Towards Open-Vocabulary Video Instance Segmentation

Haochen Wang, Shuai Wang
University of Amsterdam
{h.wang3, s.wang3}@uva.nl

Cilin Yan
Beihang University
clyan@buaa.edu.cn

Xiaolong Jiang, Xu Tang, Yao Hu
Xiaohongshu Inc.
laige@xiaohongshu.com

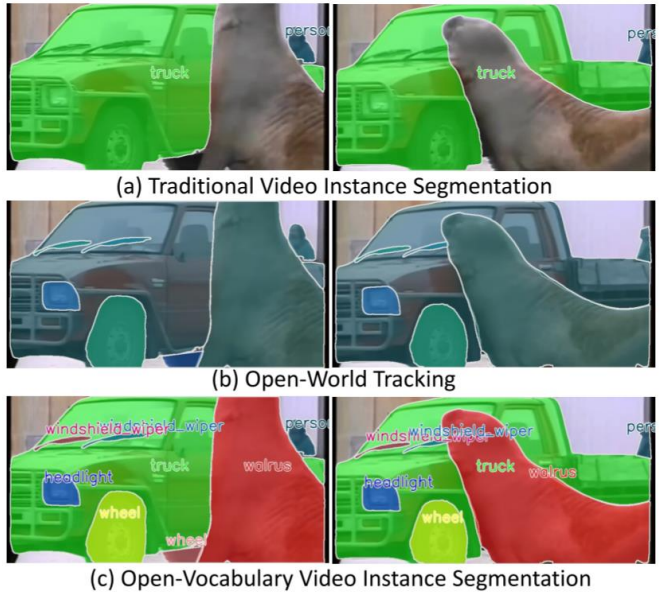
Weidi Xie
Shanghai Jiao Tong University
weidi@robots.ox.ac.uk

Efstratios Gavves
University of Amsterdam
egavves@uva.nl

Background

1. Open-Vocabulary Video Instance Segmentation

Def. Open-Vocabulary VIS aims to simultaneously segment, track, and classify objects for both training categories and novel categories.



Background

2. Related Work

a. VIS: segment, track and classify object instances from pre-defined training categories.

b. Open-Vocabulary Object Detection: detect objects in images beyond a close vocabulary set

c. Open-World Tracking: segment and track all the objects in videos

d. Open-Vocabulary Segmentation (image): segment objects in images beyond a close vocabulary set

