We also presented a structured survey of 20 widely used video segmentation datasets and benchmarking results on 7 most ones. In the end, we discussed the results and provided useful insight in shape of future research directions and open problems in the field. In conclusion, video segmentation has achieved notable progress thanks to the striking development of deep learning techniques, but several challenges still lie ahead. We expect this survey to provide an effective way to understand current state-of-the-arts and speed up the development of this research field.

REFERENCES

[1] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, 2017.

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

[3] C. Xu and J. J. Corso, "Evaluation of super-voxel methods for early video processing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1202–1209.

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

[5] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1995–2002.

[6] C.-P. Yu, H. Le, G. Zelinsky, and D. Samaras, "Efficient video segmentation using parametric graph partitioning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3155–3163.

[7] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2141–2148.

[8] N. S. Nagaraja, F. R. Schmidt, and T. Brox, "Video segmentation with just a few strokes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3235–3243.

[9] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3227–3234.

[10] V. Badrinarayanan, I. Budvytis, and R. Cipolla, "Semi-supervised video segmentation using tree structured graphical models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2751–2764, 2013.

[11] W.-D. Jang and C.-S. Kim, "Streaming video segmentation via short-term hierarchical segmentation and frame-by-frame markov random field optimization," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 599–615.

[12] B. Liu and X. He, "Multiclass semantic video segmentation with object-level active inference," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4286–4294.

[13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[14] D. M. Thounaojam, A. Trivedi, K. M. Singh, and S. Roy, "A survey on video segmentation," in *Intelligent computing, networking, and informatics*, 2014, pp. 903–912.

[15] Y.-J. Zhang, "An overview of image and video segmentation in the last 40 years," *Advances in Image and Video Segmentation*, pp. 1–16, 2006.

[16] R. Yao, G. Lin, S. Xia, J. Zhao, and Y. Zhou, "Video object segmentation and tracking: A survey," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 4, pp. 1–47, 2020.

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

[18] S. Caelles, K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. V. Gool, "One-shot video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5320–5329.

[19] V. Badrinarayanan, F. Galasso, and R. Cipolla, "Label propagation in video sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3265–3272.

[20] H. Shen, "Towards a mathematical understanding of the difficulty in learning with feedforward neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 811–820.

[21] L. G. Roberts, "Machine perception of three-dimensional solids," *Optical and Electro-Optical Information Processing*, 1965.

[22] J. Chang, D. Wei, and J. W. Fisher, "A video representation using temporal superpixels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2051–2058.

[23] F. Perazzi, O. Wang, M. H. Gross, and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3227–3234.

[24] R. Jain and H.-H. Nagel, "On the analysis of accumulative difference pictures from image sequences of real world scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 2, pp. 206–214, 1979.

[25] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, 1997.

[26] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov, "Bilayer segmentation of live video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 1, 2006, pp. 53–60.

[27] S. Brutzer, B. Höferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1937–1944.

[28] E. Hayman and J.-O. Eklundh, "Statistical background subtraction for a mobile observer," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, 2003, pp. 67–67.

[29] M. Irani and P. Anandan, "A unified approach to moving object detection in 2d and 3d scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 6, pp. 577–589, 1998.

[30] C. Yuan, G. Medioni, J. Kang, and I. Cohen, "Detecting motion regions in the presence of a strong parallax from a moving camera by multiview geometric constraints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1627–1641, 2007.

[31] J. Y. Wang and E. H. Adelson, "Layered representation for motion analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1993, pp. 361–366.

[32] H. S. Sawhney and S. Ayer, "Compact representations of videos through dominant and multiple motion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 814–830, 1996.

[33] J. Costeira and T. Kanade, "A multi-body factorization method for motion analysis," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1995, pp. 1071–1076.

[34] D. Cremers and S. Soatto, "Motion competition: A variational approach to piecewise parametric motion segmentation," *International Journal of Computer Vision*, vol. 62, no. 3, pp. 249–265, 2005.

[35] P. Ochs and T. Brox, "Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1583–1590.

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

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

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

[39] A. Faktor and M. Irani, "Video segmentation by non-local consensus voting," in *Proceedings of the British Machine Vision Conference*, 2014.

[40] T. Ma and L. J. Latecki, "Maximum weight cliques with mutex constraints for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 670–677.

