JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

MobileVOS: Real-Time Video Object Segmentation

Contrastive Learning meets Knowledge Distillation

Roy Miles, Mehmet Kerim Yucel, Bruno Manganelli, Albert Saa-Garriga

Samsung Research UK

Samsung Research

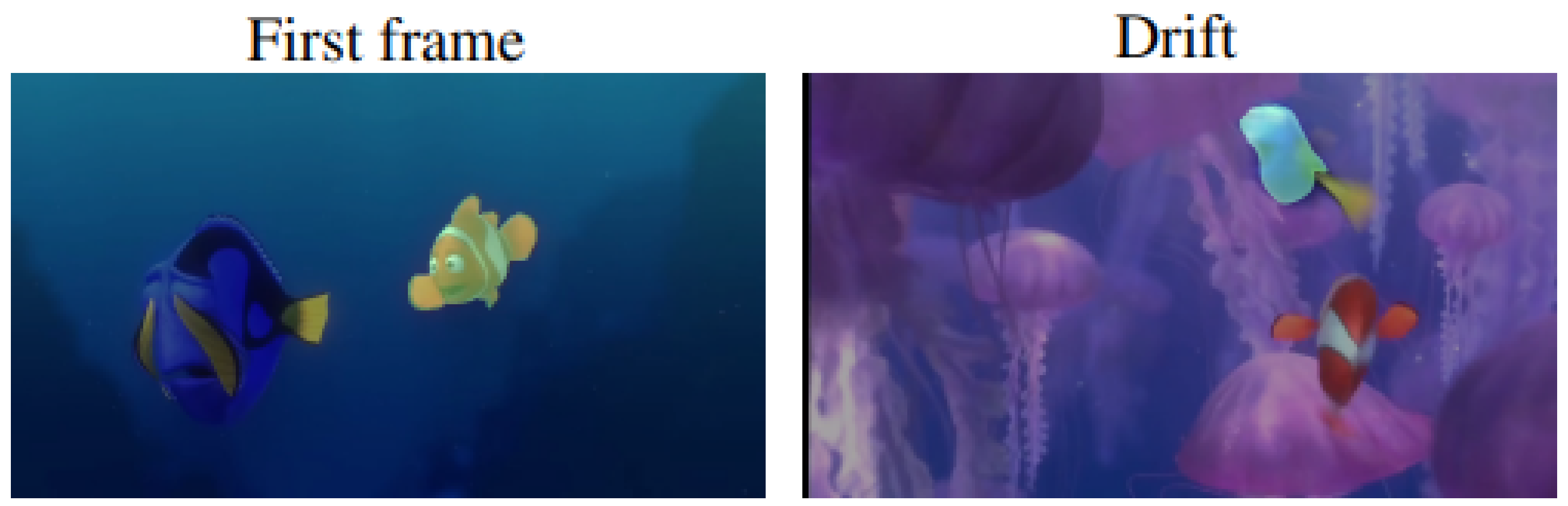
Background

- Video Object Segmentation (VOS) is a crucial aspect of computer vision, and it has been applied across numerous applications such as video editing, surveillance, autonomous driving, and augmented reality.
- VOS involves the identification and tracking of objects across multiple frames in a video sequence.
- Our work is primarily on the topic of Semi-supervised Video Object Segmentation (SVOS), a scenario in which only the initial frame provides an object description.

The difficulty of SVOS lies in being able to track an object under difference views and severe occlusions, while similarly performing in a class agnostic manner.

Limitations

The current state-of-the-art relies on **space-time-memory networks** (STM), which relies on densely matching features from previous frames.



