YouMVOS: An Actor-centric Multi-shot Video Object Segmentation Dataset

Donglai Wei[1†] Siddhant Kharbanda[2†*] Sarthak Arora[2*] Roshan Roy[2*] Nishant Jain[2*] Akash Palrecha[2*]
Tanav Shah[2*] Shray Mathur[2*] Ritik Mathur[2*] Abhijay Kemkar[2*] Anirudh Chakravarthy[2*] Zudi Lin[2]
Won-Dong Jang[2] Yansong Tang[3,4] Song Bai[5] James Tompkin[6] Philip H.S. Torr[4] Hanspeter Pfister[2]

[1]Boston College [2]Harvard University [3]Tsinghua-Berkeley Shenzhen Institute, Tsinghua University
[4]University of Oxford [5]ByteDance Inc. [6]Brown University

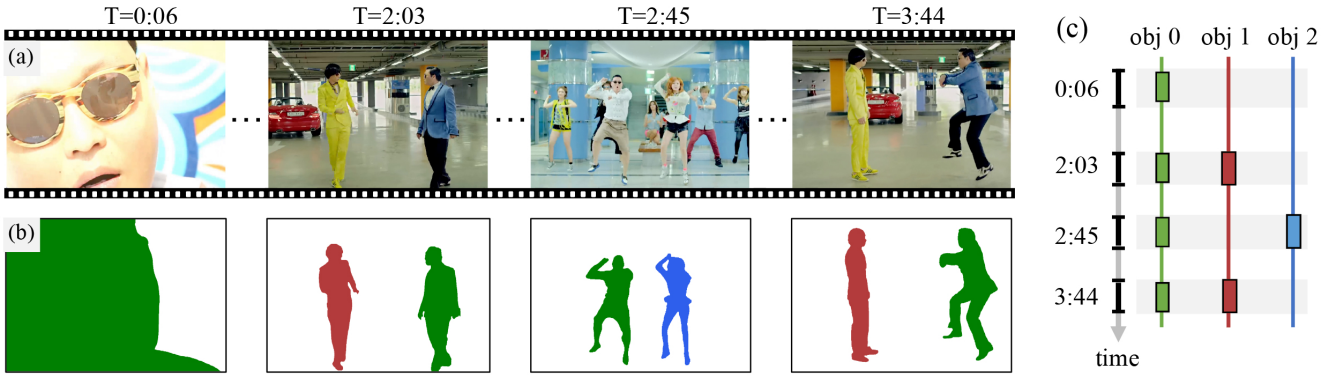

Figure 1: **Multi-shot video object segmentation (MVOS).** In multi-shot videos, MVOS aims to track and segment selected recurring objects despite changes in appearance (*e.g.*, person in green masks) and disconnected shots (*e.g.*, person in red masks). We show sample (a) frames, (b) segmentation masks, and (c) timelines for the *Gangnam Style* video in our dataset.

Abstract

Many video understanding tasks require analyzing multi-shot videos, but existing datasets for video object segmentation (VOS) only consider single-shot videos. To address this challenge, we collected a new dataset—YouMVOS—of 200 popular YouTube videos spanning ten genres, where each video is on average five minutes long and with 75 shots. We selected recurring actors and annotated 431K segmentation masks at a frame rate of six, exceeding previous datasets in average video duration, object variation, and narrative structure complexity. We incorporated good practices of model architecture design, memory management, and multi-shot tracking into an existing video segmentation method to build competitive baseline methods. Through error analysis, we found that these baselines still fail to cope with cross-shot appearance variation on our YouMVOS dataset. Thus, our dataset poses new challenges in multi-shot segmentation towards better video analysis. Data, code, and pre-trained models are available at `https://donglaiw.github.io/proj/youMVOS`

1. Introduction

The broad computer vision goal of video understanding must include the analysis of multi-shot videos with complex narratives [25, 32], including depictions of people and objects that vary in their visual appearance and space-time relationships across shot transitions. Video object segmentation (VOS) [37, 56] plays an essential role in video understanding. In multi-shot videos, this task requires accurately tracking and masking the same object across cuts despite appearance changes (Fig. 1). If multi-shot VOS is achieved, then it can ease video applications like editing [6] for privacy blur, relighting, or semantic color grading, and help in the analysis of poses and actions of specific objects. In the long term, multi-shot VOS data and methods can build towards a more sophisticated high-level video understanding.

Methods for VOS and the related video instance segmentation (VIS) problem are developed on datasets with single-shot videos that span only several seconds—a limitation highlighted in a recent survey [52]. Consequently, VOS methods often lose track of object instances in multi-shot videos, especially in videos with fast cuts and many object appearance changes, such as in music videos. As there are no existing multi-shot VOS datasets, it is hard to characterize these errors and reliably improve performance.

[†]Equally contributed.
[*]Research completed during internships at Harvard University.

To foster research in multi-shot VOS, we collect a new dataset—**YouMVOS**—consisting of 200 multi-shot YouTube videos at full length that are on average five minutes long and with 75 shots (Tab. 1). The average duration of videos in YouMVOS is at least 7.7× longer than existing VOS datasets. To choose representative online videos, we first pick ten popular video genres, including sports, cooking, and music videos, and then pick 20 popular videos in each genre with diverse content and temporal structures. Our multi-shot VOS dataset focuses on *actors*—human, animal, or virtual characters—that vary their positions, poses, and appearances across edited shots. This is similar in intent to object-specific VOS datasets, e.g., cars and pedestrians for autonomous driving [50]. To annotate our dataset efficiently, we build a semi-automatic system with manual proofreading using modern keyframe selection, mask initialization, and mask propagation. This produced 431K annotated instance masks—46% more masks than the latest VIS dataset [38].