Method

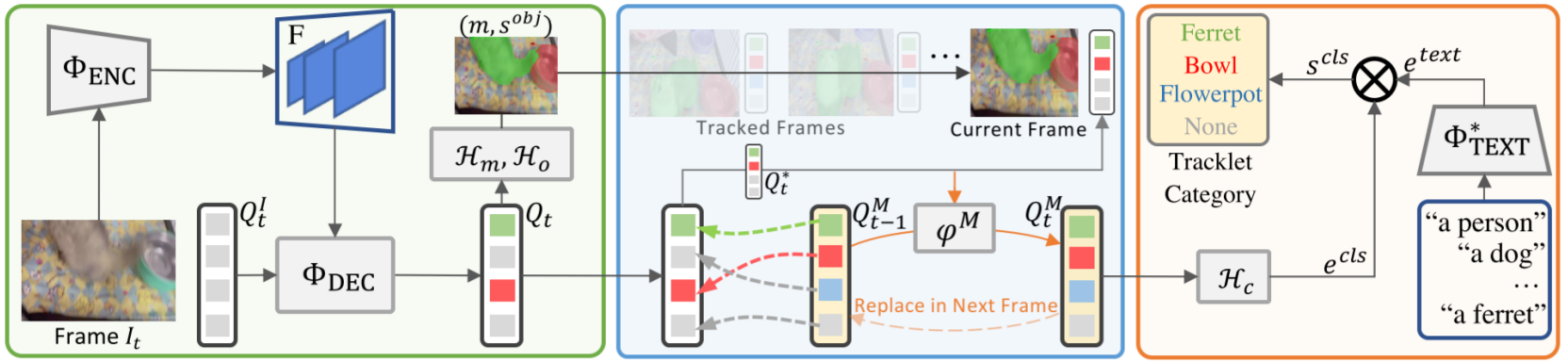


Figure 3. Overview of MindVLT. (a) Universal Object Proposal: Input with frame I_t, a transformer encoder Φ_{ENC} is adopted to extract multi-scale features $\mathcal{F}$. N class-independent queries $Q_t^I \in \mathbb{R}^{N\times d}$ are fed to a transformer decoder Φ_{DEC} to generate object-centric queries Q_t. Then Q_t is utilized to produce the segmentation mask $m \in \mathbb{R}^{N\times H\times W}$ and object score $s^{\text{obj}} \in \mathbb{R}^{N\times 1}$ with a mask head $\mathcal{H}_m$ and an object score head $\mathcal{H}_o$, respectively. (b) Memory-Induced Tracking: A set of Memory Queries $Q_{t-1}^M \in \mathbb{R}^{N\times d}$ is proposed to associate Q_t with Hungarian Algorithm. Q_{t-1}^M is updated by a function φ^M to dynamically aggregate the associated object-centric queries Q_t^* through time and obtain Q_t^M for the tracking of the next frame. (c) Open-Vocabulary Object Classification: A class head $\mathcal{H}_c$ is applied on Q_t^M to generate the class embedding $\mathbf{e}^{\text{cls}} \in \mathbb{R}^{N\times d}$. $\mathbf{e}^{\text{cls}}$ is then dot-producted with the text embedding $\mathbf{e}^{\text{text}}$ generated by a text encoder $\Phi_{\text{TEXT}} \in \mathbb{R}^{|\mathcal{C}|\times d}$ to obtain the classification score $s^{cls} \in \mathbb{R}^{N\times|\mathcal{C}|}$ for a open set of categories $\mathcal{C}$.

Method

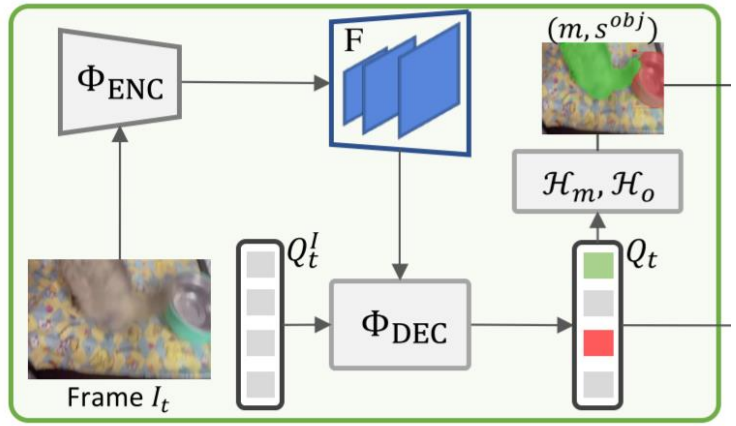
1. Universal Object Proposal

$$Q = \Phi_{\text{DEC}}(\mathcal{F}, Q^I) \in \mathbb{R}^{N \times d}.$$

$\mathcal{F}$: multi-scale feature
Q^I: N class-independent learnable object queries
Q: object-centric queries

$$m, s^{\text{obj}} = \mathcal{H}_m(Q) \circledast f^{-1}, \mathcal{H}_o(Q),$$

$\mathcal{H}_m$: mask generation head
$\mathcal{H}_o$: object score head
f^{-1} : last layer of $\mathcal{F}$
$\circledast$: dynamic conv
$\mathcal{H}_m(Q)$: dynamic convolutional kernel for each object query



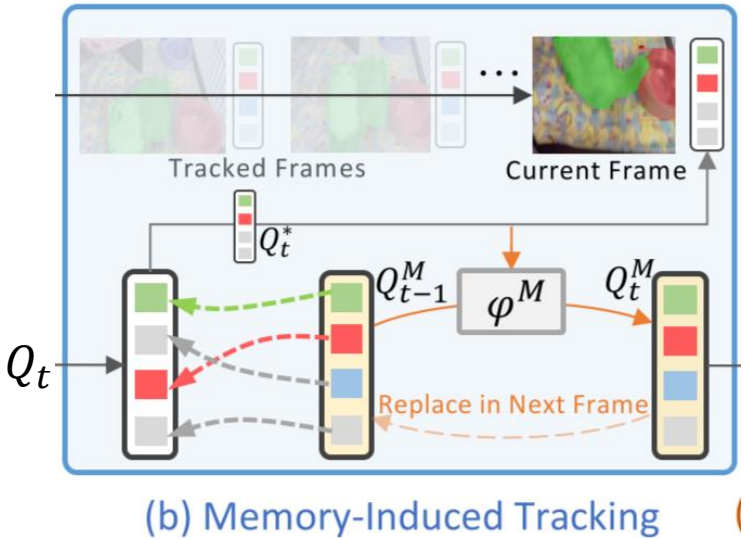
(a) Universal Object Proposal

Method

2. Memory-Induced Tracking

$$
\begin{aligned}
Q_t^M &= \varphi^M(Q_{t-1}^M, Q_t^*) \\
&= \alpha \cdot s^{\text{obj}} \cdot Q_t^* + (1 - \alpha \cdot s^{\text{obj}}) \cdot Q_{t-1}^M,
\end{aligned}
$$