First frame

Drift

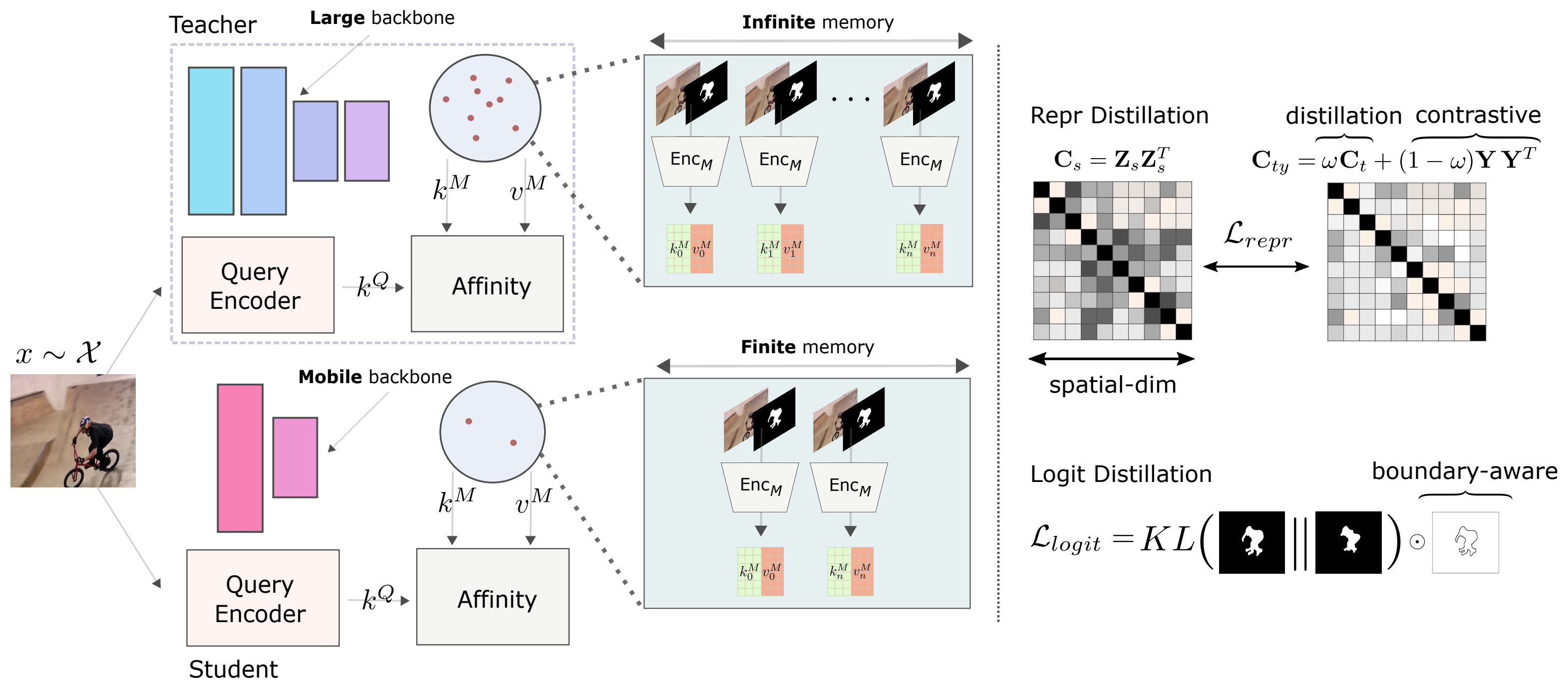
The STM memory model scales poorly for longer video sequences and introduces problems, such as drift, where the model can catastrophically degrade in performance over time.

Contributions

Our contributions can be summarized as follows

1. We introduce a new loss function that **unifies representation distillation and supervised contrastive learning** to address the gap between large and small memory models. Additionally, we proposes a **boundary-aware pixel sampling** strategy that further improves results and model convergence.
2. By using this unified loss, we demonstrate that a common network design can achieve performance comparable performance to state-of-the-art models, while being up to **5x faster and having 32x fewer parameters**.
3. The proposed loss function enables **real-time performance (30FPS+)** on **mobile devices** like the Samsung Galaxy S22 without the need for complex architectural or memory design changes, while still maintaining competitive performance with state-of-the-art models.
4. MobileVOS is also shown to be **robust to domain shift and shot changes**.

Finite memory space-time-memory networks



- To enable real-time performance, we use a much smaller MobileNet backbone.
- To encourage temporally consistent features, we distill knowledge a pre-trained infinite memory teacher.