To understand the new challenges in YouMVOS, we first benchmark existing unsupervised VOS methods [58] and video instance segmentation (VIS) methods [57, 8, 10], which were developed using short-term single-shot video datasets. To improve single-shot models, we examine model architectures and memory management practices to handle the change of actor position and appearance across shots. Then, we add multi-shot tracking to further improve baseline suitability to YouMVOS. The adapted model defines the baseline performance on YouMVOS. Finally, we perform error analysis using oracle data to identify where improvements can be made in the future, finding that cross-shot appearance variation is still a challenge.

Our contributions are 1) the YouMVOS dataset with 200 multi-shot YouTube videos and 431K annotated instance masks for the main actors, 2) an improved baseline segmentation model on YouMVOS to better handle long-term multi-shot videos, and 3) an error analysis of the improved baseline methods. Together, this provides a new challenge for the computer vision community and another step towards a more general understanding of complex videos. We also publicly release our data, code, and models.

2. Related Work

Object Segmentation in Videos. We refer readers to Wang *et al*. [52] for the survey on video object segmentation methods. There are two main settings: generic video object segmentation (VOS), where object classes are unknown, and video instance segmentation (VIS) with known classes. However, popular datasets only have single-shot videos.

For VOS, the DAVIS Challenge [37] is the de facto benchmark with semi-supervised, interactive, and unsupervised learning settings to segment single-shot video clips. Our task uses the unsupervised setting, where methods automatically discover primary objects that are frequently present across frames through co-occurrence, object saliency, or object detection [59, 33, 51, 17, 43, 34]. Instead of primary objects, our YouMVOS dataset focuses on segmenting actors, *i.e*., leading and supporting actors, from popular full-length YouTube videos that are mostly edited and multi-shot.

Datasets	Avg. dur. (sec)	Avg. #shots	Avg. #YT views†	Total #masks
$DAVIS_{17}$ [37]	2.9	1	N/A	14K
YTVOS [56]	4.5	1	0.1M	197K
YouMVOS (Ours)	**333.1**	**75**	**433.8M**	**431K**
MOTS [50]	43.4	1	N/A	65K
BDD [57]	40.0	1	N/A	129K
YTVIS [57]	4.6	1	0.1M	131K
OVIS [38]	12.7	1	N/A	296K
A2D [55]	5.0	1	0.5M	16K
J-HMDB [27]	1.0	1	N/A	31K

Table 1: **Statistics of VOS, VIS and VAAS datasets.** Our YouMVOS dataset contains full-length YouTube videos with more shots and masks. († by the time of submission)

VIS [57], or multi-object tracking and segmentation (MOTS) [50], aims to segment and track *all* instances of an object class in a single shot. For example, the YouTube-VIS dataset [57] has 40 categories of common objects, while the MOTS dataset [49] labels pedestrians and cars. Built upon existing image instance segmentation pipelines [21], early VIS and MOTS methods add a new tracking head [57, 50]. Recent advances include using better object detection modules [2, 53] and extracting richer image features [30]. However, recent state-of-the-art methods [5, 31] employ heavily engineered models with sophisticated inference schemes, making them inefficient for long-term videos, *e.g*. videos in our YouMVOS dataset. So instead, we start from an efficient baseline method and add improvements to achieve comparable performance without a heavily engineered scheme.

In addition, the actor-centric segmentation task, *e.g*., video actor-action segmentation (VAAS), has recently received attention [55, 18, 26, 12]. In addition to segmenting actors, this task requires classifying the corresponding action classes. However, videos in current datasets [55] are single-shot with only sparse and coarse mask annotation.

Video Shot Detection. Early shot detection methods cluster frames by color similarity [42], response curves from low-level visual features [40], and other modalities [45, 29]. In addition, spectral clustering [11] and dynamic programming [19, 48] algorithms have been applied. Popular benchmark datasets include IBM OVSD [41] and BBC Planet Earth [3]. The recent MovieScenes dataset [39] further groups shots into semantically consistent scenes for detection. Our baseline method adopts the shot detection results from an online k-means method.

Figure 2: **YouMVOS dataset.** We select ten major video genres with 20 popular videos in each. For each video, the dataset has high-quality segmentation masks of recurring actors for the whole video at 6 FPS.

Person Re-identification (Re-ID). To link the same actor across video shots, robust face [1] and body [23, 25] visual features are commonly used. In addition, audio features [35], text features [15, 20, 16, 46], and relational features [24] have been explored for the task. Inspired by Xia et al. [54], our improved baseline method uses multi-modal features to link actors across shots. To simplify the design, we use pre-trained Re-ID models as feature extractors.

3. YouMVOS Dataset

3.1. Dataset Construction

Video Selection. We compiled a list of YouTube video genres from online blogs and selected ten popular genres with high complexity: *music video, kid, movie trailer, cooking, pet, sports, show, how-to, education, and product* (Fig. 2). We excluded video genres with few recurring actors (*e.g.*, 'best-of' video compilations) or with static camera pose (*e.g.*, talking heads in gaming videos). For each genre, we selected 20 popular videos—200 total—while balancing gender, race, and sub-genres. The full-length videos were downloaded at 1280×720 resolution.

Recurring Actor Selection. Current VOS and VIS datasets label selected or all object instances within their single-shot videos. For our dataset, we annotated actors who appear in at least five shots in the video. In addition to human actors, we included animals and virtual characters to increase the diversity and difficulty of the dataset. In the end, we annotated on average 2.5 actors per video.

