Video Instance Segmentation 2021: Category-Aware Sequence Reduction for Propose-Reduce Paradigm

Huaijia Lin[1*] Ruizheng Wu[1*] Shu Liu[2] Jiangbo Lu[2] Jiaya Jia[1,2]
[1]The Chinese University of Hong Kong [2]SmartMore
{linhj, rzwu, leojia}@cse.cuhk.edu.hk {sliu, jiangbo}@smartmore.com

Abstract

Video instance segmentation (VIS) aims to segment and associate all instances of predefined categories for each frame in a video. A recently proposed paradigm – Propose-Reduce, generates instance sequence proposals based on multiple key frames and reduces redundant sequences of the same instance. However, new redundancy appears after assigning categories to the preserved sequences. In this work, we introduce a technique – category-aware sequence reduction, to reduce redundancy within the same category. This significantly improves the performance under the Propose-Reduce paradigm. The final model achieves the fourth place in the 2021 YouTube-VIS challenge with a mAP score of 47.8%.

1. Introduction

Video instance segmentation (VIS) [24] is a task to segment all instances of the predefined classes in each frame. Segmented instances are linked throughout the entire video. It is important in the field of video understanding, which can be applied to video editing, autonomous driving, *etc*.

One recent proposed paradigm – Propose-Reduce (Fig. 1(a)), achieves the state-of-the-art results for VIS. It tackles VIS in a two-stage pipeline. In the first stage, multiple sequence proposals are generated from sparsely sampled key frames. The redundant sequences of the same instance are reduced in the second stage. Preserved sequences are assigned with corresponding predicted categories.

However, new redundancy appears after the category assignment. Sequences assigned to the same category conflict with each other under the category-wise evaluation. To tackle the conflict, this paper proposes a category-aware reduction (Fig. 1(b)) for sequences within the same category. The idea of two types of reductions has ever been explored in two-stage instance segmentation methods [11, 13]. A category-agnostic reduction for object proposals and a category-aware reduction after object classification.

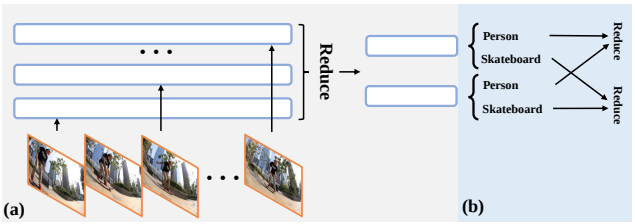

Figure 1: We equip (a) **Propose-Reduce** paradigm [15] with (b) **category-aware sequence reduction**.

Incorporating the category-aware sequence reduction technique, the accuracy of the best model in [15] is improved by 2% in terms of $\mathcal{AP}$. Equipped with a stronger classifier, our final model achieves 49.3% on the validation set and 47.8% on the testing set for the YouTube-VIS 2021 challenge.

2. Related Works

Video Instance Segmentation Methods for video instance segmentation can be grouped into three types of paradigms: 'Track-by-Detect', 'Clip-Match' and 'Propose-Reduce'. 'Track-by-Detect' [24, 6, 19] associates detected instances into tracklets in a frame-by-frame manner. 'Clip-Match' [1, 2] divides an entire video into multiple clips and matches sequences between adjacent clips. 'Propose-Reduce' [15] proposes multiple instance sequences at once and then reduces the redundant ones. This work introduces a category-aware sequence reduction technique to complete the 'Propose-Reduce' paradigm.

Image Instance Segmentation Image instance segmentation is a hot task with many solutions [11, 13, 17, 22, 4, 20] being proposed. One main stream with high performance is Mask R-CNN [11] and its variants [13, 5, 7]. It is built on a two-stage detector [21] that adds a mask head in parallel with the original detection head. In the first stage, RPN [21] obtains multiple proposals and filters redundant proposals with a category-agnostic NMS. In the second stage, the detection head predicts category scores, and a category-aware

*Equal Contribution.