[41] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 628–635.

[42] F. Xiao and Y. Jae Lee, "Track and segment: An iterative unsupervised approach for video object proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 933–942.

[43] I. Endres and D. Hoiem, "Category independent object proposals," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 575–588.

[44] S. Avinash Ramakanth and R. Venkatesh Babu, "SeamSeg: Video object segmentation using patch seams," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 376–383.

[45] N. Shankar Nagaraja, F. R. Schmidt, and T. Brox, "Video segmentation with just a few strokes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3235–3243.

[46] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3899–3908.

[47] W. Wang, J. Shen, F. Porikli, and R. Yang, "Semi-supervised video object segmentation with super-trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 985–998, 2018.

[48] D. Tsai, M. Flagg, and J. M. Rehg, "Motion coherent tracking using multi-label MRF optimization," *Proceedings of the British Machine Vision Conference*, 2010.

[49] L. Wen, D. Du, Z. Lei, S. Z. Li, and M.-H. Yang, "JOTS: Joint online tracking and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2226–2234.

[50] A. Agarwala, A. Hertzmann, D. H. Salesin, and S. M. Seitz, "Keyframe-based tracking for rotoscoping and animation," *ACM Tran. Graphics*, vol. 23, no. 3, pp. 584–591, 2004.

[51] W. Li, F. Viola, J. Starck, G. J. Brostow, and N. D. Campbell, "Roto++ accelerating professional rotoscoping using shape manifolds," *ACM Tran. Graphics*, vol. 35, no. 4, pp. 1–15, 2016.

[52] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video SnapCut: robust video object cutout using localized classifiers," *ACM Tran. Graphics*, vol. 28, no. 3, p. 70, 2009.

[53] A. Criminisi, T. Sharp, C. Rother, and P. Pérez, "Geodesic image and video editing," *ACM Tran. Graphics*, vol. 29, no. 5, pp. 134–1, 2010.

[54] F. Zhong, X. Qin, Q. Peng, and X. Meng, "Discontinuity-aware video object cutout," *ACM Tran. Graphics*, vol. 31, no. 6, p. 175, 2012.

[55] Q. Fan, F. Zhong, D. Lischinski, D. Cohen-Or, and B. Chen, "JumpCut: non-successive mask transfer and interpolation for video cutout." *ACM Tran. Graphics*, vol. 34, no. 6, pp. 195–1, 2015.

[56] Y. Lu, X. Bai, L. Shapiro, and J. Wang, "Coherent parametric contours for interactive video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 642–650.

[57] W. Wang, J. Shen, and F. Porikli, "Selective video object cutout," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5645–5655, 2017.

[58] A. Jain, S. Chatterjee, and R. Vidal, "Coarse-to-fine semantic video segmentation using supervoxel trees," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1865–1872.

[59] A. Kae, B. Marlin, and E. Learned-Miller, "The shape-time random field for semantic video labeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 272–279.

[60] G. Hartmann, M. Grundmann, J. Hoffman, D. Tsai, V. Kwatra, O. Madani, S. Vijayanarasimhan, I. Essa, J. Rehg, and R. Sukthankar, "Weakly supervised learning of object segmentations from web-scale video," in *European Conference on Computer Vision Workshop*, 2012, pp. 198–208.

[61] K. Tang, R. Sukthankar, J. Yagnik, and L. Fei-Fei, "Discriminative segment annotation in weakly labeled video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2483–2490.

[62] X. Liu, D. Tao, M. Song, Y. Ruan, C. Chen, and J. Bu, "Weakly supervised multiclass video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 57–64.

[63] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, 2013.

[64] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1–45, 2006.

[65] C. Bibby and I. Reid, "Robust real-time visual tracking using pixel-wise posteriors," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 831–844.

[66] X. Ren and J. Malik, "Tracking as repeated figure/ground segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[67] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Applied Soft Computing*, vol. 70, pp. 41–65, 2018.

[68] L. Jiao, R. Zhang, F. Liu, S. Yang, B. Hou, L. Li, and X. Tang, "New generation deep learning for video object detection: A survey," *IEEE Trans. Neural Netw. Learning Sys.*, 2021.

[69] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 38–49, 2017.

[70] D.-P. Fan, W. Wang, M.-M. Cheng, and J. Shen, "Shifting more attention to video salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8554–8564.

