Matching Anything by Segmenting Anything
—— Supplementary Material ——

Siyuan Li[1] Lei Ke[1] Martin Danelljan[1] Luigi Piccinelli [1]
Mattia Segu[1] Luc Van Gool[1,2] Fisher Yu[1]

[1]ETH Zürich [2]INSAIT

In this supplementary material, we provide additional ablation studies and qualitative results of our fast proposal generation and of our association. We also elaborate on our experimental setup, method details, and training and inference hyper-parameters.

The supplementary material is structured as follows:

A. Effectiveness on Other Backbones

In our main paper, we introduced four method variants, each building upon foundational detection and segmentation models: SAM-ViT-B, SAM-ViT-H, Grounding-DINO, and Detic. Notably, the latter two variants leverage the Swin-B backbone. Our MASA training pipeline and adapter have shown great adaptability to a range of variables, including variations in backbone structures, pre-training methods (such as detection or segmentation), and the diverse datasets employed in training these foundational models.

A critical observation from our study is the reliance of these variants on large, complex backbones and their pre-training on extensive datasets. This reliance poses an important question about scalability and efficiency: Can our method sustain its effectiveness when applied to smaller, more streamlined backbones, like the ResNet-50, especially with standard ImageNet pre-training? To explore this, we devised a new variant, "**Ours-R50**," which integrates the ResNet-50 backbone pre-trained on the ImageNet classification task (IN-Sup R50). This new variant maintains the MASA adapter architecture from our main research, adhering to the identical training protocol established by our initial four variants.

We have assessed the performance of Ours-R50 across various benchmarks, including BDD MOTS, BDD MOT, and TAO TETA. The quantitative results, detailed in Tables 1, 2, and 3, demonstrate the efficacy of Ours-R50. These findings are significant as they suggest that our approach can be effectively adapted to smaller backbones, offering potential for more efficient and scalable solutions in detection and segmentation tasks.

BDD MOTS: For BDD MOTS (Table 1), Ours-R50, equipped with the ResNet-50 backbone and our MASA training approach, not only outperforms the UNINEXT-H model with a +0.2 mIDF1 and +0.4 AssocA but also shows minimal performance drop compared to the strongest variant, Ours-SAM-H (-1 mIDF1 and -0.9 AssocA). This highlights our method's ability to yield competitive instance embeddings, even without the advanced features provided by larger, specialized models.

BDD MOT: In the BDD MOT benchmark (Table 2), Ours-R50 surpasses ByteTrack in terms of IDF1 (+0.9) and AssocA (+0.2) scores. Its performance is on par with our other variants, showing only a slight decrease compared to Ours-Detic (-1 mIDF1 and -1.2 AssocA). These results reaffirm the adaptability of our method across various backbone architectures and pre-training environments.

TAO TETA: Evaluating on TAO TETA (Table 3), Ours-R50, with its standard ResNet-50 backbone, continues to perform robustly. It closely matches the fully supervised TETer model, with only a slight decrease in AssocA (-1). This performance, consistent with our other variants, further validates the generalizability of our MASA approach across different backbones and pre-training methodologies.

B. Zero-shot Evaluation on YoutubeVIS

In this section, we evaluate our association method in a zero-shot setting on the Youtube-VIS 2019 [27] benchmark. To be

Table 1. State-of-the-art comparison on BDD MOTS. All methods in the table use the same object detection observations. AssocA, mIDF1, and IDF1 mainly focus on the association quality.

Method	mIDF1↑	AssocA↑	TETA↑	mMOTSA↑	mHOTA↑
Fully-supervised, in-domain					
UNINEXT-H [26]†	48.5	53.2	53.6	35.7	40.6
Self-supervised, zero-shot					
Ours-Detic†	49.5	53.5	54.4	**36.4**	40.2
Ours-Grounding-DINO†	48.6	52.3	54.0	36.1	40.0
Ours-SAM-B†	49.2	53.9	**54.8**	35.2	40.7
Ours-SAM-H†	**49.7**	**54.5**	54.7	35.8	**40.8**
Ours-R50†	48.7	53.6	54.7	35.2	40.4

Table 2. State-of-the-art comparison on BDD MOT val set. All methods in the table use the same object detection observations. Our training method learns the most robust and accurate association.

Method	mIDF1↑	IDF1↑	TETA↑	AssocA↑	mMOTA↑
Fully-supervised, in-domain					
ByteTrack [30]†	54.8	70.4	**55.7**	51.5	**45.5**
Self-supervised, zero-shot					
Ours-Detic†	**55.8**	71.3	54.4	**52.9**	44.6
Ours-Grounding-DINO†	55.6	**71.7**	54.5	52.7	44.5
Ours-SAM-B†	55.6	71.6	54.0	52.6	44.1
Ours-SAM-H†	55.3	**71.7**	54.2	51.9	44.5
Ours-R50†	54.8	71.3	54.0	51.7	44.2

Table 3. State-of-the-art comparison on TAO MOT. All methods in the table use the same object detection observations.

Method	AssocA	TETA	LocA	ClsA
Fully-supervised, in-domain				
TETer [18]†	36.7	34.6	**52.1**	15.0
Self-supervised, zero-shot				
Ours-Detic†	36.4	34.7	51.9	**15.8**
Ours-Grounding-DINO†	**37.6**	**34.9**	51.8	15.4
Ours-SAM-B†	36.6	34.5	51.9	15.1
Ours-SAM-H†	36.4	34.5	51.8	15.4
Ours-R50†	35.7	34.1	**52.1**	15.0

Table 4. State-of-the-art comparison on Youtube-VIS 2019. † represents that we provide the same object detection observations. Our method does not train using any image or any annotation from Youtube-VIS 2019.