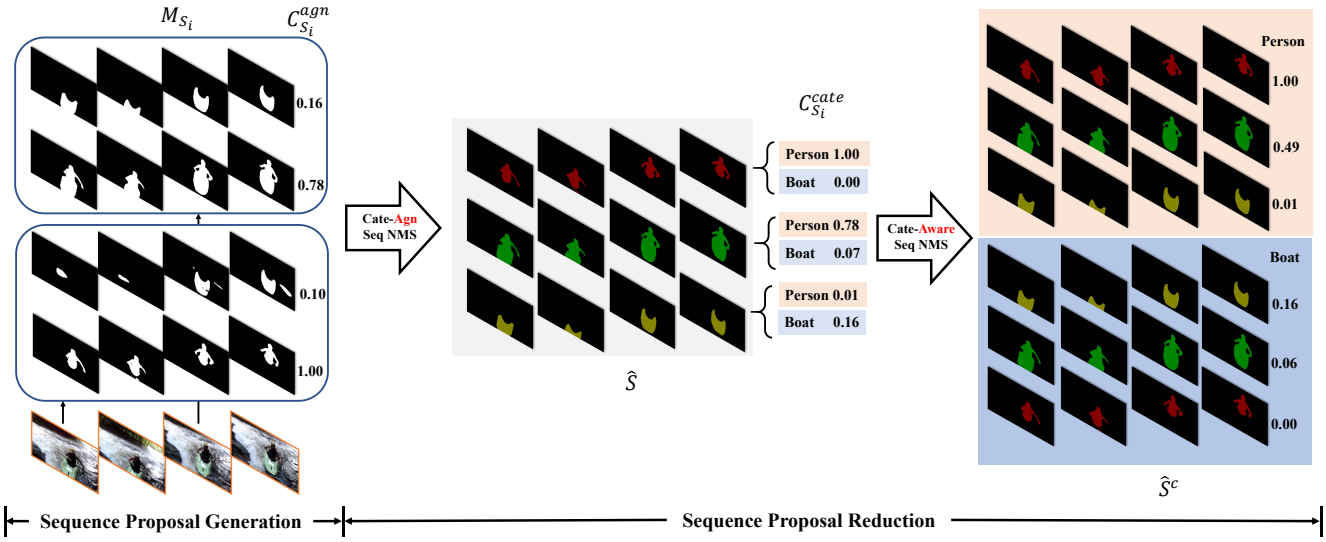

Figure 2: **Paradigm illustration.** $\mathcal{M}_{S_i}$, $\mathcal{C}^{agn}_{S_i}$ and $\mathcal{C}^{cate}_{S_i}$ denote the masks, agnostic score and category scores for a sequence. $\hat{\mathbb{S}}$ and $\hat{\mathbb{S}}^c$ denote sequence sets after reductions. 'Cate-Agn Seq NMS' and 'Cate-Aware Seq NMS' refer to category-agnostic (Sec. 3.1) and category-aware (Sec. 3.2) sequence NMS, respectively.

NMS is applied within each category across all predictions. The category-agnostic sequence NMS proposed in [15] can be analogous to the first NMS and the category-agnostic sequence NMS introduced in this paper to the second one.

3. Method

Given a category set $\mathbb{C}$, VIS targets finding all instance sequences that belong to the category set in a video. An instance sequence consists of a sequence mask, an assigned category, and a corresponding score. Sec. 3.1 reviews the 'Propose-Reduce' paradigm and Sec. 3.2 introduces the proposed category-aware sequence reduction.

3.1. Propose-Reduce Paradigm

As shown in Fig. 2, 'Propose-Reduce' consists of two stages. Multiple sequences are first proposed to ensure high recall, where redundant sequences are then reduced.

Sequence Proposals For a T-frame video, it evenly selects K key frames to detect O instances in each key frame. They are collected as the proposed instance set $\mathbb{S} = \{S_0, .., S_i, .., S_{O\times K-1}\}$. Each instance mask is propagated to the whole video as the corresponding sequence mask $\mathcal{M}_{S_i} \in [0,1]^{T\times H\times W}$. [15] proposes Sequence Mask R-CNN (Seq Mask R-CNN) to instantiate this procedure, that attains a Seq-Prop Head on the Mask R-CNN. Mask R-CNN obtains instance segmentation in key frames and sequence masks are propagated via the Seq-Prop head.

Sequence Scoring To reduce redundant sequences in $\mathbb{S}$, each sequence requires a score to measure its prediction confidence. In [15], such a sequence score for $\mathcal{S}_i$ is calculated from the per-frame detection scores $\mathcal{C}^{det}_{S_i} \in [0,1]^{T\times|\mathbb{C}|}$ (obtained via the detection head in Seq Mask R-CNN). Then a category-agnostic score

$$\mathcal{C}^{agn}_{S_i} = \max_{c\in|\mathbb{C}|} \mathcal{C}^{cate}_{S_i}(c)\ ,\ \mathcal{C}^{cate}_{S_i} = \frac{1}{T}\sum_t \mathcal{C}^{det}_{S_i}(t)\ , \qquad (1)$$

is used to measure the sequence confidence.