Q_t: current frame object queries
Q_t^M: Memory queries
Q_t^*: associated object-centric queries after Hungarian algorithm(upon inner-product similarity matrix)
α: factor to control the update ratio
s^{obj}: object score for each queries



(b) Memory-Induced Tracking

Method

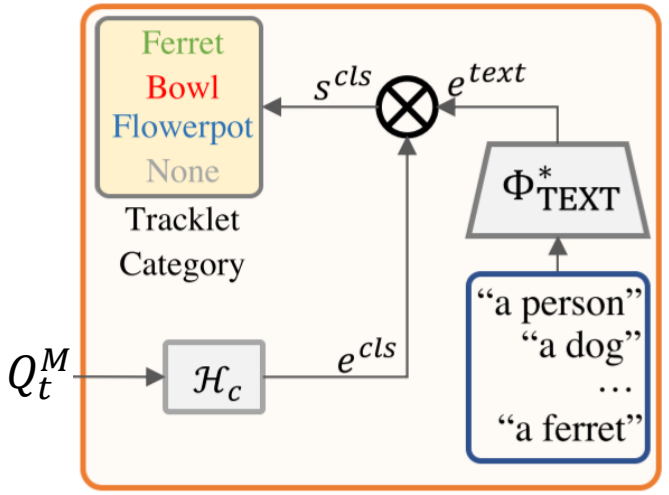
3. Open-Vocabulary Classification

$$\mathbf{e}^{\text{text}} = \Phi_{\text{TEXT}}(\text{“this is a photo of [dog]”}),$$

$$s_{i,j}^{\text{cls}} = \sigma(\cos(\mathbf{e}_i^{\text{cls}}, \mathbf{e}_j^{\text{text}})/\epsilon),$$

$\mathcal{H}_c$: classification score head
e^{cls}: class embeddings
e^{text}: text embeddings from CLIP Text Encoder

The memory queries Q_t^M aggregate the object features from all frames weighted by the object score, the object features with low confidence are constrained thus leading to robust video object classification.

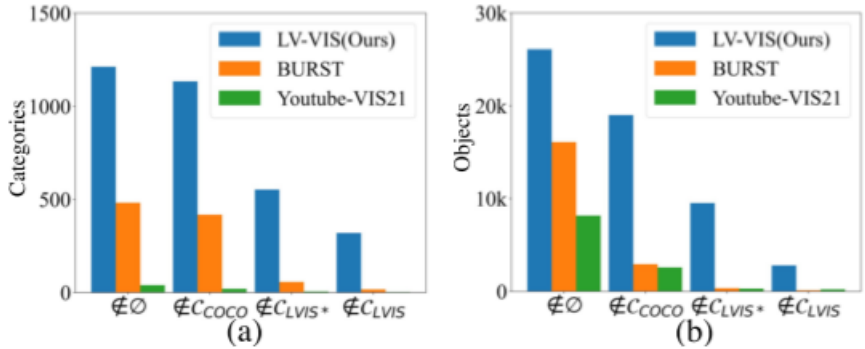


c) Open-Vocabulary Classification

Dataset

LV-VIS

Dataset	UVO [34]	YT19 [38]	YT21 [38]	OVIS [30]	BURST [1]	LV-VIS
Videos	11228	2883	3859	901	2914	4832
Instances	104898	4883	8171	5223	16089	26099
Masks	593k	131k	232k	296k	600k	656k
Mask/Frame	12.3	1.7	2.0	4.7	3.1	3.9
Object/Video	9.3	1.6	2.1	5.8	5.5	5.4
Categories	80*	40	40	25	482	1212



The LV-VIS dataset comprises 4,832 real-world videos with a total length of 7.8 hours and contains 656,130 pixel-level annotated segmentation masks from 1,212 categories. The categories in LV-VIS are split into 659 base categories (seen during training) inherited from frequent and common categories in LVIS and 553 novel categories disjoint with the base categories.