Method	Zero-shot	Association Label	Video	VIS2019 val	
				AP	AP_{75}
VisTR [21]	✗	✓	✓	36.2	36.9
MaskProp [1]	✗	✓	✓	40.0	42.9
IFC [15]	✗	✓	✓	42.8	46.8
SeqFormer [23]	✗	✓	✓	47.4	51.8
IDOL [24]	✗	✓	✓	49.5	52.9
MFVIS [16]	✗	✓	✓	46.6	49.7
VITA [14]	✗	✓	✓	49.8	54.5
UNINEXT-R50 [26]†	✗	✓	✓	**53.0**	**59.1**
Ours-SAM-B†	✓	✗	✗	51.8	58.1

specific, we test our MASA adapter with SAM-ViT-B as the base model directly on Youtube-VIS 2019 for association. Our method uses the same object detection observations as the state-of-the-art VIS method UNINEXT-R50 [26]. As shown in Table 4, our method achieves comparable performance with SOTA UNINEXT trained with the in-domain YoutubeVIS data, while outperforming all other approaches significantly. This outcome underlines the robust zero-shot association capabilities of our method, highlighting its effectiveness in scenarios without domain-specific training.

C. Visualization of Instance Embeddings

In Figure 1, we use t-SNE to visualize instance embeddings learned in different ways. We compare self-supervised approaches such as MoCo-v2 [4], VFS [25], and DINO [2], alongside two base models: SAM ViT-B [17], originally pre-trained on SA-1B for segmentation tasks, and IN-Sup R50 [13], initially pre-trained on ImageNet for image classification. Additionally, we present embeddings from fully supervised in-domain video models [18] and the same base models enhanced with our MASA adapters. In these visualizations, instances that share the same ground-truth ID are represented in the same colors. We use the BDD100K sequence as the data source.

Our observations indicate that the embeddings from the original SAM, IN-Sup R50, as well as the self-supervised methods like MoCo, VFS, and DINO, do not consistently separate different instances within certain complex scenarios, as highlighted by the instances marked in green, orange, and yellow. In contrast, by applying our MASA adapter to the original SAM ViT-B and IN-Sup R50 features, the resulting adapted embeddings exhibit a successful delineation of distinct instances. This performance is comparable to that of fully supervised methods that have been trained on labelled in-domain videos. Significantly, our method achieves these results without any labelled in-domain video data, demonstrating its considerable potential for robust instance-level correspondence learning.

D. Domain Gap and Adaptation

Except for previously mentioned applications, MASA can also serve as a useful domain adaption method for instance association. To be specific, due to the domain gaps such as object categories, scenarios, and lighting conditions, trackers trained on data of domain A may suffer from performance drop when evaluating on domain B. For example, compared with BDD [29], TAO [10] covers much more diverse scenarios and object categories. Thus, we choose BDD100K [29] as the source domain and TAO [10] as the target domain. Then we train two separate models with the same architecture as TETer [18] using labeled data of BDD and LVIS+TAO [10, 12] respectively. These two models are represented by the blue and green bars in Figure 2. Please note that when evaluating their associating ability on TAO, they use the same object detection observations. As shown in

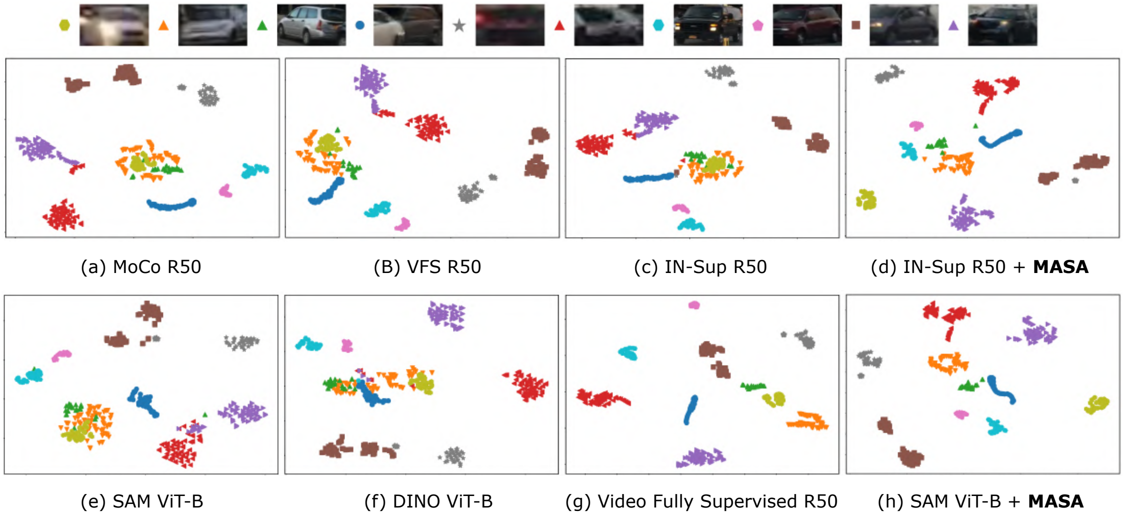


Figure 1. t-SNE visualization demonstrates the distinctiveness of instance embeddings across various methods on a selected BDD100k sequence. The embeddings generated by our method (indicated by MASA-enhanced models) exhibit greater inter-instance separation and tighter intra-instance clustering than other self-supervised methods (MoCo, VFS, DINO) and the original supervised methods (IN-Sup, SAM). This enhanced discrimination highlights the effectiveness of our adapted features for downstream tasks.