Sequence Reduction With the category-agnostic score $\mathcal{C}^{agn}_{S_i} \in [0,1]$, category-agnostic sequence NMS [15] is applied to remove redundant low-score sequences.

$$\hat{\mathbb{S}} \leftarrow \text{NMS}(\{\mathcal{C}^{agn}_{S_0}, \mathcal{M}_{S_0}\}..\{\mathcal{C}^{agn}_{S_i}, \mathcal{M}_{S_i}\}..)\ . \qquad (2)$$

Preserved sequences $\hat{\mathbb{S}} = \{\hat{S}_0.., \hat{S}_j, ..\}$ are assigned with categories and corresponding scores $\mathcal{C}^{cate}_{\hat{S}_j}$ as the output.

3.2. Category-Aware Sequence NMS

New redundancy appears after the category assignment. An example is shown in Fig. 2. The first two sequences both have a high 'person' score. During the evaluation, when the first sequence matches with the ground truth, the second one will become a high-ranking false positive that harms the accuracy. To solve this problem, we introduce category-aware sequence NMS that performs reduction within categories.

$$\hat{\mathbb{S}}^c \leftarrow \text{NMS}(\{\mathcal{C}^{cate}_{\hat{S}_0}(c), \mathcal{M}_{\hat{S}_0}\}..\{\mathcal{C}^{cate}_{\hat{S}_j}(c), \mathcal{M}_{\hat{S}_j}\}..)\ , c \in \mathbb{C}\ . \qquad (3)$$

Redundant sequences in $\hat{\mathbb{S}}^c$ are suppressed to a lower score (*e.g.*, $0.78 \rightarrow 0.49$ for the second 'person').

In practice, Eq. 2 is implemented by traditional NMS [9] to reduce the number of sequences, while Eq. 3 is implemented by soft NMS [3] to achieve better accuracy.

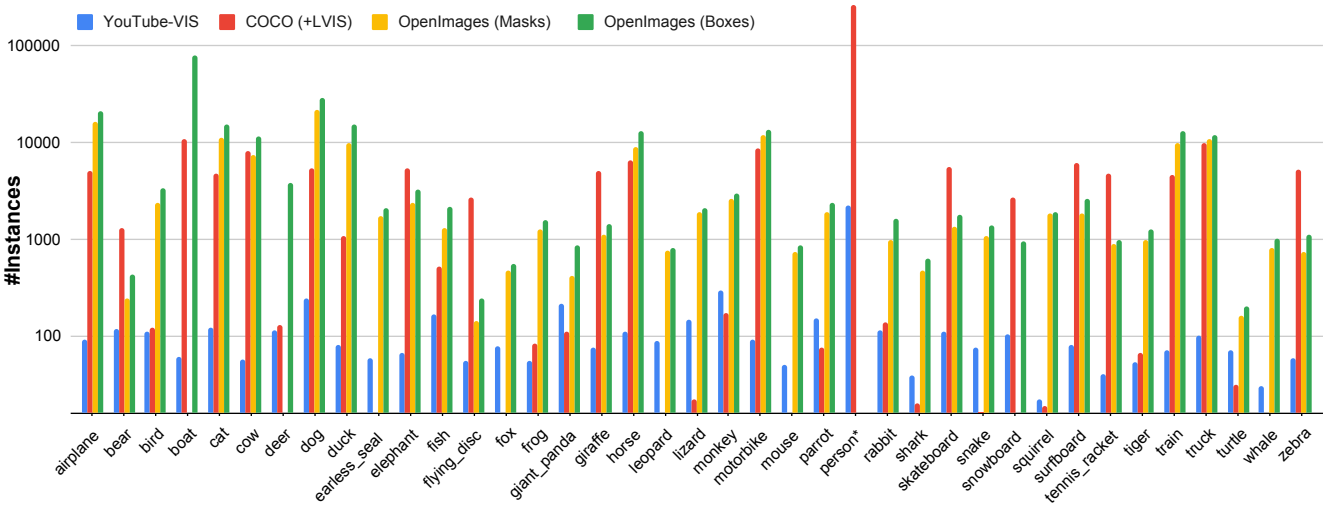

Figure 3: Data statistics for training datasets. Empty columns indicate categories that do not exist in the datasets. *: Category 'person' in OpenImages is abandoned since it contains too many 'person' annotations.

4. Experiments

Following [15], we incorporate COCO image dataset [16] to ease the over-fitting issue on YouTube-VIS video dataset. However, some categories are not collected in COCO (see Fig. 3). To tackle the category missing problem, we collect data from OpenImages [14] to cover all categories. Due to limited computation resources (6 GPUs), OpenImages is only used to train a stronger image-level classifier to save training time.