Video-Level Statistics. In Fig. 3a, we plot the mean number of YouTube views to show the popularity gap among video genres (blue bars) and video shots as a measure of complexity (red bars). As expected, *music video* has both the most number of views and the highest shot change frequency, while videos made by amateurs (*e.g.*, *pet*) have less sophisticated video structures. We categorize annotated actors into adult, child, animal, and virtual characters and plot a histogram of their occurrence (Fig. 3b). Virtual characters

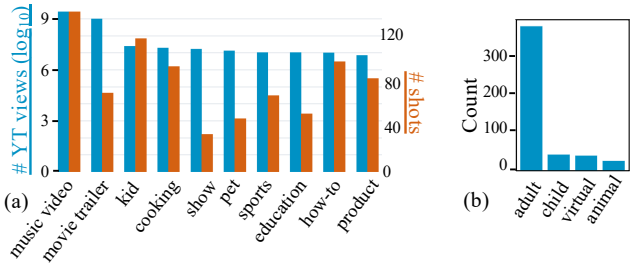

Figure 3: **Video-level statistics.** We plot (a) the average number of YouTube views and shots for videos in each genre, and (b) histogram of four different types of actors.

in YouMVOS pose new challenges in appearance that are absent in current VOS and VIS datasets.

3.2. Dataset Annotation

We annotated objects in representative keyframes and then propagated annotations to frames within corresponding video shots. To refine the actor masks, we built a semi-automatic annotation pipeline for annotators to correct errors from automatic results. Similar to Xu et al. [56], we annotated frames at 6 FPS.

Step 1: Shot Detection and Selection. The goal is to find frames at 1 FPS with the selected actors for annotation. We first divided frames into shots by frame clustering nearest neighbors. For clustering, we extracted features from the average pooling layer in a ResNet-18 network trained on ImageNet and computed the cosine distance among features. We built a Web visualization tool to correct the shot detection results and select shots containing actors of interest. Then, we pick frames that are closest to cluster centers as keyframes (Fig. 4a). These represent 0.01% of all frames, which reduces downstream mask initialization work.

Step 2: Mask Initialization. To create initial annotations, we run a pre-trained PointRend network [28] to generate segmentation masks on selected keyframes (Fig. 4b). Then, using the VAST volumetric segmentation annotation software [4] on our video data, human annotators selected masks

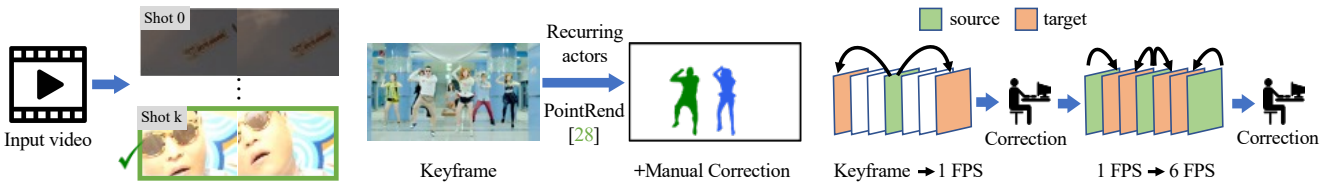

Figure 4: **Annotation pipeline.** (a) We divide frames into shots at 1-FPS, and then select shots with recurring actors. (b) We segment one keyframe per selected shot with a pre-trained PointRend network [28] and manual correction. (c) We iteratively propagate masks from keyframes at 1-FPS to 6-FPS frames with a pre-trained STM network [36] and manual correction.

for recurring actors from the PointRend result and manually corrected the masks. Due to the challenging scene composition and actor appearance, we found it not uncommon for automatic prediction to fail totally; in these cases, our human annotators manually segmented the actors.

Step 3: Mask Propagation. We propagate masks using a Space-Time Memory network (STM) [36]. However, its results can quickly degenerate due to complex actor appearance. Thus, we take a coarse-to-fine approach to mask propagation: from keyframes to labeling a frame every second, and then to labeling a frame around every 0.17 seconds to achieve the final 6 FPS annotations (Fig. 4c). After each propagation step, annotators correct the segmentation in corresponding frames. As mask density increases during annotation, the STM results improve significantly. For post-processing, we remove mask regions that are tiny.

Annotation Quality. To ensure the high quality of labeled masks, our annotators examined and corrected the segmentation results on all frames for each video. On average, each frame was inspected by three different annotators. Our annotation team had ten annotators who were trained for a week before the formal annotation. To examine annotation consistency, we select representative images and compare the segmentation mask IoU from our semi-automatic annotation pipeline and those from different annotators labeled from scratch. Overall, the IoU score is 0.93, which shows that our annotation pipeline results are similar to a fully-manual approach. we repeat the same annotation consistency evaluation protocol for the YouTube-VOS dataset, which has similar mask quality (0.89) to our dataset. We refer readers to the supplementary material for details.

3.3. Challenges in Multi-shot VOS

Actor Changes in Different Shots. Cross-shot tracking is challenging due to the on-and-off presence of actors and their sudden appearance and position change (Fig. 5a). We plot a histogram of the number of presence switches for all actors. For appearance change, we compute the cosine distance of pre-trained ResNet-18 features between cropped neighboring frames and average them for each actor for same-shot and cross-shot pairs. We quantize the distance and plot the distance frequency. For position, we compute the center change for same-shot and cross-shot cases and plot the distance. Actors undergo more appearance and position changes across shots than within same-shot frames.