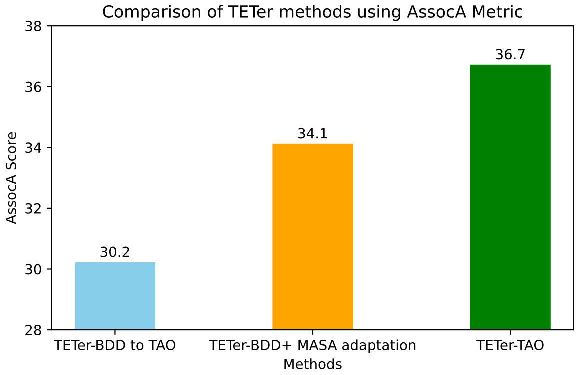


Figure 2. Domain adaptation for TETer with MASA.

Figure 2, directly applying embeddings trained on BDD to TAO (blue bar) leads to poor AssocA, which is 6.5% lower than the model trained on in-domain TAO (green bar). To alleviate this performance gap, we fine-tune the track head of the original TETer model represented by the blue bar with the MASA training pipeline, while freezing all other parameters (orange bar). Specifically, we only fine-tune the model using unlabeled images of LVIS and TAO, while not using any original TAO annotation. As shown in Figure 2, compared with the blue bar, the orange bar achieves an improvement of 3.9% on AssocA, reducing the domain gap by 60%. This demonstrates that MASA can effectively improve the association performance in out-of-domain scenarios, only requiring unlabeled images from the target domain.

E. Impact of Photometric Augmentation

In Section 4.3 of our main paper, we focused on various geometric augmentations, including random affine transformations, and large-scale jittering. We also use MixUp to enhance the instance diversity and simulate the occlusion effect. This section delves into the impact of additional photometric augmentation. We specifically examine the effects of motion blur, Gaussian noise, snow, fog, and brightness adjustments. Photometric augmentations are characterized by their ability to modify pixel values in an image. These alterations often mimic changes in environmental factors such as lighting and weather, impacting how scenes are captured by cameras. Unlike geometric augmentations that change the spatial arrangement of pixels through rotation, scaling, or cropping, photometric augmentations do not alter the structural integrity of objects within an image. Figure 3 illustrates these augmentations visually.

We maintained the same training regimen as our ablation study in the main paper, and using SAM-ViT-B as the foundational model for our experiments. Table 5 presents the results, indicating that the inclusion of photometric augmentation yields only modest improvements. We observed a marginal increase of +0.1 mIDF1 and +0.2 AssocA on the BDD dataset and +0.1 AssocA on the TAO dataset. Consequently, these augmentations are not included as a default in our methodology to achieve a better balance between performance improvement and the potential increase in computational complexity.

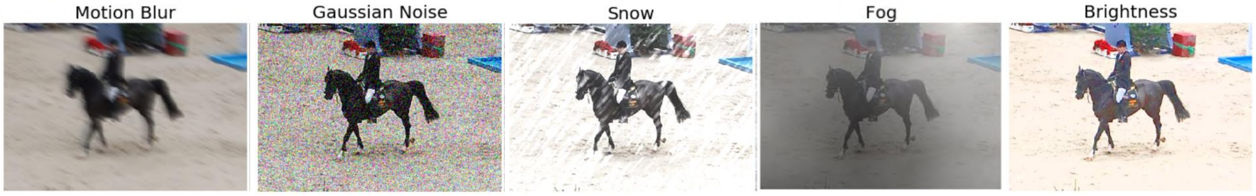


Figure 3. Beyond the strong geometric augmentations utilized in the main study, this figure presents an exploration of five additional photometric augmentations: motion blur, Gaussian noise, snow, fog, and brightness adjustments.

Table 5. Assessing the Impact of Additional Photometric Augmentation. The standard augmentation set includes flipping, color jittering, and random cropping. The more intensive "Strong Aug" set comprises random affine transformations, large-scale jittering, and mix-up techniques. The photometric augmentation set tested here includes motion blur, Gaussian noise, snow, fog, and brightness adjustments.

Standard Aug	Strong Aug	Photometric Aug	BDD MOT		TAO
			mIDF1	AssocA	AssocA
✓			48.2	43.9	28.5
✓	✓		54.9	51.9	35.8
✓	✓	✓	55	52.1	35.9

F. Comparison of Proposal Diversity

In our main paper, we assessed different proposal generation mechanisms within the context of association learning. Specifically, we focused on training using raw images from the BDD dataset. We experimented by replacing SAM in our MASA pipeline with Mask2former-SwinL, pre-trained on the COCO dataset (see [5]). As detailed in Table 9c of the main paper, the model utilizing SAM's proposals demonstrated enhanced performance. This was evident both in in-domain tracking on the BDD dataset and in zero-shot tracking scenarios on the TAO dataset. Such findings highlight the crucial role of SAM's dense and diverse object proposals in facilitating effective contrastive similarity learning.

Further, we present visual comparisons of the proposals generated by Mask2former and SAM in Figure 4. These comparative visualizations distinctly showcase the superior diversity in SAM's proposals relative to those generated by Mask2former. SAM exhibits an enhanced ability to identify a wider array of instances within raw images, providing proposals with greater diversity. This diversity is pivotal in instance similarity learning and significantly contributes to the out-of-domain generalization capabilities of the learned instance representations.