4.1. Datasets

YouTube-VIS YTV-VIS 2021 (abbreviated as YTV) contains 2,985 training videos, 421 validation videos and 453 test videos. It covers 40 categories that are slightly different from the 2019 version [24]. Category 'bird' is a union of 'eagle' and 'owl' in 2019. Category 'monkey' is a union of 'monkey' and 'ape' in 2019.

COCO To ease the data insufficiency in YTV, we collect COCO images to construct 3-frame pseudo videos with $\pm 30^\circ$ rotation [15]. COCO has 35 categories overlapping with YTV. Out of the 35 categories, 20 are from the COCO annotations while the remaining 15 from LVIS [10] annotations. Note that the 'bird' and 'monkey' in YTV are mapped to 'eagle+owl' and 'monkey+gorilla' in COCO.

OpenImages Two types of annotations in OpenImages are collected. One is mask annotations that cover 37 categories in YTV, while the other is box annotations covering all categories. The 'bird' and 'fish' in YTV are mapped to 'eagle+owl' and 'golden fish' in OpenImages.

Mask annotations are used to train an instance segmentation model [11], while box annotations are used to train a detection model [21]. In practice, only the detection head is used in the sequence scoring step. Note that OpenImages is not densely annotated, where the background may contain target objects (see Fig. 4). Inspired by [18], we multiply the image with annotations to exclude the background regions.

Figure 4: Examples in OpenImages where not all cars are annotated. To reduce ambiguity, non-annotated regions are filled with ImageNet [8] mean values, with respect to the mask and box annotations.

4.2. Implementation Details

The 'Propose-Reduce' paradigm is instantiated with Seq Mask R-CNN [15]. OpenImages mask annotations (OImgMask) are used to train a Mask R-CNN [11] and box annotations (OImgBox) are used to train a Faster R-CNN [21].

Training Following the two-stage training in [15], Seq Mask R-CNN is first trained on the mixed 'COCO+YTV' videos for 4 epochs, and then fine-tuned on YTV for 5 epochs. Similarly, Mask R-CNN and Faster R-CNN are trained on the mixed 'COCO+YTV+ OImgMask/OImgBox' images and then fine-tuned on YTV.

Input image size is resized to a fixed 640×320 input size (keep aspect ratio). All models are trained on 6 NVIDIA Titan X GPUs. It takes about 3 days to train with ResNet-50 [12] backbone and 5 days with ResNeXt-101 [23].

Inference In the sequence proposals stage, the number of key frames (K) is set as 5. In each key frame, the top 10 (O) detected instances with scores higher than 0.2 are used for generating sequence proposals.

	trainset	$\mathcal{AP}$	$\mathcal{AR}$@10
R50	YTV	39.4	51.9
	+ COCO	**41.7**	**55.0**
X101	YTV	42.8	52.9
	+ COCO	**47.9**	**58.0**

(a) **Backbone Analysis**: 'YTV' denotes training on YTV only, while '+COCO' refers to two-stage training with 'COCO+YTV' (Sec. 4.2).

Seq Scoring	$\mathcal{AP}$	$\mathcal{AR}$@10
-	47.9	58.0
+OMask	49.0	**58.3**
+OBox	**49.3**	58.1

(b) **Stronger Classifier**: '+OMask' and '+OBox' denote models that trained with additional OpenImages Mask/Box data.

Category-agnostic	aware	$\mathcal{AP}$	$\mathcal{AR}$@10
0.5	-	44.9	52.1
0.5	0.5	47.0	55.4
0.7	0.5	47.2	56.2
0.9	0.5	**47.9**	58.0
-	0.5	47.7	**58.8**

(c) **IoU Threshold**: '-' denotes that the corresponding NMS is not performed.

	NMS^c	$\mathcal{AP}$	$\mathcal{AR}$@10
R50		38.7	46.4
	✓	**41.7**	**55.0**
X101		44.9	52.1
	✓	**47.9**	**58.0**
X101*		47.5	53.6
	✓	**49.3**	**58.1**

(d) **Cate-aware NMS**: 'NMS^c' denotes category-aware sequence NMS. 'X101*' means using a stronger classifier trained on OImg Box data.

Table 1: **Ablations** in the 2021 YouTube-VIS (validation set). We only show $\mathcal{AP}$ and $\mathcal{AR}$@10 (%) for simplicity.

In the sequence reduction stage, sequence IoU [15] is used to measure the overlapping regions between sequence masks. Category-agnostic sequence NMS is implemented with traditional NMS [9] of 0.9 IoU threshold. Category-aware sequence NMS is implemented with soft-NMS [3] of 0.5 IoU threshold. Categories with a score threshold larger than $1e$-3 are selected for final results.