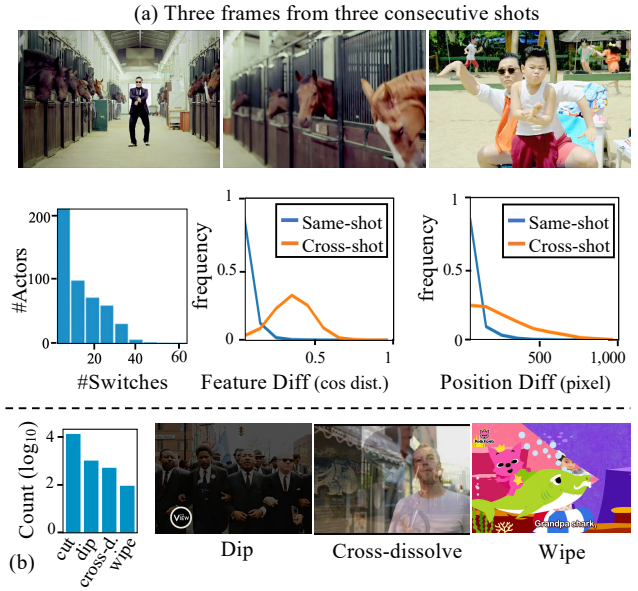

Figure 5: **Challenges in multi-shot VOS.** (a) Even in frames from consecutive shots, actors can switch locations (*e.g.*, a jump cut) and change appearance. We plot the number of switches per actor, and the frequencies of actor appearance feature distances and center location distance for same-shot and cross-shot frames. (b) Beyond the cut shot transition, the dip, cross-dissolve, and wipe transitions can be confusing for models to track and segment actors.

Shot Transitions. Our dataset contains four visual shot transition effects: cut, dip, cross-dissolve, and wipe. We plot a histogram of transition types (Fig. 5b). Dip, cross-dissolve, and wipe are challenging for VOS and VIS models. For example, cross-dissolve transition blends light intensity, which can lead to false linking of different actors.

3.4. Task Setting and Evaluation Metrics

For simplicity, we adopt the unsupervised online setting of VOS for our dataset, which does not need the initial mask input. We use the evaluation metrics defined for the unsupervised track in the DAVIS VOS challenge [7]: region similarity score $\mathcal{J}$ and contour accuracy $\mathcal{F}$. For each video, we use the Hungarian algorithm to match the ground truth video segments to the predicted ones. For each ground truth actor, we average $\mathcal{J}$ and $\mathcal{F}$ with the best-matched proposals. The final score is the weighted average over all ground-truth actors in all videos according to their number of appearances. Due to the annotation difficulty on certain frames, *e.g.*, ambiguous actor boundary in both space and time, we exclude these frames for evaluation.

4. Improved Baseline Model

As a dataset paper, we also provide a baseline method for the community to compare against. In this section, we first examine existing models (Sec. 4.1), and then improve the module designs of one baseline model for both single-shot (Sec. 4.2) and multi-shot (Sec. 4.3) videos.

4.1. Baseline VIS Models

For the unsupervised online actor segmentation task, we can directly apply unsupervised video object segmentation (VOS) methods or video instance segmentation (VIS) methods without using class labels. Empirically, we find it hard to adapt unsupervised VOS methods to our multi-shot dataset, as they either require offline processing [43, 34] or have undesirable performance due to the lack of a tracking module [61]. Thus, we evaluate VIS baseline models developed on the YouTube-VIS dataset [57] whose object categories overlap with ours.

The state-of-the-art VIS methods MaskProp [5] and Propose-Reduce [31] have not publicly released their training codes, making it hard to finetune on YouMVOS. Further, we find the recent transformer-based method VISTR [53] runs significantly slower than the CNN-based methods, making it impractical for long-term videos. Thus, we benchmark MaskTrack R-CNN [57], SipMask [8], and ObjProp [10] that have training code publicly available and run at acceptable speeds for long videos. From these, we choose the ObjProp [10] model as a baseline—best performing among the three—and improve it for both single-shot and cross-shot predictions (Fig 6).

The ObjProp model explicitly tracks instances based on pairwise matching scores between actors on the current frame and those stored in a memory queue. During inference, the final matching score between n-th actor in memory and i-th actor in the current frame is defined as

$$\mathcal{S}_{\text{VIS}}(i,n) = \mathcal{S}_{\text{DET}} + \alpha\mathcal{S}_{\text{CLS}} + \beta\mathcal{S}_{\text{BOX}} + \gamma\mathcal{S}_{\text{TRK}}, \quad (1)$$

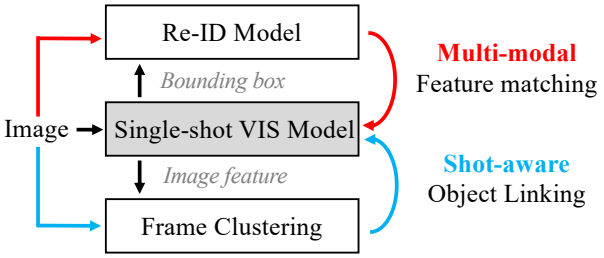

Figure 6: **Improved baseline model.** Given a single-shot VIS model, we first improve its single-shot module design (SMD), then add a pre-trained Re-ID network to better handle actor appearance change, then finally add frame clustering to handle actor position changes across different shots in a multi-shot module design (MMD).

where $\mathcal{S}_{\text{DET}}$ is the detection score, $\mathcal{S}_{\text{CLS}}$ is the class score that is one only if actor i and n have the same class label (zero otherwise), $\mathcal{S}_{\text{BOX}}$ is the bounding box intersection over union (IoU) between actors, and $\mathcal{S}_{\text{TRK}}$ is the tracking score for actor appearance. The hyperparameters $\alpha = 10, \beta = 2, \gamma = 1$ are the same as previous methods [57, 10]. The predicted instance is assigned to the actor with the best $\mathcal{S}_{\text{VIS}}$ score.

4.2. Improved Single-shot Module Design (SMD)

We incorporate existing good practices for single-shot segmentation and tracking to improve the ObjProp model.

Model Architecture. Following MaskProp [5], we use the hybrid task cascade (HTC) framework [13] to improve single-frame instance detection and segmentation results. Further, we split the shared bounding box head and classification head to reduce errors from misclassification.