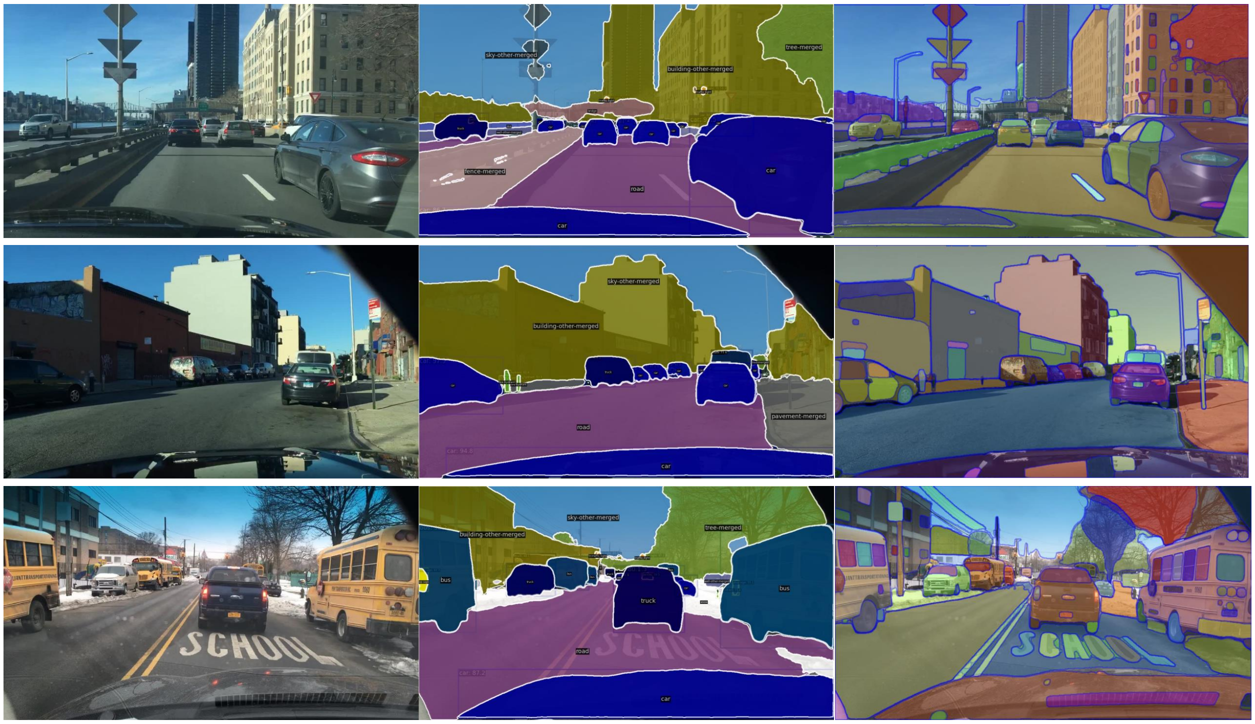
G. Compare with Self-Supervised Methods

The task of extracting meaningful information from purely unlabeled images is notably challenging. UniTrack [22] has showcased the potential of self-supervised trained representations, such as MoCo [4] and VFS [25], in generalizing to various tracking tasks across different domains. However, as depicted in Figure 5, current self-supervised methods predominantly employ contrastive training with clean, object-centered images or videos. In particular, VFS trains on the Kinetics dataset, while MoCo and DINO utilize ImageNet.

However, these approaches primarily focus on frame-level similarities and fail to leverage instance information effectively. Consequently, they struggle to learn accurate instance representations in complex domains with multiple instances appearing together, demonstrating a notable weakness in extracting robust and generalized representations.

Visualization of Object-Centered Training Data We visualize the training data of VFS [25], the Kinetics dataset, and compare it with the driving videos from BDD100K in Figure 6. Kinetics, being an action recognition dataset, ensures the presence of instances throughout its videos by focusing on contained actions. Centred entities in Kinetics videos usually remain consistent over time, making VFS's sampling strategy suitable for Kinetics. In contrast, BDD100K driving videos present a more dynamic and unpredictable environment. These videos frequently feature objects that enter and exit the frame, leading to a significant variation in the presence of instances across different frames. This characteristic of BDD100K poses a challenge as two frames sampled from the same video may not share the same instances, highlighting a fundamental difference in the nature of training data between the two datasets.

Training with Different Data Sources For a fair comparison, when comparing our method with other self-supervised counterparts in Table 6 of the main paper, we train all methods using the same raw training images (BDD and COCO), which are not object-centered and usually contain multiple instances in complex environments. In this section, we also present the tracking performance of those self-supervised methods using their original object-centered training data. As shown in the table below, the AssocA of MoCo trained on images from BDD and COCO remains relatively stable compared to its original version trained on ImageNet, with only a slight drop on the BDD MOT dataset. However, for VFS, training on images with multiple instances leads to a significant performance drop of 15.9 AssocA on BDD MOT and 12.7 AssocA on TAO, respectively. The reason is as



(a) Original Images (b) Mask2Former Predictions (c) SAM Predictions

Figure 4. Comparison between predictions of Mask2Former and SAM.While Mask2Former is limited to identifying 'things' and 'stuff' from categories included in its training set, SAM demonstrates a broader detection scope. It effectively identifies objects of more diverse classes and finer granularity, such as windows, wheels, and traffic signs.

follows: VFS considers frames from the same video as positive samples and frames from different videos as negative samples. This strategy is reasonable for Kinetics but not for BDD, as demonstrated in Figure 6. Specifically, centred entities in Kinetics videos usually do not change over time, but in BDD videos, objects frequently move in and out of frames. Two frames from the same BDD video may not contain the same instances at all. Lastly, in DINO's training process, it forces representations of two augmented views from the same image to be similar without explicitly using negative samples. However, for images in BDD and COCO, two augmented views may contain many different instances, considering the complex scenes of these two datasets. This training strategy may cause the learned embeddings to be less discriminative.