4.3. Ablation Studies

Backbone Tab. 1a compares the behavior of different backbones (*i.e.*, ResNet-50 vs. ResNeXt-101). In terms of $\mathcal{AP}$, the large backbone is prone to suffering from over-fitting in YTV and benefits more from incorporating more training data. Intriguingly, the $\mathcal{AR}$@10 of both backbones attains significant improvement with more data.

Stronger Classifier Tab. 1b reports the improvement of using a stronger classifier in the sequence scoring step (Sec. 3.1). Using a model trained with OpenImages improves by roughly 1% compared to using the original classification head in Seq Mask R-CNN. Classifier trained on OpenImage box annotations is slightly better than it is on mask annotations.

IoU Threshold Tab. 1c demonstrates the influence of the IoU threshold for two types of sequence NMS. The category-aware NMS largely improves the accuracy (44.9 vs. 47.0) by reducing redundancy within categories. Note that the accuracy changes little with and without the category-agnostic NMS (47.9 vs. 47.7). But the category-agnostic NMS can reduce largely redundant sequences to reduce computation requirements.

	$\mathcal{AP}$	$\mathcal{AP}$@50	$\mathcal{AP}$@75	$\mathcal{AR}$@1	$\mathcal{AR}$@10
tuantng	**54.1**	74.2	**61.6**	43.3	58.9
eastonssy	52.3	**76.7**	57.7	**43.9**	57.0
vidit98	49.1	68.1	54.5	41.0	55.0
Ours	47.8	69.3	52.7	42.2	**59.1**
hongsong.wang	47.6	68.4	52.9	41.4	54.6
gb7	47.3	66.5	51.1	40.5	51.6
zfonemore	46.1	64.4	51.0	38.3	50.6
DeepBlueAI	46.0	64.6	52.0	38.7	54.2

Table 2: Results in the 2021 YouTube-VIS Challenge (test set), compared to the top 7 other teams.

Category-Aware Sequence NMS Tab. 1d studies the effect of category-aware sequence NMS. Equipped with the category-aware NMS, the accuracy is stably improved over different baseline models. This technique can serve as an effective module for the 'Propose-Reduce' paradigm.

4.4. Testing Challenge

Tab. 2 illustrates the results of the challenge. The submitted model is a Seq Mask R-CNN of ResNeXt-101 backbone, equipped with a strong classifier trained on OpenImage Box data. Our approach achieves the highest $\mathcal{AR}$@10, which indicates its high recall characteristics.

5. Conclusion

In this paper, we introduce a category-aware sequence NMS to enhance the 'Propose-Reduce' paradigm for video instance segmentation. We also study the improvement of using extra OpenImages data to train a stronger classifier.

References

[1] Ali Athar, Sabarinath Mahadevan, Aljoša Ošep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *ECCV*, 2020. 1

[2] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *CVPR*, 2020. 1

[3] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *ICCV*, 2017. 2, 4

[4] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *ICCV*, 2019. 1

[5] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: high quality object detection and instance segmentation. *PAMI*, 2019. 1

[6] Jiale Cao, Rao Muhammad Anwer, Hisham Cholakkal, Fahad Shahbaz Khan, Yanwei Pang, and Ling Shao. Sipmask: Spatial information preservation for fast image and video instance segmentation. In *ECCV*, 2020. 1

[7] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi,

Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *CVPR*, 2019. 1

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 3

[9] Ross Girshick. Fast r-cnn. In *ICCV*, 2015. 2, 4

[10] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 3

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[13] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *CVPR*, 2019. 1

[14] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *IJCV*, 2020. 3

[15] Huaijia Lin, Ruizheng Wu, Shu Liu, Jiangbo Lu, and Jiaya Jia. Video instance segmentation with a propose-reduce paradigm. *arXiv:2103.13746*, 2021. 1, 2, 3, 4

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 3

[17] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *CVPR*, 2018. 1

[18] Jonathon Luiten, Philip Torr, and Bastian Leibe. Video instance segmentation 2019: A winning approach for combined detection, segmentation, classification and tracking. In *ICCVW*, 2019. 3

[19] Jonathon Luiten, Idil Esen Zulfikar, and Bastian Leibe. Unovost: Unsupervised offline video object segmentation and tracking. In *WACV*, 2020. 1

[20] Lu Qi, Xiangyu Zhang, Yingcong Chen, Yukang Chen, Jian Sun, and Jiaya Jia. Pointins: Point-based instance segmentation. *arXiv preprint arXiv:2003.06148*, 2020. 1

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 3

[22] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *ECCV*, 2020. 1

[23] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 3

[24] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, 2019. 1, 3