Experiments

Trained on LVIS

Method	Backbone	Val			Test			fps
		AP	AP_b	AP_n	AP	AP_b	AP_n	
DetPro [13]-SORT [5]	R50	9.9	14.9	4.1	5.7	8.8	2.8	3.1
Detic [44]-SORT [5]	R50	10.1	15.1	4.2	5.7	8.5	3.1	6.0
DetPro [13]-OWTB [26]	R50	11.8	17.1	5.7	7.1	9.9	4.0	3.1
Detic [44]-OWTB [26]	R50	11.5	16.8	5.3	7.1	10.1	3.8	5.9
MindVLT(Ours)	R50	**14.1**	**18.1**	**9.4**	**8.6**	**10.7**	**6.4**	**20.1**
Detic [44]-SORT [5]	SwinB	14.9	20.7	8.2	11.0	16.2	5.4	6.7
Detic [44]-OWTB [26]	SwinB	15.8	21.5	9.1	11.7	16.4	6.6	6.8
MindVLT(Ours)	SwinB	**21.4**	**26.3**	**15.8**	**14.9**	**18.8**	**10.6**	**16.8**

Table 2. The performance comparison on LV-VIS validation and test set. The AP, AP_b, and AP_n mean the average precision of overall categories, base categories, and novel categories.

Experiments

Image-level Open-Vocabulary Detection.

By removing the Memory-Induced Tracking module

Method	Backbone	Stage	LVIS		LV-VIS(O)		fps
			AP	AP_n	AP	AP_n	
ViLD [15]	R50	2	25.5	16.6	-	-	-
DetPro* [13]	R50	2	25.7	18.7	21.4	13.6	3.4
Detic [44]	R50	2	30.2	16.4	22.8	12.7	6.2
OV-DETR [42]	R50	1	26.6	17.4	21.5	13.0	0.10
MindVLT(Ours)	R50	1	25.4	17.5	20.9	13.2	21.6
Detic [44]	SwinB	2	38.4	21.9	28.5	18.1	7.6
MindVLT(Ours)	SwinB	1	31.4	21.8	27.1	18.4	18.2

Table 3. Image-level performance on LVIS dataset and LV-VIS validation dataset. The LV-VIS(O) means the oracle image evaluation, where we regard all the frames in LV-VIS as an image dataset and evaluate on the images separately. DetPro* is conducted by single-scale inference for fair comparison.

Experiments

Class-Dependent vs. Class-Independent Queries.

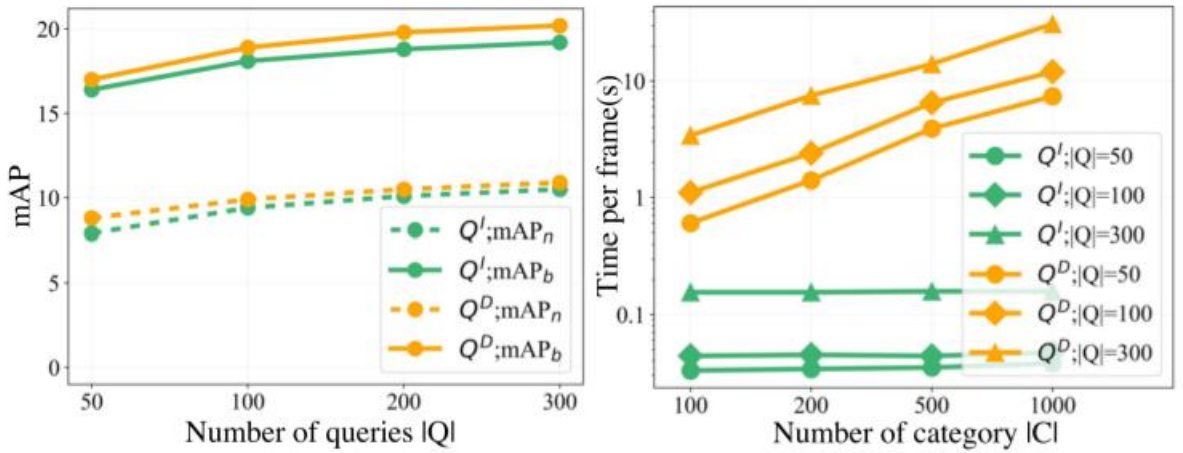


Figure 4. mAP and inference time per frame (s) comparison on LV-VIS validation dataset between class-dependent queries (Q^D) and class-independent queries (Q^I).

Experiments

Memory Queries.

$$
\begin{aligned}
Q_t^M &= \varphi^M(Q_{t-1}^M, Q_t^*) \\
&= \alpha \cdot s^{\text{obj}} \cdot Q_t^* + (1 - \alpha \cdot s^{\text{obj}}) \cdot Q_{t-1}^M,
\end{aligned}
$$

Methods	LV-VIS		Youtube-VIS2019	
	mAP_b	mAP_n	mAP_b	mAP_n
Average	17.7	8.7	25.5	9.8
Memory Queries	**18.1**	**9.4**	**25.8**	**10.4**

Table 5. Comparison between two ways to obtain the classification scores. The Memory Queries in the table indicate obtaining classification scores from the memory queries. The Average in the table means directly averaging the per-frame classification scores.