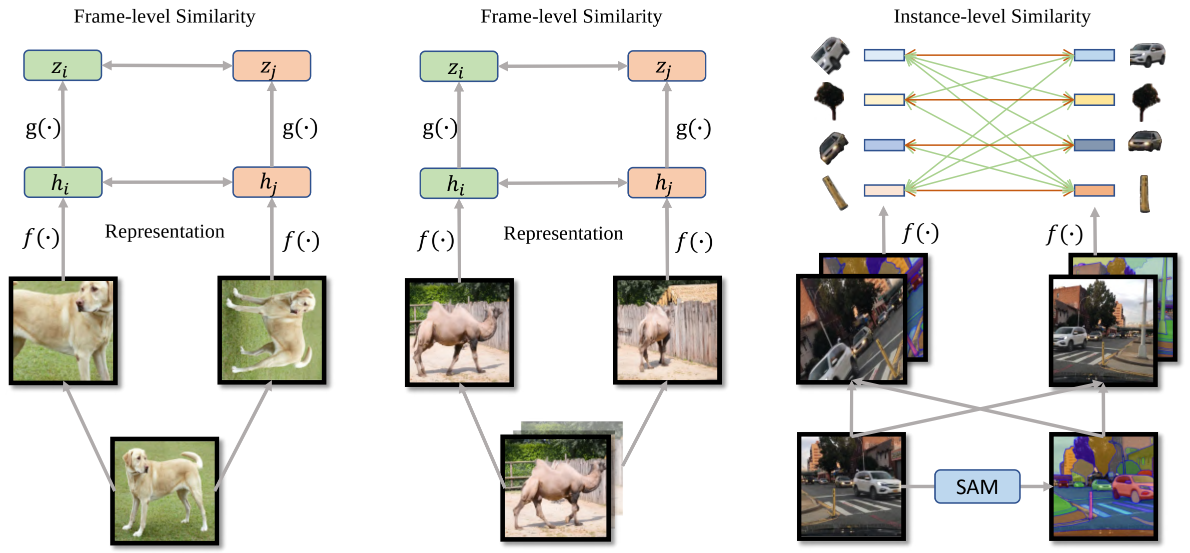
Our approach, which leverages instance-level knowledge from the pre-trained SAM, moves beyond frame-level similarity to embrace a more nuanced instance-level similarity. The strong results obtained underscore the effectiveness of our proposed methods in learning robust representations for tracking purposes.

Table 6. Compare with self-supervised based methods. All methods use the same detection observations for testing. Object-centred data means ImageNet for MoCO and DINO, and Kinetics for VFS.

Method	Video	BDD MOT		TAO	BDD MOTS	
		AssocA	mIDF1	AssocA	AssocA	mIDF1
Train on object-centred data						
VFS	✓	45.1	49.9	31.8	50.3	44.9
MoCov2	✗	44.1	48.6	31.1	50.5	45.6
DINO	✗	41.7	46.5	26	46	40.5
Train on BDD & COCO						
VFS	✓	29.2	35.0	19.1	30.7	30.1
MoCov2	✗	42.7	46.7	30.7	51	45.3
DINO	✗	23.1	16.8	12.9	20.2	22.2
Ours-SAM-B	✗	**51.9**	**54.9**	**35.8**	**53.7**	**49.1**

H. Comparison with VOS-based Methods

The recent segmentation foundation model, SAM, has demonstrated exceptional ability in segmenting any object. However, simultaneously tracking all instances generated by SAM in videos remains a challenging task. Current methods typically employ SAM as a mask generator for the first frame of a video, then apply off-the-shelf video object segmentation (VOS) methods to propagate the initialized mask to subsequent frames [6–8, 28]. One notable method, Deva [6], utilizes XMem [7] for mask propagation to track multiple instances simultaneously. However, these methods encounter



(a) Self-supervised on images (b) Self-supervised on videos (c) Ours

Figure 5. Comparison of self-supervised representation learning methods for object association. (a) Traditional methods, such as SimCLR [3], MoCo [4], focus on learning representations by leveraging frame-level similarity. They utilize augmented views of entire images to extract meaningful features. These methods often struggle with complex scenarios involving multiple objects. The reliance on frame-level similarity can be limiting in environments where object-centric learning is crucial. (b) Methods like VFS [25] take a different route by extracting positive pairs from different frames within the same video. This approach aims to capture temporal consistency and object dynamics. Similar to traditional methods, it also requires clean, object-centred video data. The complexity increases significantly in multi-object environments, where distinguishing between different objects becomes challenging. (c) Our method innovatively combines data augmentation with SAM's [17] mask generation technique. This synergy allows for learning dense instance-level correspondences from unlabelled images. By focusing on dense correspondences at the instance level, it can effectively disentangle and learn from intricate object interactions and dynamics in complex environments.