Memory Management. Most existing VIS methods [57, 50] only keep the latest object features in a queue to link to the current frame predictions. To encourage actor re-detection, we use a memory bank to store information of *a*ll instances of actors detected until that frame. With this, the baseline model can link detected actors in disconnected frames when the detection module fails on frames in between. To achieve a high recall score for actor detection, the baseline model produces many spurious detections that are linked across only few frames forming short tracklets. To speed up the tracking score computation and improve the tracking accuracy by pruning spurious tracklets, we remove tracklets that are shorter than 7 frames during the postprocessing, where the threshold is set empirically.

4.3. Improved Multi-shot Module Design (MMD)

Due to the appearance and location changes of actors across shots, we add multi-modal features [54] for more robust tracking and a frame clustering module to enable bounding box tracking for frames from different shots.

Shot-aware Object Linking (SOL). Linking objects across shots requires two changes: 1) making sure that frame-to-frame bounding box IoU scores $\mathcal{S}_{\text{BOX}}$ for an instance are computed only within the same shot, and 2) refraining from propagating object across shots to fill in empty masks as is done in [10] for same-shot frames. Thus, we detect shot changes with a simple online k-means method that assigns the latest frame either to an existing shot cluster or to a new shot cluster if the cosine distance is above a distance threshold θ_c. For this, we obtain each frame's features from the tracking head module by using a bounding box to cover the input image. Then, we use a long-term memory bank to store the frame feature centers and corresponding frame numbers.

For our multi-shot videos, we add the bounding box IoU score only when the predicted instance i and the instance to match n are in the same shot cluster $g_i = g_n$. Thus, the new scores become

$$\mathcal{S}'_{\text{BOX}}(i,n) = \delta(g_i, g_n)\mathcal{S}_{\text{BOX}}(i,n) \tag{2}$$

where $\delta(g_i, g_n) = 1$ only if frame i and n belong to the same frame cluster, and $\delta(g_i, g_n) = 0$ otherwise.

Multi-modal Feature Tracking (MFT). The ObjProp model consists of a tracking branch [57] to learn the tracking feature $\mathbf{f}_i$ for each instance i. Consequently, the appearance tracking score for the predicted current-frame instance i and the n-th instances in the memory bank is the n-th value in the log softmax of the dot product: $\mathcal{S}_{\text{TRK}}(i,n) = \log\text{softmax}(\mathbf{f}_i^T\mathbf{F})(n)$, where tracking features for N actors is $\mathbf{F} = [\mathbf{f}_0, \mathbf{f}_1, \cdots, \mathbf{f}_{N-1}, \mathbf{0}]$, where $\mathbf{0}$ is for the new actor.

As actors in our dataset are mostly human, we follow Xia *et al.* [54] to improve the appearance tracking by incorporating human pose and face features via pre-trained person re-identification (Re-ID) models. Specifically, we crop the predicted instance patch and extract the pose [47] and face [44] feature vectors as $\mathbf{f}_i^{\text{pose}}$ and $\mathbf{f}_i^{\text{face}}$ To balance the weight among three features, we normalize each feature to obtain the new multi-modal appearance feature $\mathbf{f}'_i = \left[\frac{\mathbf{f}_i}{\|\mathbf{f}_i\|}, \frac{\mathbf{f}_i^{\text{pose}}}{\|\mathbf{f}_j^{\text{pose}}\|}, \frac{\mathbf{f}_i^{\text{face}}}{\|\mathbf{f}_i^{\text{face}}\|}\right]$. To adjust the number of new tracklets, we add a threshold θ_t to the new actor instance, the last element of the tracking score. The updated score is

$$\mathcal{S}'_{\text{TRK}}(i,n) = \log\text{softmax}(\mathbf{f}_i'^T\mathbf{F}' + [\vec{0}, \theta_t])(n) \tag{3}$$

by plugging in the new multi-modal appearance features.

5. Experiments

We present the performance of our baselines adapted to YouMVOS, and then describe the remaining challenges with error analysis (Sec. 5.2), ablation studies (Sec. 5.3), and a comparison to the YouTube-VIS dataset (Sec. 5.4).

Method	Val set		Test set	
	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
SipMask [8]	20.9	20.0	16.1	15.1
MaskTrack R-CNN [57]	21.4	20.2	20.1	19.2
ObjProp [10]	21.8	20.9	21.2	20.0
[10]+SMD (Ours)	25.8	24.8	25.0	24.9
[10]+SMD+MMD (Ours)	**31.8**	**30.8**	**30.9**	**30.6**

Table 2: **Quantitative results on YouMVOS.** The proposed single-shot (SMD) and multi-shot (MMD) module designs significantly improve the baseline ObjProp [10].

5.1. Experiment Setup

Dataset Splits and Metrics. For each of the ten video genres, we randomly split the 20 videos into 14 training, 3 validation, and 3 test videos. In total, YouMVOS has 140 training videos (353 actors and 9,042 shots), 30 validation videos (61 actors and 2,002 shots), and 30 test videos (78 actors and 2,406 shots). The final region similarity score $\mathcal{J}$ and the contour accuracy $\mathcal{F}$ are averaged over all actors in the test videos (Sec. 3.4).

Implementation Details. For the baseline models [57, 8, 43, 58], we use official implementations. The single-shot VIS Models (SipMask, MaskTrack R-CNN, and ObjProp) are pre-trained on YouTube-VIS [57] for 12 epochs, and the tracking and segmentation heads are finetuned on our YouMVOS for another two epochs, with a learning rate of 5×10^{-4} . For the HTC-based models, we additionally finetune the bounding box head. Please see the supplemental materials for more details.

5.2. Benchmark Results

Quantitative Results. The state-of-the-art VIS methods [8, 57, 10] designed for single-shot videos achieve around 20 $\mathcal{J}$ and $\mathcal{F}$ scores for our multi-shot videos (Tab. 2). Adding the single-shot module design (SMD) and multi-shot module design (MMD) improves ObjProp [10] by around 10 points in absolute $\mathcal{J}$ and $\mathcal{F}$ scores for both the validation and test sets. The large performance gap shows the effectiveness of the proposed improved model designs for multi-shot data. Although our proposed adaptations also improve SipMask on YouMVOS to a lesser extent, the single-stage SipMask performs worse than the two-stage MaskTrack R-CNN due to many false positive object proposals.