[71] W. Wang, J. Shen, X. Lu, S. C. Hoi, and H. Ling, "Paying attention to video object pattern understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[72] S. D. Jain, B. Xiong, and K. Grauman, "Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 686–695.

[73] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[74] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang, "Segflow: Joint learning for video object segmentation and optical flow," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 686–695.

[75] P. Tokmakov, K. Alahari, and C. Schmid, "Learning video object segmentation with visual memory," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4491–4500.

[76] ——, "Learning motion patterns in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 531–539.

[77] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4040–4048.

[78] J. Li, A. Zheng, X. Chen, and B. Zhou, "Primary video object segmentation via complementary cnns and neighborhood reversible flow," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1417–1425.

[79] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3282–3289.

[80] S. Li, B. Seybold, A. Vorobyov, A. Fathi, Q. Huang, and C.-C. Jay Kuo, "Instance embedding transfer to unsupervised video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6526–6535.

[81] G. Li, Y. Xie, T. Wei, K. Wang, and L. Lin, "Flow guided recurrent neural encoder for video salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3243–3252.

[82] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, "Video segmentation by tracking many figure-ground segments," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2192–2199.

[83] S. Li, B. Seybold, A. Vorobyov, X. Lei, and C.-C. Jay Kuo, "Unsupervised video object segmentation with motion-based bilateral networks," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 215–231.

[84] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 744–760.

[85] M. Siam, C. Jiang, S. Lu, L. Petrich, M. Gamal, M. Elhoseiny, and M. Jagersand, "Video segmentation using teacher-student adaptation in a human robot interaction (hri) setting," in *International Conference on Robotics and Automation*, 2019, pp. 50–56.

[86] C. Ventura, M. Bellver, A. Girbau, A. Salvador, F. Marques, and X. Giro-i Nieto, "Rvos: End-to-end recurrent network for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5277–5286.

[87] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 davis challenge on video object segmentation," *arXiv preprint arXiv:1704.00675*, 2017.

[88] L. Yang, Y. Fan, and N. Xu, "Video instance segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 5188–5197.

[89] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, "See more, know more: Unsupervised video object segmentation with co-attention siamese networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3623–3632.

[90] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2015.

[91] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3166–3173.

[92] Y. Yang, A. Loquercio, D. Scaramuzza, and S. Soatto, "Unsupervised moving object detection via contextual information separation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 879–888.

[93] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S. C. Hoi, and H. Ling, "Learning unsupervised video object segmentation through visual attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3064–3074.

[94] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 280–287.

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

[96] H. Li, G. Chen, G. Li, and Y. Yu, "Motion guided attention for video salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7274–7283.

[97] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 136–145.

[98] Z. Yang, Q. Wang, L. Bertinetto, W. Hu, S. Bai, and P. H. Torr, "Anchor diffusion for unsupervised video object segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 931–940.

[99] P. Tokmakov, C. Schmid, and K. Alahari, "Learning to segment moving objects," *Int. J. Comput. Vis.*, vol. 127, no. 3, pp. 282–301, 2019.

[100] T. Zhou, S. Wang, Y. Zhou, Y. Yao, J. Li, and L. Shao, "Motion-attentive transition for zero-shot video object segmentation," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 13 066–13 073.

[101] N. Xu, L. Yang, Y. Fan, D. Yue, Y. Liang, J. Yang, and T. Huang, "Youtube-vos: A large-scale video object segmentation benchmark," *arXiv preprint arXiv:1809.03327*, 2018.

[102] Y. Gu, L. Wang, Z. Wang, Y. Liu, M.-M. Cheng, and S.-P. Lu, "Pyramid constrained self-attention network for fast video salient object detection," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 10 869–10 876.

[103] X. Lu, W. Wang, J. Shen, Y.-W. Tai, D. J. Crandall, and S. C. Hoi, "Learning video object segmentation from unlabeled videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8960–8970.

[104] J. Valmadre, L. Bertinetto, J. F. Henriques, R. Tao, A. Vedaldi, A. W. Smeulders, P. H. Torr, and E. Gavves, "Long-term tracking in the wild: A benchmark," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 670–685.