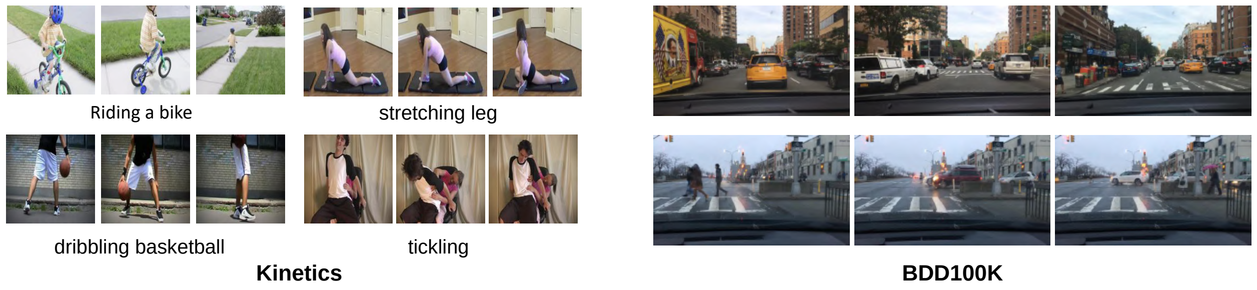


Figure 6. Comparison between Kinetics and BDD100K videos. Kinetics, as an action recognition dataset, ensures that actions are contained within selected videos, thus guaranteeing the presence of instances throughout the video. Centred entities in Kinetics videos usually do not change over time. This makes VFS's sampling strategy reasonable for Kinetics. However, in BDD videos, objects get into and out of the frames frequently. Two frames sampled from the same video may not contain the same instances at all.

several key disadvantages.

Inadequate Mask Propagation Quality: Trained on relatively small-scale video segmentation datasets, these methods experience substantial domain gaps when tasked with tracking any object in any domain, resulting in inadequate mask propagation quality. Our main paper illustrates that our method significantly outperforms Deva [6] in zero-shot testing across various multiple object tracking benchmarks, especially in driving scenes, which are out-of-domain for both Deva and our method. We further provide a qualitative comparison in Figure 7. Additional video comparisons can be found in the provided video file. Testing Deva in the driving domain, which differs significantly from its training data, results in poor mask quality and accumulating errors over time. Moreover, there is no effective mechanism to handle the rapid entry and exit of objects in a scene, a common occurrence in real-world applications like autonomous driving. In contrast, our method exhibits stable performance

Table 7. Performance Comparison on the UVO Dataset for tracking objects and their parts. This table presents a detailed analysis of tracking performance using the UVO dataset. VOS-based methods like Deva have to resolve overlaps by assigning each pixel to a unique instance. Tracking parts leads to an incomplete representation of object masks on UVO, thus affecting performance negatively. In contrast, our method, capable of handling multiple granularities, tracks both entire objects and their parts without compromising performance on the UVO dataset.

Track	Method	AR100
video	*Zero-shot test*	
	Deva-SAM-H: track all instances (with parts) [6]	19.4
	Deva-SAM-H: track only whole objects (no parts) [6]	36.0
	Ours-SAM-H: track all instances (with parts)	**37.5**

in such scenarios.

Difficulty in Managing Multiple Granularities of Pixels: Furthermore, these methods are primarily developed for video object segmentation (VOS) tasks, which typically involve videos and annotations of single, rather than multiple, diverse objects. As a result, most VOS-based approaches are designed to track only one instance at a time. While recent advancements like those in [6, 28] allow for the simultaneous tracking of multiple instances, they often work on the premise that each pixel is part of a single instance. This overlooks complexities in pixel granularity, where a pixel may be part of multiple instances depending on the level of granularity—a common situation in the outputs of SAM, as depicted in Figure 8. This issue is further illustrated using the UVO dataset, which contains only coarse object-level annotations, often omitting finer details of object parts.

We apply SAM to generate mask predictions for each frame in the UVO dataset for both methods. To track objects segmented by SAM, a VOS-based method like Deva has to resolve overlaps by assigning each pixel to a unique instance. For example, if a group of pixels belongs to a part of an object, it must decide whether to track the part or the whole object. Assigning pixels to a part implies that the corresponding object is partially excluded, as shown with the cars in Figure 8. Conversely, assigning pixels to the object results in the removal of the part mask. We present the quantitative results of these scenarios on the UVO dataset in Table 7. Tracking parts leads to an incomplete representation of object masks on UVO, thus affecting performance negatively. In contrast, our method, capable of handling multiple granularities, tracks both entire objects and their parts without compromising performance on the UVO dataset.

I. More Qualitative Results

We provide **a video file** containing our qualitative tracking results on multiple domains. Here we provide some visualization results regarding fast proposal generation and dense object association.

I.1. Fast Proposal Generation

In Figure 9, we compare the segmentation quality of our fast proposal generation with SAM's original everything mode on raw images from COCO validation set. By default, we output 300 bounding boxes per image, and use a bounding box NMS with 0.5 threshold as the only post-processing during inference. The results show that our fast proposal generation can achieve similar segmentation quality to the everything mode of SAM, despite using much less time.

I.2. Open-Vocabulary Tracking

We show qualitative results of open-vocabulary tracking in Figure 10. We observe that our method does well on tracking, and is able to generalize even to very exotic classes, such as minions. More results can be found in the provided video.

I.3. Joint Segment and Track Everything

We provide qualitative results on our joint segmentation and tracking models. Since we learn proposal generation and association in a joint way, it makes our model capable of segmenting and associating anything in videos. Figure 11 shows the qualitative association performance using our self-generated proposals. We notice that although we can learn strong associations using MASA, it is very difficult to generate consistent proposals across frames. For example, we can see the missing segmentation for the building on the left in the second row. Those inconsistent detections will lead to severe flickering effects when visualising the results on videos. This indicates we still need further efforts on consistent proposal generation for robust detecting objects in videos.

J. Implementation Details

We provide more details regarding our model architecture, training, and inference.