Unifying contrastive learning and knowledge distillation

We propose a novel unification of both representation distillation and supervised contrastive representation learning. This is achieved through the following objective.

$$\mathcal{L}_{repr} = \frac{1}{|\mathbf{C}_s|}\left(\log_2 \|\mathbf{C}_s\|^2 - \log_2 \|\mathbf{C}_s \odot \mathbf{C}_t\|^2\right) \tag{1}$$

where $\mathbf{C}_s, \mathbf{C}_t \in \mathbb{R}^{HW \times HW}$ capture the relationship between all pairs of pixels in the student and teacher feature space respectively.

By introducing known relationships between the pixel-wise features, we can provide a natural scheme to interpolate between knowledge distillation and supervised constrastive learning.

$$\mathbf{C}_{ty} = \omega\mathbf{C}_t + (1-\omega)\mathbf{Y}\mathbf{Y}^T \tag{2}$$

In the case where $\omega = 0$, we arrive at a familiar supervised constrastive setting.

$$\mathcal{L}_{repr} \rightarrow \mathcal{L}_{SupCon} = -\frac{1}{|\mathbf{C}_s|}\log_2 \sum_i \frac{\sum_{j \in \mathcal{P}_i} sim(\mathbf{Z}_i, \mathbf{Z}_j)}{\sum_k sim(\mathbf{Z}_i, \mathbf{Z}_k)} \tag{3}$$

Boundary-aware sampling

Sampling the boundary pixels not only improves model convergence and addresses observed limitations of SVOS models, but also significantly reduces the memory constraints in constructing these matrices.