Qualitative Results. We show the predicted segmentation masks overlaid with frames from different videos (Fig. 7). Specifically, our multi-shot module design (MMD) improves segmentation by using pose Re-ID features when the face is not visible or too small (Fig. 7a), using face Re-ID features to link characters despite costume and scene changes (Fig. 7b–c), and clustering frames into camera shots (Fig. 7d). In

Figure 7: **Qualitative results of our improved baseline on YouMVOS validation set.** (a-d) For success cases, each row shows five sample frames from different shots in the same video sequence. (e) For failure cases, errors can come from poor detection results due to uncommon actor appearance and camera poses, and poor tracking due to special visual effects.

Single-frame	Tracking	$\mathcal{J}$	$\mathcal{F}$
Baseline	Baseline	31.8	30.77
Baseline	Oracle	65.1	62.2
Oracle Box	Baseline	35.2	35.1
Oracle Box	Oracle	80.9	81.3
Oracle Mask	Baseline	44.7	45.5
Oracle Mask	Oracle	100	100

Table 3: **Oracle analysis on YouMVOS val dataset.** Most remaining errors come from actor tracking due to the multi-shot and minute-level long videos in the dataset.

Fig. 7e, we show typical failure cases, including ambiguous appearance (first image), rare camera poses in the music video (second image), drastic change of instance scales in the soccer game video (third image), and an unexpected split-screen video effect that breaks the one-to-one tracking assumption (last two images).

Oracle Analysis. We examine the source of errors in our improved baseline method on the YouMVOS validation data by using oracle—known correct—results for different components (Tab. 3). We focus on tracking error caused by long-term cross-shot videos, and produce results with oracle tracking, oracle bounding boxes, and oracle masks. For oracle tracking, we match per-frame predictions to their closest ground truth objects with a 0.5 IoU threshold, and then aggregate instances using ground truth object identities.

We find that resolving tracking errors leads to a significant 30+ points boost on both region similarity $\mathcal{J}$ and contour accuracy $\mathcal{F}$, showing that errors are caused mostly by misattributions across shots and across long sequences. Second, providing oracle bounding boxes for segmentation improve baseline performance slightly, but cause another 15 points increase when combined with oracle tracking. This suggests that both tracking and localization improvements are required in the future. Finally, providing oracle masks on top produces perfect scores as expected, but without oracle tracking, many attribution errors remain. In summary, there is still large space for improving existing approaches from different perspectives, and *tracking* error is currently the dominating factor for the unsatisfactory scores.

5.3. Ablation Studies

We analyze the effectiveness of each component of the improved baseline on our YouMVOS validation set.

Cumulative Results. We sequentially add the single-shot module design (SMD), shot-aware object linking (SOL), and multi-modal feature tracking (MFT). Adding SMD leads to a 1 point improvement by reducing lost track errors given

(a) Cumulative results.

Single-shot HTC	Single-shot Mem	Multi-shot Tracking (MT) SOL	Multi-shot Tracking (MT) MFT	$\mathcal{J}$	$\mathcal{F}$
				21.8	20.9
✓				24.3	23.2
✓	✓			25.8	24.8
✓	✓	✓		28.7	27.6
✓	✓	✓	+pose [47]	29.6	28.7
✓	✓	✓	+face [44]	29.9	28.9
✓	✓	✓	+pose [47] +face [44]	**31.8**	**30.8**

(b) Clustering hyperparameter for SOL.

θ_c	0.88	0.90	0.92
$\mathcal{J}/\mathcal{F}$	30.6/29.6	**31.8/30.8**	31.5/30.6

(c) Multi-modal Feature Tracking (MFT).

Pose	LightMBN [22]	ABDNet [14]	CoSAM [47]
$\mathcal{J}/\mathcal{F}$	31.2/30.2	31.1/30.2	**31.8/30.8**

Face	VGGFace2 [9]	CASIA-WebFace [60]
$\mathcal{J}/\mathcal{F}$	**31.8/30.8**	31.1/30.2

Table 4: **Ablation studies on YouMVOS validation set.** (a) We show the cumulative improved results by adding each component. Starting from the best model, we modify it with different (b) SOL hyper-parameters and (c) MFT features.

more instances to match (Tab. 4). SOL further leads to a 2 point improvement by using the location consistency of the object within frame clusters. There can be cases where these assumptions cause errors, but we observe overall improvement. Compared with the original MFT tracking features, the face [44] and pose [47] Re-ID features improve performance by 2–3 points each, and by 10 points combined.

Shot-aware Object Linking (SOL). We try three different cosine similarity thresholds (θ) for deciding features in the same cluster in our online nearest neighbor frame clustering method. Intuitively, a bigger θ_c will lead to finer cluster results, but with less utilization of the IoU between bounding boxes in the matching score to take advantage of the locality consistency. We empirically find that $\theta_c = 0.9$ achieves the best overall score (Tab. 4b).

Multi-modal Feature Tracking (MFT). Starting from the best baseline model, we compare popular Re-ID models for pose and face by replacing them one at a time with the current the model. We find that adding pose features [47] achieves around a 3 point improvement over other pose Re-ID features [22, 14, 47]. For face Re-ID features, the FaceNet model [44] pre-trained on VGGFace2 [9] achieves around 5 points improvement over that pre-trained on CASIA-WebFace dataset [60] (Tab. 4c).