J.1. Architecture Detail

MASA Adapter The MASA Adapter comprises two main parts. The first part involves the construction of a feature pyramid and dynamic feature fusion. The second part is the FasterRCNN-based detection head for the object prior to distillation and the track head for producing tracking features. The construction process for the feature pyramid varies depending on the backbone used. These variations are detailed in the respective sections for each model. The dynamic feature fusion employs standard deformable convolution, as outlined in [32], to aggregate information across spatial locations and feature levels. Additionally, task-aware attention and scale-aware attention from [9] are utilized for SAM-based models. In total, three fusion blocks are established for the feature fusion process.

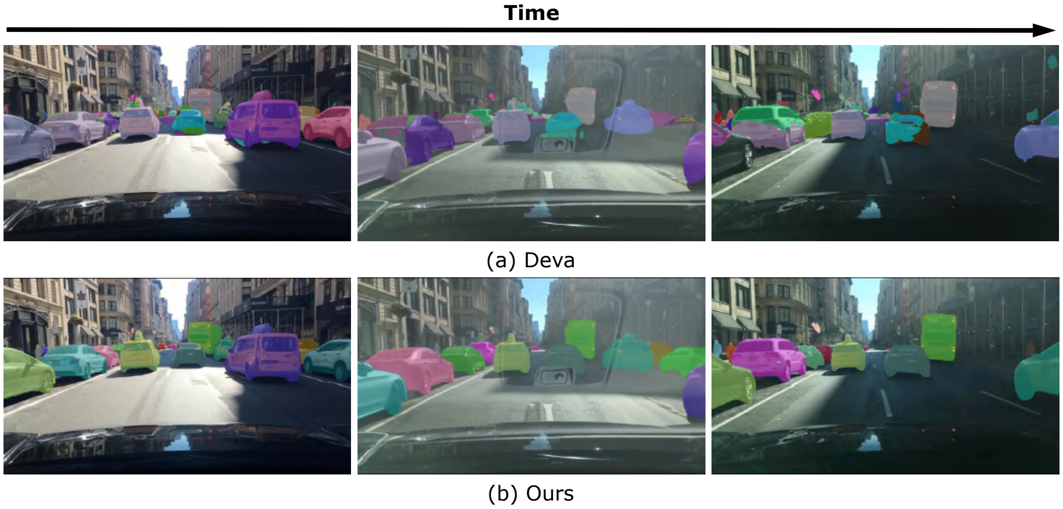


Figure 7. Qualitative Comparison between Our Method and Deva [6] on BDD100K. This figure illustrates the challenges Deva faces in driving scenarios, a domain beyond its training environment. Key issues include inadequate mask propagation and an increasing incidence of false positives over time.

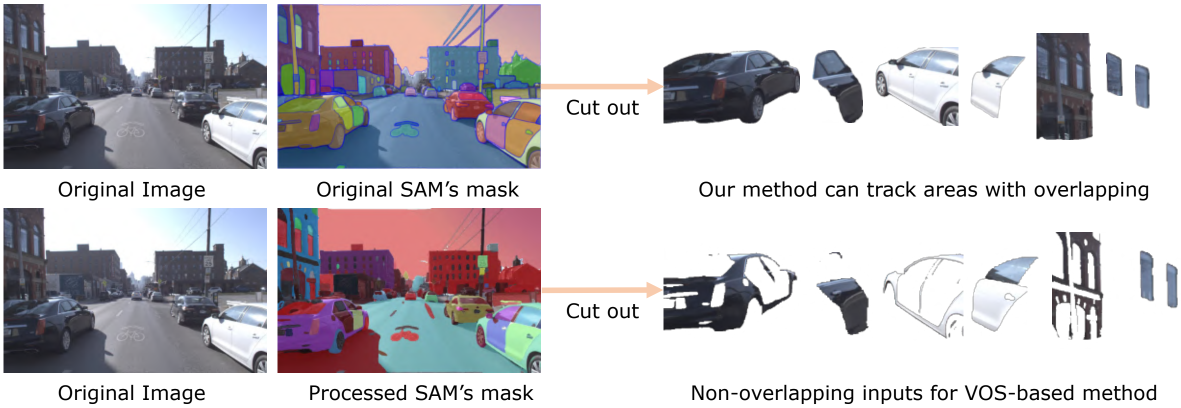

Cut out
Original Image
Original SAM's mask
Our method can track areas with overlapping
Cut out
Original Image
Processed SAM's mask
Non-overlapping inputs for VOS-based method

Figure 8. Challenges of VOS-Based Methods with Multi-Granular Pixel Overlaps. This figure illustrates the complexity encountered when dealing with overlapping masks in SAM's output, where a single pixel may be associated with multiple instances at different granularities. Traditional VOS methods, operating under the assumption that each pixel belongs to only one instance, often resort to heuristics to resolve these overlaps, as depicted in the second row. In contrast, our method effectively handles such overlapping masks, showcasing its adaptability in complex scenarios.

The FasterRCNN-based detection head includes a region proposal network and a class-agnostic box regression head. The track head comprises four convolutional layers and one fully connected layer, used to generate instance embeddings.

Ours-Detic We utilize the pre-trained Detic [31] model with Swin-B [20] as the backbone. The pre-trained model adheres to the open-vocabulary object detection setup described in [11], where rare classes from LVIS are excluded from training. We freeze the Detic Swin-B backbone and employ the standard FPN for constructing the feature pyramid. Specifically, we extract features from the 4^{th}, 22^{nd}, and 24^{th} blocks of the Swin-B backbone. Subsequently, we integrate the dynamic feature fusion atop the feature pyramid to learn tracking features through detection distillation and instance contrastive learning.

Ours-Grounding-DINO We employ the pre-trained Grounding-DINO [19] model with Swin-B [20] as the backbone. The Swin-B backbone is frozen, and we use the standard FPN to construct the feature pyramid. Apart from the differing pre-training and window sizes for the Swin backbone, all learnable components are identical to Ours-Detic.