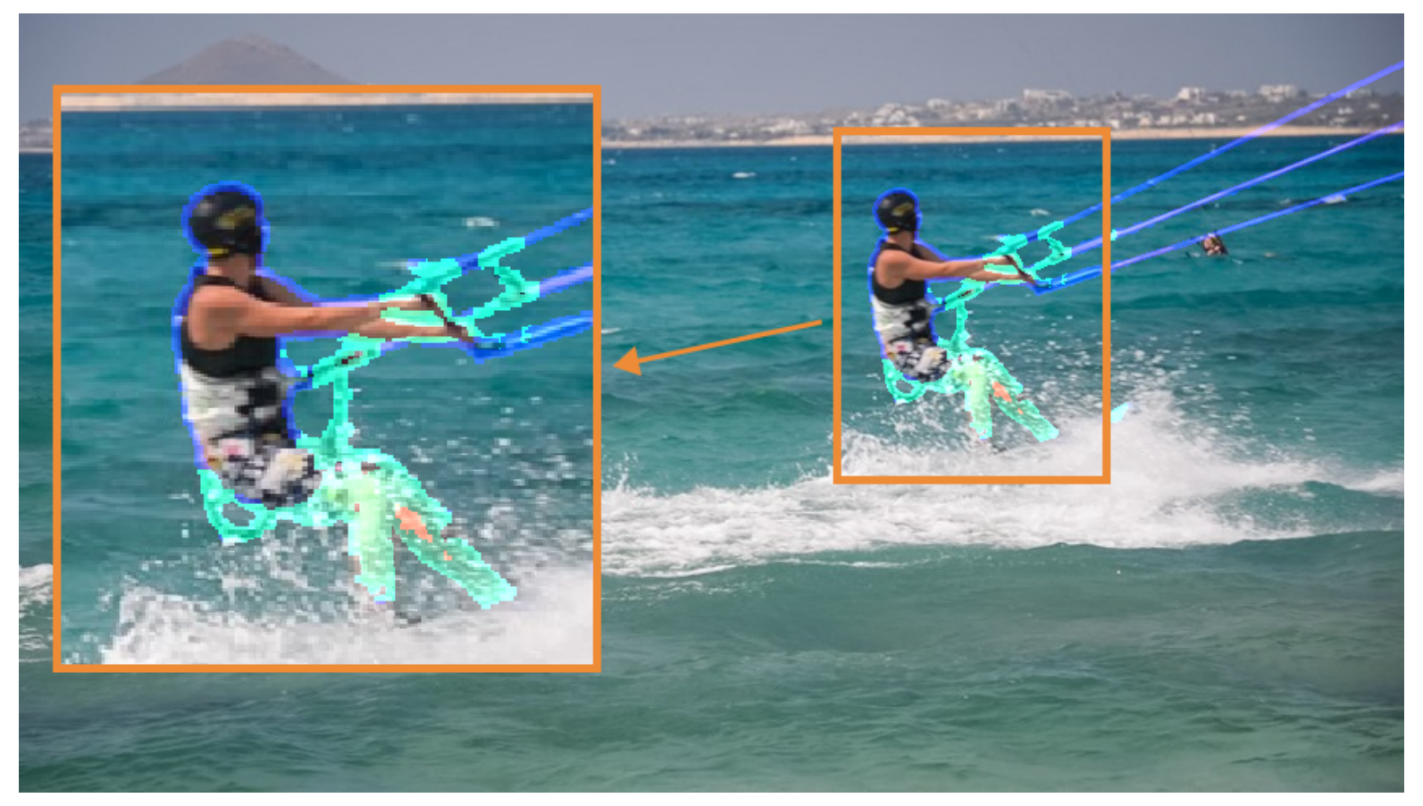


Figure 1. Prediction errors, shown in cyan, can typically occur on the boundaries of the segmented object, thus motivating the emphasis on distilling and contrasting boundary pixels.

Comparisons to state-of-the-art

Method	CC	$\mathcal{J}\&\mathcal{F}$	FPS
STM†	✗	89.3	6.3
MiVOS†*	✗	91.0	16.9
STCN†*	✗	91.7	26.9
BATMAN	✗	**92.5**	-
XMem†*	✗	92.0	**29.6**
SwiftNet†	✓	90.4	25.0
RDE-VOS†*	✓	**91.6**	35.0
MobileVOS			
ResNet18†*	✓	91.4	**100.1**
MobileNetV2†	✓	90.5	81.8
↳ wo/ ASPP†	✓	90.1	86.0

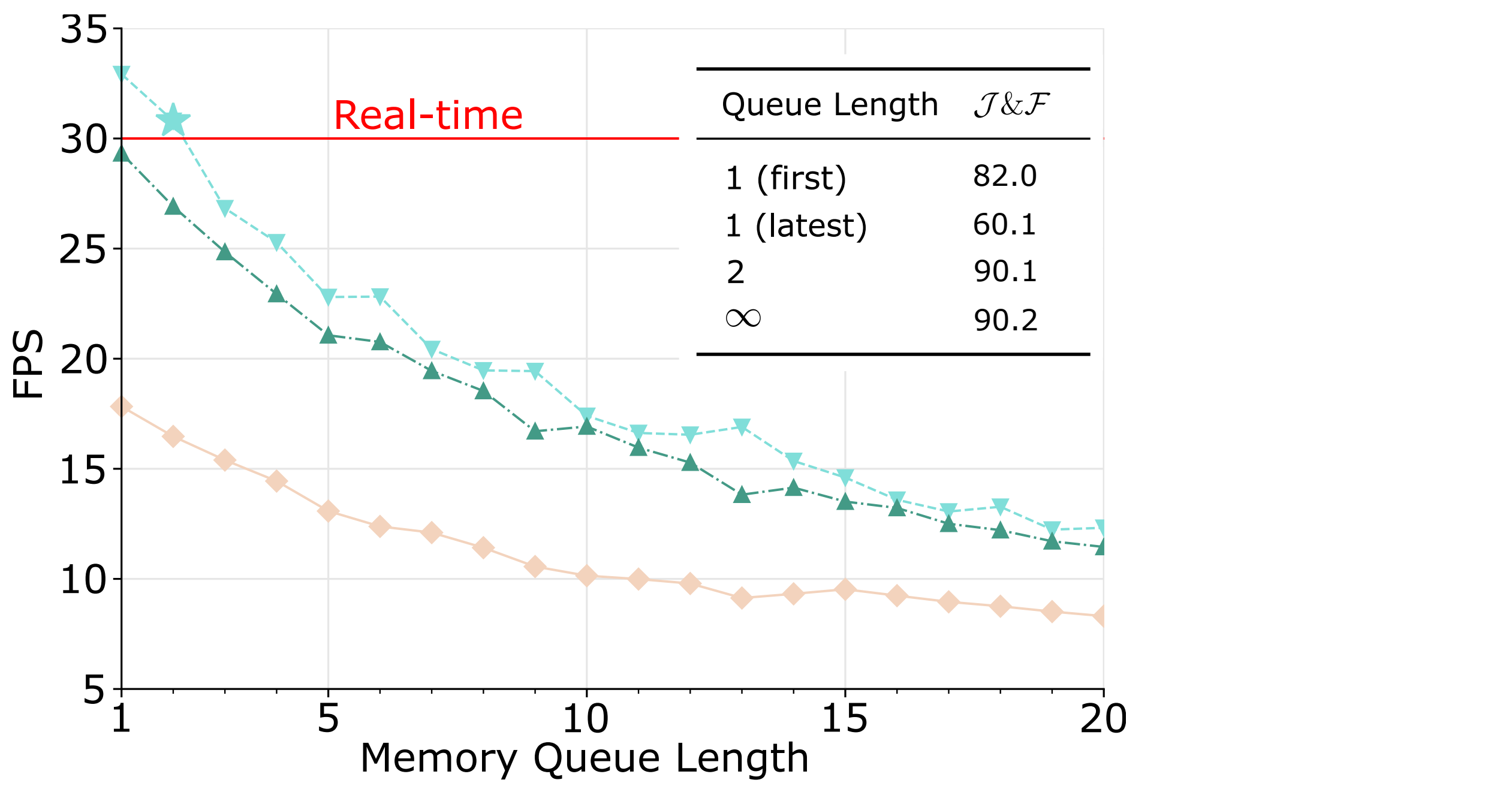
Method	CC	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	FPS
STM†	✗	81.8	79.2	84.3	10.2
STCN†*	✗	85.3	82.0	88.6	20.2
BATMAN	✗	86.2	83.2	89.4	-
XMem†*	✗	**87.7**	**84.0**	**91.4**	**22.6**
SwiftNet†	✓	81.1	78.3	83.9	<25.0
RDE-VOS†*	✓	**86.1**	82.1	**90.0**	27.0
MobileVOS					
ResNet18†*	✓	85.0	81.7	88.3	**90.6**
MobileNetV2†	✓	82.2	78.7	85.7	79.1
↳ wo/ ASPP†	✓	81.8	78.3	85.3	81.3

