A. Model details

A.1. Segmentation head

We train a segmentation head on top of the frozen OWL-ViT model solely to enforce the one-output-per-pixel constraint required by the OWTA metric. The segmentation head predicts cropped masks within the bounding boxes predicted by the main model. It consists of a ResNet-26 encoder and Hourglass mask heads as described in [4], trained on Open Images V5 [1, 20].

After training this head on the OWL-ViT model, we apply the same (frozen) head on object queries in Video OWL-ViT (without re-training or fine-tuning) to obtain rough segmentation masks.

Example qualitative segmentation masks are shown in Figure 8.

A.2. Architecture

We provide an overview of architecture hyperparameters of Video OWL-ViT in Table 7. We use pre-norm [34] in all transformer layers.

A.3. Data augmentation

We use the following data augmentations for training on TAO-OW: 1) we randomly left-right flip all frames (jointly) in a training clip, 2) we randomly invert the temporal axis, 3) we apply random cropping (jointly across all frames in a clip), and 4) we apply a temporal video mosaic augmentation. All 6-frame clips used for training are randomly sampled from the training videos at 4FPS.

For cropping, we sample a random 480×640 crop of the original video and discard bounding boxes if less than 50% of their original box area remains after cropping. For temporal video mosaic, we take two processed video clips of length 6 (with augmentation as described above), concatenate them along the time axis, and sample a random temporal window of length 6 over the joint sequence. We apply temporal video mosaic to 50% of training samples.

To obtain pseudo-videos from images (incl. individual TAO-OW training frames), we apply a random crop (of size 50% of height and width of the original image) that simulates linear camera motion over the image. We similarly discard bounding boxes if less than 50% of their original box area remains after cropping.

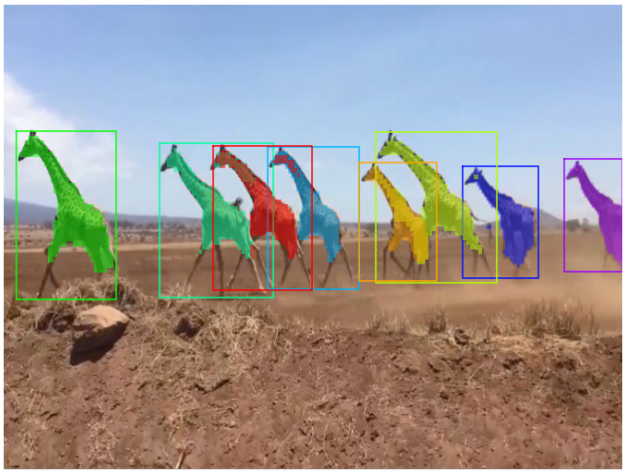

Figure 8: Example of segmentation masks used for enforcing the non-overlap constraint of the OWTA metric.

Table 7: Video OWL-ViT architecture overview.

Backbone	ViT-L/14	
Decoder	Layers	6
	Heads	8
	Hidden dim	1024
	MLP size	4096
	QKV dim	1024
	Dropout rate	0.1
Box head	MLP size	1024
	MLP hidden layers	2
	MLP activation	GELU [15]

A.4. Training

We train Video OWL-ViT using the Adam [17] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and with a batch size of 32 and a learning rate of 3e-6 for 100k training steps. We clip gradients to a maximum norm of 1. We linearly "warm up" the learning rate over the first 1k steps and decay it to 0 over the course of training using a cosine schedule.

For our loss, we use the same hyperparameters as OWL-ViT [26], i.e. equal weighting between bounding box, gIoU, and classification losses, and focal loss coefficients of $\alpha = 0.3$ and $\gamma = 2$.

For simplicity, we do not filter class labels in upstream text-image and detection pre-training, i.e. objects of classes that are considered "unknown" in the TAO-OW video tracking setting can appear in static images during training, but are never seen in natural video. We verified that filtering these classes during upstream pre-training has negligible effect on our reported metrics.

B. Additional results

B.1. Backbone size

To evaluate the effect of model size, we compare our default Video OWL-ViT model, which uses a ViT-L/14 backbone, to a model variant with a smaller backbone (ViT-B/16 at 768×768 resolution). Our results in Table 8 indicate clear

Table 8: **TAO** open world tracking with Video OWL-ViT for different ViT backbone size. All metrics in %.

		LVIS		Known			Unknown		
ViT	**Resolution**	AP	APr	OWTA	D. Re.	A. Acc.	OWTA	D. Re.	A. Acc.
B/16	768	27.2	20.6	55.2	64.3	48.9	41.6	48.6	37.9
L/14	672	33.4	31.8	59.0	69.0	51.5	45.4	53.4	40.5

performance gains when using the larger ViT-L/14 backbone across all metrics, incl. upstream LVIS detection performance.

B.2. Qualitative results

We show further qualitative results of high scoring tracks for Video OWL-ViT and our tracking-by-detection baseline in Figure 9 (TAO-OW) and Figure 10 (YT-VIS). Qualitative results in video format are provided in the supplementary zip file. Video OWL-ViT generally maintains consistent tracks and avoids transfer of instance predictions across semantically different objects compared to our tracking-by-detection baseline.

Figure 9: Qualitative examples for Video OWL-ViT detection and tracking of multiple instances on the **TAO-OW** validation set. Tracking-by-detection (odd rows) vs Video OWL-ViT (even rows). Known classes include: cat, dog, zebra. Unknown classes include: fish, rabbit, hippopotamus. Colors uniquely correspond to query IDs. Numbers indicate objectness scores. Only the first 6 frames/seconds of each video are shown.

Figure 10: Qualitative examples for Video OWL-ViT detection and tracking of multiple instances on the **YT-VIS** validation/test sets. Tracking-by-detection (odd rows) vs Video OWL-ViT (even rows). Known classes include: dog, car, airplane. Unknown classes include: duck, shark. Colors uniquely correspond to query IDs. Numbers indicate objectness scores. The video clips are shown at a reduced frame rate (1 FPS).