[105] X. Lu, W. Wang, M. Danelljan, T. Zhou, J. Shen, and L. Van Gool, "Video object segmentation with episodic graph memory networks," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[106] L. Zhang, J. Zhang, Z. Lin, R. Mech, H. Lu, and Y. He, "Unsupervised video object segmentation with joint hotspot tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[107] M. Jiang, S. Huang, J. Duan, and Q. Zhao, "Salicon: Saliency in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1072–1080.

[108] M. Everingham, S. M. A. Eslami, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, 2015.

[109] M. Zhen, S. Li, L. Zhou, J. Shang, H. Feng, T. Fang, and L. Quan, "Learning discriminative feature with crf for unsupervised video object segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 445–462.

[110] D. Liu, D. Yu, C. Wang, and P. Zhou, "F2net: Learning to focus on the foreground for unsupervised video object segmentation," in *AAAI Conference on Artificial Intelligence*, 2021.

[111] T. Zhou, J. Li, X. Li, and L. Shao, "Target-aware object discovery and association for unsupervised video multi-object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[112] S. Ren, W. Liu, Y. Liu, H. Chen, G. Han, and S. He, "Reciprocal transformations for unsupervised video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15 455–15 464.

[113] Y. Yang, B. Lai, and S. Soatto, "Dystab: Unsupervised object segmentation via dynamic-static bootstrapping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2826–2836.

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

[115] Y. Tsai, M. Yang, and M. J. Black, "Video segmentation via object flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3899–3908.

[116] A. Fathi, Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy, "Semantic instance segmentation via deep metric learning," *arXiv preprint arXiv:1703.10277*, 2017.

[117] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proc. Advances Neural Inf. Process. Syst*, 2015, pp. 2440–2448.

[118] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung, "Learning video object segmentation from static images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3491–3500.

[119] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1155–1162.

[120] W.-D. Jang and C.-S. Kim, "Online video object segmentation via convolutional trident network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5849–5858.

[121] V. Jampani, R. Gadde, and P. V. Gehler, "Video propagation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3154–3164.

[122] J. S. Yoon, F. Rameau, J. Kim, S. Lee, S. Shin, and I. S. Kweon, "Pixel-level matching for video object segmentation using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2186–2195.

[123] P. Voigtlaender and B. Leibe, "Online adaptation of convolutional neural networks for video object segmentation," in *Proceedings of the British Machine Vision Conference*, 2017.

[124] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[125] A. Khoreva, R. Benenson, E. Ilg, T. Brox, and B. Schiele, "Lucid data dreaming for video object segmentation," *International Journal of Computer Vision*, vol. 127, no. 9, pp. 1175–1197, 2019.

[126] L. Bao, B. Wu, and W. Liu, "CNN in MRF: Video object segmentation via inference in a CNN-based higher-order spatio-temporal MRF," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5977–5986.

[127] J. Cheng, Y.-H. Tsai, W.-C. Hung, S. Wang, and M.-H. Yang, "Fast and accurate online video object segmentation via tracking parts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7415–7424.

[128] S. Wug Oh, J.-Y. Lee, K. Sunkavalli, and S. Joo Kim, "Fast video object segmentation by reference-guided mask propagation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7376–7385.

[129] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos, "Efficient video object segmentation via network modulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6499–6507.

[130] H. Xiao, J. Feng, G. Lin, Y. Liu, and M. Zhang, “Monet: Deep motion exploitation for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1140–1148.

[131] P. Hu, G. Wang, X. Kong, J. Kuen, and Y.-P. Tan, “Motion-guided cascaded refinement network for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1400–1409.

[132] J. Han, L. Yang, D. Zhang, X. Chang, and X. Liang, “Reinforcement cutting-agent learning for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9080–9089.

[133] K.-K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool, “Video object segmentation without temporal information,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1515–1530, 2018.

[134] Y.-T. Hu, J.-B. Huang, and A. G. Schwing, “Videomatch: Matching based video object segmentation,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 56–73.

[135] X. Li and C. C. Loy, “Video object segmentation with joint re-identification and attention-aware mask propagation,” in *Proc. Eur. Conf. Comput. Vis.*, 2018.

[136] H. Ci, C. Wang, and Y. Wang, “Video object segmentation by learning location-sensitive embeddings,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 524–539.

[137] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, “Tracking emerges by colorizing videos,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 391–408.