5.4. Results on YouTube-VIS Single-shot Dataset

To demonstrate the effectiveness of our improved module design for single-shot videos (SMD), we benchmark upon the YouTube-VIS dataset [57] with the ResNet-50 backbone without using any external data for a fair comparison (Tab. 5). On the validation split, adding our single-shot module design (SMD) significantly boosts model performance by 8.7 mAP for ObjProp [10]. State-of-the-art methods MaskProp [5] and Propose-Reduce [31] do not provide code and so could not be improved; in any case, they use sophisticated and computationally-heavy inference schemes that are more difficult to apply to long videos.

Method	mAP↑	AP_{50}	AP_{75}	AR_1	AR_{10}
MaskTrack R-CNN [57]	30.3	51.1	32.6	31.0	35.5
STEm-Seg [2]	30.6	50.7	33.5	31.6	37.1
SipMask [8]	33.7	54.1	35.8	35.4	40.1
ObjProp [10]	35.1	56.2	38.6	38.6	44.9
VisTR [53]	36.2	59.8	36.9	37.2	42.4
MaskProp† [5]	40.0	-	42.9	-	-
Propose-Reduce† [31]	**40.4**	**63.0**	**43.8**	**41.1**	**49.7**
[10] + SMD (Ours)	39.0	61.2	42.9	38.9	47.6

Table 5: **Benchmark results on YouTube-VIS Val [57].** Our single-shot module design (SMD) boosts baseline model performance to much closer to the state of the art (ResNet-50 backbone without external training data). The training code for methods with † is unavailable during our submission.

6. Conclusion

We have expanded the problem of video object segmentation to long-term multi-shot videos with a new actor-centric 200-video segmentation dataset containing 431K segmentation masks. This provides a new challenge for the computer vision community in addressing objects with location or pose changes, appearance variations, and more complex presence/absence within different narrative structures. Given baseline methods, we analyze sources of error. We discover that cross-shot tracking error is the dominant cause of multi-shot segmentation error. Overall, better analysis of multi-shot videos moves us towards longer-term and more-complex computational video understanding—our YouMVOS segmentation dataset provides an early step towards this goal.

Acknowledgements

This work has been supported by NSF grants NCS-FO-2124179, NIH grant R01HD104969, UKRI grant Turing AI Fellowship EP/W002981/1, and EPSRC/MURI grant EP/N019474/1. We also thank the Royal Academy of Engineering and FiveAI.

References

[1] Ognjen Arandjelovic and Andrew Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *CVPR*, 2005. 3

[2] 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. 2, 8

[3] Lorenzo Baraldi, Costantino Grana, and Rita Cucchiara. A deep siamese network for scene detection in broadcast videos. In *ACM international conference on Multimedia*, 2015. 2

[4] Daniel R Berger, H Sebastian Seung, and Jeff W Lichtman. Vast (volume annotation and segmentation tool): efficient manual and semi-automatic labeling of large 3d image stacks. *Frontiers in neural circuits*, 12:88, 2018. 3

[5] Gedas Bertasius and Lorenzo Torresani. Classifying, Segmenting, and Tracking Object Instances in Video with Mask Propagation. In *CVPR*, 2020. 2, 5, 8

[6] Benjamin Bratt. *Rotoscoping*. Taylor & Francis, 2012. 1

[7] Sergi Caelles, Jordi Pont-Tuset, Federico Perazzi, Alberto Montes, Kevis-Kokitsi Maninis, and Luc Van Gool. The 2019 Davis Challenge on VOS: Unsupervised Multi-object Segmentation. *arXiv preprint arXiv:1905.00737*, 2019. 5

[8] 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. 2, 5, 6, 8

[9] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018. 8

[10] Anirudh S Chakravarthy, Won-Dong Jang, Zudi Lin, Donglai Wei, Song Bai, and Hanspeter Pfister. Object propagation via inter-frame attentions for temporally stable video instance segmentation. *arXiv preprint arXiv:2111.07529*, 2021. 2, 5, 6, 8

[11] Vasileios T Chasanis, Aristidis C Likas, and Nikolaos P Galatsanos. Scene detection in videos using shot clustering and sequence alignment. *Transactions on multimedia*, 2008. 2

[12] Jie Chen, Zhiheng Li, Jiebo Luo, and Chenliang Xu. Learning a weakly-supervised video actor-action segmentation model with a wise selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9901–9911, 2020. 2

[13] 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. 5

[14] Tianlong Chen, Shaojin Ding, Jingyi Xie, Ye Yuan, Wuyang Chen, Yang Yang, Zhou Ren, and Zhangyang Wang. Abd-net: Attentive but diverse person re-identification. In *CVPR*, pages 8351–8361, 2019. 8

[15] Timothee Cour, Benjamin Sapp, Akash Nagle, and Ben Taskar. Talking pictures: Temporal grouping and dialog-supervised person recognition. In *CVPR*, 2010. 3

[16] Mark Everingham, Josef Sivic, and Andrew Zisserman. Hello! my name is... buffy"–automatic naming of characters in tv video. In *BMVC*, 2006. 3

[17] Shubhika Garg, Vidit Goel, and Somesh Kumar. Unsupervised video object segmentation using online mask selection and space-time memory networks. In *The 2020 DAVIS Challenge on Video Object Segmentation-CVPR Workshops*, volume 6, 2020. 2

[18] Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5958–5966, 2018. 2

[19] Bo Han and Weiguo Wu. Video scene segmentation using a novel boundary evaluation criterion and dynamic programming. In *International conference on multimedia and expo*, 2011. 2

[20] Monica-Laura Haurilet, Makarand Tapaswi, Ziad Al-Halah, and Rainer Stiefelhagen. Naming tv characters by watching and analyzing dialogs. In *WACV*, 2016. 3

[21] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[22] Fabian Herzog, Xunbo Ji, Torben Teepe, Stefan Hörmann, Johannes Gilg, and Gerhard Rigoll. Lightweight multi-branch network for person re-identification. *arXiv preprint arXiv:2101.10774*, 2021. 8