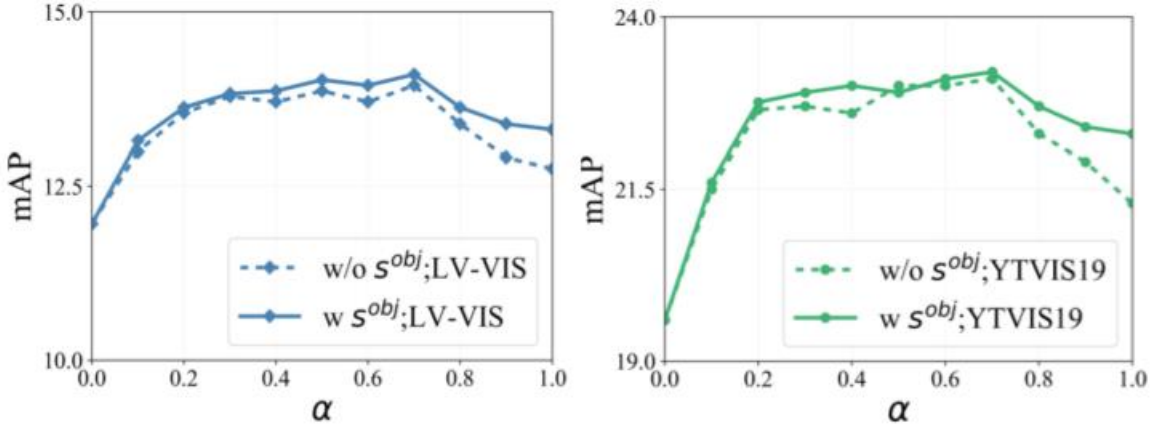


Figure 5. Overall mean Average Precision (mAP) under different memory update factor α on LV-VIS and Youtube-VIS2019 validation datasets. The w/o s^{obj} means removing the object score s^{obj} in the memory update module.

Experiments

Zero-shot Generalization on VIS Datasets.

Method	Open	Backbone	YTVIS2019			YTVIS2021			BURST			OVIS
			mAP	mAP_b	mAP_n	mAP	mAP_b	mAP_n	mAP	mAP_b	mAP_n	mAP
FEELVOS [38]	✗	R50	26.9	-	-	-	-	-	-	-	-	9.6
MaskTrack [38]	✗	R50	30.3	-	-	28.6	-	-	-	-	-	10.8
SipMask [7]	✗	R50	33.7	-	-	31.7	-	-	-	-	-	10.2
Mask2Former [9]	✗	R50	46.4	-	-	40.6	-	-	-	-	-	17.3
Detic [44]-SORT [5]	✓	R50	14.6	17.0	3.5	12.7	14.4	3.1	1.9	1.8	**2.5**	6.7
Detic [44]-OWTB [26]	✓	R50	17.9	20.7	4.5	16.7	18.6	5.8	2.7	2.8	1.8	9.0
MindVLT(Ours)	✓	R50	**23.1**	**25.8**	**10.4**	**20.9**	**22.7**	**10.9**	**3.7**	**3.9**	2.4	**11.4**
Detic [44]-SORT [5]	✓	SwinB	23.8	27.2	7.9	21.6	23.7	9.8	2.5	2.7	1.0	11.7
Detic [44]-OWTB [26]	✓	SwinB	30.0	34.3	9.7	27.1	29.9	11.4	3.9	4.1	2.4	13.6
MindVLT(Ours)	✓	SwinB	**37.6**	**41.1**	**21.3**	**33.9**	**36.7**	**18.2**	**4.9**	**5.3**	**3.0**	**17.5**

Table 4. Performance comparison on the validation sets of four published datasets: Youtube-VIS19, Youtube-VIS21, BURST, OVIS. The Open in the table indicates whether a method is capable of tackling objects from novel categories following the setting of Open-Vocabulary VIS. The methods above the double horizontal lines are trained on target training videos of each dataset; we only report the overall mean average precision mAP of those methods. The methods below the double horizontal lines are trained on image dataset LVIS and evaluated on the video instance segmentation datasets without fine-tuning; we report both the mAP, mAP_b, and mAP_n of those methods.