[138] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, “The kinetics human action video dataset,” *arXiv preprint arXiv:1705.06950*, 2017.

[139] H. Xiao, B. Kang, Y. Liu, M. Zhang, and J. Feng, “Online meta adaptation for fast video object segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1205–1217, 2019.

[140] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen, “Feelvos: Fast end-to-end embedding learning for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9481–9490.

[141] S. Xu, D. Liu, L. Bao, W. Liu, and P. Zhou, “Mhp-vos: Multiple hypotheses propagation for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 314–323.

[142] H. Lin, X. Qi, and J. Jia, “Agss-vos: Attention guided single-shot video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3949–3957.

[143] J. Johnander, M. Danelljan, E. Brissman, F. S. Khan, and M. Felsberg, “A generative appearance model for end-to-end video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8953–8962.

[144] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr, “Fast online object tracking and segmentation: A unifying approach,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1328–1338.

[145] B. A. Griffin and J. J. Corso, “Bubblenets: Learning to select the guidance frame in video object segmentation by deep sorting frames,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8914–8923.

[146] Z. Wang, J. Xu, L. Liu, F. Zhu, and L. Shao, “Ranet: Ranking attention network for fast video object segmentation,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3978–3987.

[147] G. Li and Y. Yu, “Visual saliency based on multiscale deep features,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.

[148] X. Zeng, R. Liao, L. Gu, Y. Xiong, S. Fidler, and R. Urtasun, “Dmm-net: Differentiable mask-matching network for video object segmentation,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3929–3938.

[149] L. Zhang, Z. Lin, J. Zhang, H. Lu, and Y. He, “Fast video object segmentation via dynamic targeting network,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 5582–5591.

[150] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, “Video object segmentation using space-time memory networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9226–9235.

[151] D. F. Fouhey, W.-c. Kuo, A. A. Efros, and J. Malik, “From lifestyle vlogs to everyday interactions,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4991–5000.

[152] T. Meinhardt and L. Leal-Taixé, “Make one-shot video object segmentation efficient again,” in *Proc. Advances Neural Inf. Process. Syst*, 2020.

[153] Y. Liang, X. Li, N. Jafari, and Q. Chen, “Video object segmentation with adaptive feature bank and uncertain-region refinement,” in *Proc. Advances Neural Inf. Process. Syst*, 2020.

[154] X. Huang, J. Xu, Y.-W. Tai, and C.-K. Tang, “Fast video object segmentation with temporal aggregation network and dynamic template matching,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8879–8889.

[155] M. Sun, J. Xiao, E. G. Lim, B. Zhang, and Y. Zhao, “Fast template matching and update for video object tracking and segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 791–10 799.

[156] X. Chen, Z. Li, Y. Yuan, G. Yu, J. Shen, and D. Qi, “State-aware tracker for real-time video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9384–9393.

[157] A. Robinson, F. J. Lawin, M. Danelljan, F. S. Khan, and M. Felsberg, “Learning fast and robust target models for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7406–7415.

[158] Y. Zhang, Z. Wu, H. Peng, and S. Lin, “A transductive approach for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6949–6958.

[159] Z. Lai, E. Lu, and W. Xie, “Mast: A memory-augmented self-supervised tracker,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6479–6488.

[160] Y. Li, Z. Shen, and Y. Shan, “Fast video object segmentation using the global context module,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 735–750.

[161] H. Seong, J. Hyun, and E. Kim, “Kernelized memory network for video object segmentation,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 629–645.

[162] Z. Yang, Y. Wei, and Y. Yang, “Collaborative video object segmentation by foreground-background integration,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 332–348.

[163] G. Bhat, F. J. Lawin, M. Danelljan, A. Robinson, M. Felsberg, L. Van Gool, and R. Timofte, “Learning what to learn for video object segmentation,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 777–794.

[164] R. Wu, H. Lin, X. Qi, and J. Jia, “Memory selection network for video propagation,” in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 175–190.

[165] Y. Li, N. Xu, J. Peng, J. See, and W. Lin, “Delving into the cyclic mechanism in semi-supervised video object segmentation,” in *Proc. Advances Neural Inf. Process. Syst*, 2020.

[166] F. Lin, H. Xie, Y. Li, and Y. Zhang, “Query-memory re-aggregation for weakly-supervised video object segmentation,” in *AAAI Conference on Artificial Intelligence*, 2021.