[23] Qingqiu Huang, Wentao Liu, and Dahua Lin. Person search in videos with one portrait through visual and temporal links. In *ECCV*, 2018. 3

[24] Qingqiu Huang, Yu Xiong, and Dahua Lin. Unifying identification and context learning for person recognition. In *CVPR*, pages 2217–2225, 2018. 3

[25] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *ECCV*, 2020. 1, 3

[26] Tianrui Hui, Shaofei Huang, Si Liu, Zihan Ding, Guanbin Li, Wenguan Wang, Jizhong Han, and Fei Wang. Collaborative spatial-temporal modeling for language-queried video actor segmentation. In *CVPR*, 2021. 2

[27] Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J Black. Towards understanding action recognition. In *CVPR*, 2013. 2

[28] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9799–9808, 2020. 3, 4

[29] Chao Liang, Yifan Zhang, Jian Cheng, Changsheng Xu, and Hanqing Lu. A novel role-based movie scene segmentation method. In *Pacific-Rim Conference on Multimedia*, 2009. 2

[30] Chung-Ching Lin, Ying Hung, Rogerio Feris, and Linglin He. Video Instance Segmentation Tracking With a Modified VAE Architecture. In *CVPR*, pages 13147–13157, 2020. 2

[31] Huaijia Lin, Ruizheng Wu, Shu Liu, Jiangbo Lu, and Jiaya Jia. Video instance segmentation with a propose-reduce paradigm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1739–1748, 2021. 2, 5, 8

[32] Xiaolong Liu, Yao Hu, Song Bai, Fei Ding, Xiang Bai, and Philip HS Torr. Multi-shot temporal event localization: a benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12596–12606, 2021. 1

[33] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *CVPR*, pages 3623–3632, 2019. 2

[34] Jonathon Luiten, Idil Esen Zulfikar, and Bastian Leibe. Unovost: Unsupervised offline video object segmentation and tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2000–2009, 2020. 2, 5

[35] Arsha Nagrani and Andrew Zisserman. From benedict cumberbatch to sherlock holmes: Character identification in tv series without a script. In *BMVC*, 2017. 3

[36] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019. 4

[37] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 1, 2

[38] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip HS Torr, and Song Bai. Occluded video instance segmentation. *arXiv preprint arXiv:2102.01558*, 2021. 2

[39] Anyi Rao, Linning Xu, Yu Xiong, Guodong Xu, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A local-to-global approach to multi-modal movie scene segmentation. In *CVPR*, 2020. 2

[40] Zeeshan Rasheed and Mubarak Shah. Scene detection in hollywood movies and tv shows. In *Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, 2003. 2

[41] Daniel Rotman, Dror Porat, and Gal Ashour. Optimal sequential grouping for robust video scene detection using multiple modalities. *International Journal of Semantic Computing*, 2017. 2

[42] Yong Rui, Thomas S Huang, and Sharad Mehrotra. Exploring video structure beyond the shots. In *Proceedings. IEEE International Conference on Multimedia Computing and Systems*, 1998. 2

[43] S. Kumar S. Garg, V. Goel. Unsupervised video object segmentation using online mask selection and space-time memory networks. *The 2020 DAVIS Challenge on Video Object Segmentation - CVPR Workshops*, 2020. 2, 5, 6

[44] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 6, 8

[45] Panagiotis Sidiropoulos, Vasileios Mezaris, Ioannis Kompatsiaris, Hugo Meinedo, Miguel Bugalho, and Isabel Trancoso. Temporal video segmentation to scenes using high-level audiovisual features. *Transactions on Circuits and Systems for Video Technology*, 2011. 2

[46] Josef Sivic, Mark Everingham, and Andrew Zisserman. "who are you?"-learning person specific classifiers from video. In *CVPR*, 2009. 3

[47] Arulkumar Subramaniam, Athira Nambiar, and Anurag Mittal. Co-segmentation inspired attention networks for video-based person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 562–572, 2019. 6, 8

[48] Makarand Tapaswi, Martin Bauml, and Rainer Stiefelhagen. Storygraphs: visualizing character interactions as a timeline. In *CVPR*, 2014. 2

[49] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, pages 9481–9490, 2019. 2

[50] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. MOTS: Multi-object tracking and segmentation. In *CVPR*, pages 7942–7951, 2019. 2, 5

[51] Wenguan Wang, Hongmei Song, Shuyang Zhao, Jianbing Shen, Sanyuan Zhao, Steven CH Hoi, and Haibin Ling. Learning unsupervised video object segmentation through visual attention. In *CVPR*, pages 3064–3074, 2019. 2

[52] Wenguan Wang, Tianfei Zhou, Fatih Porikli, David Crandall, and Luc Van Gool. A survey on deep learning technique for video segmentation. *arXiv preprint arXiv:2107.01153*, 2021. 1, 2

[53] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021. 2, 5, 8

[54] Jiangyue Xia, Anyi Rao, Qingqiu Huang, Linning Xu, Jiangtao Wen, and Dahua Lin. Online multi-modal person search in videos. In *European Conference on Computer Vision*, pages 174–190. Springer, 2020. 3, 5, 6

[55] Chenliang Xu, Shao-Hang Hsieh, Caiming Xiong, and Jason J Corso. Can humans fly? action understanding with multiple classes of actors. In *CVPR*, 2015. 2

[56] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. In *ECCV*, 2018. 1, 2, 3

[57] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5188–5197, 2019. 2, 5, 6, 8

[58] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. Efficient video object segmentation via network modulation. In *CVPR*, 2018. 2, 6

[59] Zhao Yang, Qiang Wang, Luca Bertinetto, Weiming Hu, Song Bai, and Philip HS Torr. Anchor diffusion for unsupervised video object segmentation. In *ICCV*, pages 931–940, 2019. 2

[60] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 8

[61] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *AAAI*, 2020. 5