Table 1. **left:** DAVIS 2016. **right:** DAVIS 2017. CC denotes constant cost during the inference. † indicates YouTube-VOS is added during the training stage. $*$ denotes BL30K is added during the training stage. For both CC and non-CC methods, the best results are highlighted in **bold**, while the second best results are underlined. FPS was averaged over 3 runs.

Mobile Performance

Method	Params(M)	FPS NVIDIA A40		FPS NVIDIA 1080Ti	
		short	long (10×)	short	long (10×)
STM	38.9	8.9	4.3	6.8	✗
GSFM	67.0	18.4	4.2	7.6	✗
STCN	54.4	37.4	8.3	18.1	✗
RDE-VOS	64.0	32.0	34.2	14.4	14.1
XMem	62.2	38.6	39.9	12.6	12.7
MobileVOS					
ResNet18	8.1	**144.7**	**145.4**	**76.0**	**76.3**
MobileNetV2	2.5	99.9	99.1	61.6	60.6
↳ wo/ ASPP	**1.9**	105.1	103.4	66.8	67.4

Table 2. Our models are the first to attain real-time performance on both server-grade and consumer-grade GPUs for both long and short video sequences.



Queue Length	$\mathcal{J}\&\mathcal{F}$
1 (first)	82.0
1 (latest)	60.1
2	90.1
∞	90.2

Figure 2. Runtimes of our proposed models with different memory queue lengths were evaluated on a Samsung Galaxy S22 GPU. The line with ▼ markers is the MobileNetV2 wo/ ASPP, ▲ markers is the MobileNetV2, while ♦ markers are for the Resnet18. The table shows $\mathcal{J}\&\mathcal{F}$ on the DAVIS 2016 validation set for MobileNetV2 wo/ ASPP on memory queue lengths of 1, 2 and unbounded.

r.miles18@imperial.ac.uk