[167] H. Wang, X. Jiang, H. Ren, Y. Hu, and S. Bai, “Swiftnet: Real-time video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1296–1305.

[168] H. Park, J. Yoo, S. Jeong, G. Venkatesh, and N. Kwak, “Learning dynamic network using a reuse gate function in semi-supervised video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8405–8414.

[169] B. Duke, A. Ahmed, C. Wolf, P. Aarabi, and G. W. Taylor, “Sstvos: Sparse spatiotemporal transformers for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[170] W. Ge, X. Lu, and J. Shen, “Video object segmentation using global and instance embedding learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16 836–16 845.

[171] L. Hu, P. Zhang, B. Zhang, P. Pan, Y. Xu, and R. Jin, “Learning position and target consistency for memory-based video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4144–4154.

[172] H. Xie, H. Yao, S. Zhou, S. Zhang, and W. Sun, “Efficient regional memory network for video object segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1286–1295.

[173] W. Wang, X. Lu, J. Shen, D. J. Crandall, and L. Shao, “Zero-shot video object segmentation via attentive graph neural networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9236–9245.

[174] X. Lu, W. Wang, J. Shen, D. Crandall, and J. Luo, “Zero-shot video object segmentation with co-attention siamese networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[175] S. Caelles, J. Pont-Tuset, F. Perazzi, A. Montes, K.-K. Maninis, and L. Van Gool, "The 2019 davis challenge on vos: Unsupervised multi-object segmentation," *arXiv:1905.00737*, 2019.

[176] J. Luiten, I. E. Zulfikar, and B. Leibe, "Unovost: Unsupervised offline video object segmentation and tracking," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 2000–2009.

[177] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2961–2969.

[178] A. Benard and M. Gygli, "Interactive video object segmentation in the wild," *arXiv preprint arXiv:1801.00269*, 2017.

[179] Y. Chen, J. Pont-Tuset, A. Montes, and L. Van Gool, "Blazingly fast video object segmentation with pixel-wise metric learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1189–1198.

[180] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, "Fast user-guided video object segmentation by interaction-and-propagation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5247–5256.

[181] J. Miao, Y. Wei, and Y. Yang, "Memory aggregation networks for efficient interactive video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 366–10 375.

[182] Y. Heo, Y. J. Koh, and C.-S. Kim, "Interactive video object segmentation using global and local transfer modules," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[183] B. Chen, H. Ling, X. Zeng, G. Jun, Z. Xu, and S. Fidler, "Scribblebox: Interactive annotation framework for video object segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[184] Z. Yin, J. Zheng, W. Luo, S. Qian, H. Zhang, and S. Gao, "Learning to recommend frame for interactive video object segmentation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15 445–15 454.

[185] Y. Heo, Y. J. Koh, and C.-S. Kim, "Guided interactive video object segmentation using reliability-based attention maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7322–7330.

[186] H. K. Cheng, Y.-W. Tai, and C.-K. Tang, "Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5559–5568.

[187] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3119–3127.

[188] K. Duarte, Y. S. Rawat, and M. Shah, "Capsulevos: Semi-supervised video object segmentation using capsule routing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8480–8489.

[189] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Advances Neural Inf. Process. Syst*, 2017.

[190] K. Gavrilyuk, A. Ghodrati, Z. Li, and C. G. Snoek, "Actor and action video segmentation from a sentence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5958–5966.

[191] A. Khoreva, A. Rohrbach, and B. Schiele, "Video object segmentation with language referring expressions," in *Asian Conference on Computer Vision*, 2018, pp. 123–141.

[192] H. Wang, C. Deng, J. Yan, and D. Tao, "Asymmetric cross-guided attention network for actor and action video segmentation from natural language query," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3939–3948.

[193] H. Wang, C. Deng, F. Ma, and Y. Yang, "Context modulated dynamic networks for actor and action video segmentation with language queries," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 12 152–12 159.

[194] K. Ning, L. Xie, F. Wu, and Q. Tian, "Polar relative positional encoding for video-language segmentation," in *IJCAI*, 2020.

[195] B. McIntosh, K. Duarte, Y. S. Rawat, and M. Shah, "Visual-textual capsule routing for text-based video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9942–9951.

[196] S. Seo, J.-Y. Lee, and B. Han, "Urvos: Unified referring video object segmentation network with a large-scale benchmark," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[197] T. Hui, S. Huang, S. Liu, Z. Ding, G. Li, W. Wang, J. Han, and F. Wang, "Collaborative spatial-temporal modeling for language-queried video actor segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[198] L. Ye, M. Rochan, Z. Liu, X. Zhang, and Y. Wang, "Referring segmentation in images and videos with cross-modal self-attention network," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[199] S. Caelles, A. Montes, K.-K. Maninis, Y. Chen, L. Van Gool, F. Perazzi, and J. Pont-Tuset, "The 2018 davis challenge on video object segmentation," *arXiv preprint arXiv:1803.00557*, 2018.

[200] N. Xu, B. Price, S. Cohen, J. Yang, and T. S. Huang, "Deep interactive object selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 373–381.

[201] E. Shelhamer, K. Rakelly, J. Hoffman, and T. Darrell, "Clockwork convnets for video semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 852–868.

[202] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[203] A. Kundu, V. Vineet, and V. Koltun, "Feature space optimization for semantic video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3168–3175.

[204] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[205] J. Hur and S. Roth, "Joint optical flow and temporally consistent semantic segmentation," in *European Conference on Computer Vision*, 2016, pp. 163–177.

[206] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar, A. Geiger, and B. Leibe, "Mots: Multi-object tracking and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7942–7951.

[207] B. Mahasseni, S. Todorovic, and A. Fern, "Budget-aware deep semantic video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1029–1038.

[208] X. Jin, X. Li, H. Xiao, X. Shen, Z. Lin, J. Yang, Y. Chen, J. Dong, L. Liu, Z. Jie *et al.*, "Video scene parsing with predictive feature learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5580–5588.

[209] R. Gadde, V. Jampani, and P. V. Gehler, "Semantic video cnns through representation warping," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4453–4462.

[210] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2349–2358.

[211] F. S. Saleh, M. S. Aliakbarian, M. Salzmann, L. Petersson, and J. M. Alvarez, "Bringing background into the foreground: Making all classes equal in weakly-supervised video semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2125–2135.

[212] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6819–6828.

[213] Y. Li, J. Shi, and D. Lin, "Low-latency video semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5997–6005.

[214] Y.-S. Xu, T.-J. Fu, H.-K. Yang, and C.-Y. Lee, "Dynamic video segmentation network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6556–6565.

[215] P.-Y. Huang, W.-T. Hsu, C.-Y. Chiu, T.-F. Wu, and M. Sun, "Efficient uncertainty estimation for semantic segmentation in videos," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 520–535.

[216] S. Chandra, C. Couprie, and I. Kokkinos, "Deep spatio-temporal random fields for efficient video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8915–8924.

[217] S. Jain, X. Wang, and J. E. Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8866–8875.

[218] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro, "Improving semantic segmentation via video propagation and label relaxation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8856–8865.

[219] M. Ding, Z. Wang, B. Zhou, J. Shi, Z. Lu, and P. Luo, "Every frame counts: joint learning of video segmentation and optical flow," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 10 713–10 720.

[220] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, and F. Perazzi, "Temporally distributed networks for fast video semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8818–8827.

[221] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[222] G. Bertasius and L. Torresani, "Classifying, segmenting, and tracking object instances in video with mask propagation," in

Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9739–9748.

[223] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Video panoptic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9859–9868.

[224] L. Porzi, M. Hofinger, I. Ruiz, J. Serrat, S. R. Bulo, and P. Kontschieder, "Learning multi-object tracking and segmentation from automatic annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6846–6855.

[225] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2636–2645.

[226] C.-C. Lin, Y. Hung, R. Feris, and L. He, "Video instance segmentation tracking with a modified vae architecture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13 147–13 157.

[227] Y. Liu, C. Shen, C. Yu, and J. Wang, "Efficient semantic video segmentation with per-frame inference," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 352–368.

[228] J. Cao, R. M. Anwer, H. Cholakkal, F. S. Khan, Y. Pang, and L. Shao, "Sipmask: Spatial information preservation for fast image and video instance segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020.

[229] A. Athar, S. Mahadevan, A. Ošep, L. Leal-Taixé, and B. Leibe, "Stem-seg: Spatio-temporal embeddings for instance segmentation in videos," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 158–177.

[230] L.-C. Chen, R. G. Lopes, B. Cheng, M. D. Collins, E. D. Cubuk, B. Zoph, H. Adam, and J. Shlens, "Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 695–714.

[231] Y. Fu, L. Yang, D. Liu, T. S. Huang, and H. Shi, "Compfeat: Comprehensive feature aggregation for video instance segmentation," in *AAAI Conference on Artificial Intelligence*, 2021.

[232] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, and J. Yuan, "Track to detect and segment: An online multi-object tracker," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[233] D. Liu, Y. Cui, W. Tan, and Y. Chen, "Sg-net: Spatial granularity network for one-stage video instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[234] Y. Wang, Z. Xu, X. Wang, C. Shen, B. Cheng, H. Shen, and H. Xia, "End-to-end video instance segmentation with transformers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[235] H. Lin, R. Wu, S. Liu, J. Lu, and J. Jia, "Video instance segmentation with a propose-reduce paradigm," *arXiv preprint arXiv:2103.13746*, 2021.

[236] L. Hoyer, D. Dai, Y. Chen, A. Koring, S. Saha, and L. Van Gool, "Three ways to improve semantic segmentation with self-supervised depth estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11 130–11 140.

[237] S. Woo, D. Kim, J.-Y. Lee, and I. S. Kweon, "Learning to associate every segment for video panoptic segmentation," in *CVPR*, 2021, pp. 2705–2714.

[238] S. Qiao, Y. Zhu, H. Adam, A. Yuille, and L.-C. Chen, "Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[239] Q. Liu, V. Ramanathan, D. Mahajan, A. Yuille, and Z. Yang, "Weakly supervised instance segmentation for videos with temporal mask consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[240] Y. Fu, S. Liu, U. Iqbal, S. De Mello, H. Shi, and J. Kautz, "Learning to track instances without video annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[241] M. Yamaguchi, K. Saito, Y. Ushiku, and T. Harada, "Spatio-temporal person retrieval via natural language queries," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1453–1462.

[242] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "Tall: Temporal activity localization via language query," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5267–5275.

[243] L. Anne Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with natural language," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5803–5812.

[244] Z. Li, R. Tao, E. Gavves, C. G. Snoek, and A. W. Smeulders, "Tracking by natural language specification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6495–6503.

[245] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with em routing," in *Proc. Int. Conf. Learn. Representations*, 2018.

[246] S. Liu, T. Hui, S. Huang, Y. Wei, B. Li, and G. Li, "Cross-modal progressive comprehension for referring segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[247] J. Miao, Y. Wei, Y. Wu, C. Liang, G. Li, and Y. Yang, "Vspw: A large-scale dataset for video scene parsing in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021.

[248] J. Qi, Y. Gao, Y. Hu, X. Wang, X. Liu, X. Bai, S. Belongie, A. Yuille, P. Torr, and S. Bai, "Occluded video instance segmentation," *arXiv preprint arXiv:2102.01558*, 2021.

[249] A. Hu, A. Kendall, and R. Cipolla, "Learning a spatio-temporal embedding for video instance segmentation," in *Proc. Int. Conf. Learn. Representations*, 2019.

[250] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, and R. Urtasun, "Upsnet: A unified panoptic segmentation network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8818–8826.

[251] S. D. Jain and K. Grauman, "Supervoxel-consistent foreground propagation in video," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 656–671.

[252] S. Caelles, A. Montes, K.-K. Maninis, Y. Chen, L. Van Gool, F. Perazzi, and J. Pont-Tuset, "The 2018 davis challenge on video object segmentation," *arXiv preprint arXiv:1803.00557*, 2018.

[253] C. Xu, S.-H. Hsieh, C. Xiong, and J. J. Corso, "Can humans fly? action understanding with multiple classes of actors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 2264–2273.

[254] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 3192–3199.

[255] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[256] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[257] S. R. Richter, Z. Hayder, and V. Koltun, "Playing for benchmarks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2232–2241.

[258] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, 2004.

[259] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 4, pp. 509–522, 2002.

[260] E. Bengio, P.-L. Bacon, J. Pineau, and D. Precup, "Conditional computation in neural networks for faster models," in *Proc. Int. Conf. Learn. Representations*, 2016.