Ours-SAM-B This model is based on SAM, with all original SAM components frozen. To obtain multi-level hierarchical features from the plain ViT backbone of SAM, we extract feature maps from the outputs of the 3^{rd}, 6^{th}, 9^{th}, and 12^{th} blocks. Transposed Convolutions are used to upscale the feature map from the 3^{rd} block by $4\times$ and from the 6^{th} block by $2\times$. We maintain the 9^{th} feature map as is, and downscale the feature map from the 12^{th} block by $1/2$ using MaxPooling. This approach yields hierarchical features with scale ratios of $\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$. The remainder of the model mirrors the two models mentioned above.

Ours-SAM-H The learnable portion is largely similar to Ours-SAM-B. The sole distinction is that we extract features from the outputs of the 8^{th}, 16^{th}, 24^{th}, and 32^{nd} blocks to construct the feature pyramid.

J.2. More Training Details

For SAM-based models, we turn off MixUp augmentation in the last two epochs. After that, we finetune the track heads of the SAM-based models while freezing the other parts with all augmentations for 6 epochs.

For training our model with any raw image collection, the following pipeline is utilized. Initially, the 'everything' mode of SAM is employed to generate training data on raw images offline, using the SAM-ViTH model to ensure higher quality. We adhere to the default SAM settings, which involve using 32 sampling points along each side of an image. Additionally, an Intersection over Union (IoU) prediction threshold of 0.88 is applied to filter out low-quality predictions. Subsequently, small disconnected regions and holes in masks are removed. Bounding box Non-Maximum Suppression (NMS) is also used to eliminate overlapping predictions with a threshold of 0.7. In our ablation studies, this pipeline is applied to generate data on raw COCO and BDD100K images.

J.3. Inference with Given Observations

Notably, during testing on UVO, in addition to using proposals generated by our fast-segmenting everything mode, we also incorporate the same per-frame mask observation as employed in [6]. This inclusion aims to minimize the temporal inconsistency in SAM's mask predictions on videos.

J.4. Inference Details

Overall, our inference scheme is illustrated in Algorithm 1. In terms of similarity computation, we provide the formula that we use:

$$
\begin{gathered}
s_1(\tau, r) = \frac{1}{2}\left[\frac{\exp(\mathbf{q}_r \cdot \mathbf{q}_\tau)}{\sum_{r' \in P} \exp(\mathbf{q}_{r'} \cdot \mathbf{q}_\tau)} + \frac{\exp(\mathbf{q}_r \cdot \mathbf{q}_\tau)}{\sum_{\tau' \in \mathcal{T}} \exp(\mathbf{q}_r \cdot \mathbf{q}_{\tau'})}\right] \\
s_2(\tau, r) = \frac{\mathbf{q}_r \cdot \mathbf{q}_\tau}{\|\mathbf{q}_r\| \|\mathbf{q}_\tau\|} \\
s(\tau, r) = \frac{1}{2}(s_1(\tau, r) + s_2(\tau, r))
\end{gathered}
\tag{1}
$$

where $\mathbf{s}(\tau, r)$ represents the similarity score between a track τ and an object candidate r. Here, $\mathbf{q}_r$ denotes the detection embedding of the object candidate r, encapsulating its appearance features, while $\mathbf{q}_\tau$ represents the track embedding for track τ, capturing the features of the tracked object. The $s_1(\tau, r)$ employs an exponential function to compute the dot product of these embeddings, reflecting the degree of similarity between the object candidate and the track. This similarity score is normalized twice: firstly, across all object candidates r' in the set P for a given track τ, and secondly, across all tracks τ' in the set $\mathcal{T}$ for a given object candidate r. This dual normalization ensures a balanced and comprehensive assessment of similarity, facilitating accurate object association in dynamic video sequences. $s_2(\tau, r)$ computes the cosine similarity. The final $s(\tau, r)$ score is the average between $s_1(\tau, r)$ and $s_2(\tau, r)$.

Algorithm 1 Inference pipeline of MASA for associating objects across a video sequence.

```
Input: frame index $t$, object candidates $r \in P$, confidence
    $p_r$, detection embeddings $\mathbf{q}_r$, and track embeddings $\mathbf{q}_\tau$
    for all $\tau \in \mathcal{T}$.
DuplicateRemoval($P$)
for $r \in P, \tau \in \mathcal{T}$                      # compute matching scores
    $\mathbf{f}(r, \tau) = \texttt{similarity}(\mathbf{q}_r, \mathbf{q}_\tau)$
end for
for $r \in P$                                  # track management
    $c = \texttt{max}(\mathbf{f}(r))$                       # match confidence
    $\tau_{\texttt{match}} = \texttt{argmax}(\mathbf{f}(r))$                # matched track ID
    if $c > \beta$ and $p_i > \beta_{\texttt{obj}}$             # object match found
        updateTrack($\tau_{\texttt{match}}, r, \mathbf{q}_r, t$) # update track
    else if $p_r > \gamma$
        createTrack($r, \mathbf{q}_r, t$)           # create new track
    end if
end for
```

K. Limitations

One key limitation of our approach is handling temporal inconsistencies in detection or segmentation results across video frames. This issue, common in open-world object detection and segmentation models like SAM, is evident when an object detected in one frame is missed in the next, causing flickering effects in video visualization, as seen in our demonstrations. While our MASA adapter excels in learning associations, it cannot rectify foundational models' detection or segmentation errors. The challenge of generating consistent proposals across frames highlights an important area for future research to enhance the robustness and stability of object detection in dynamic video environments.

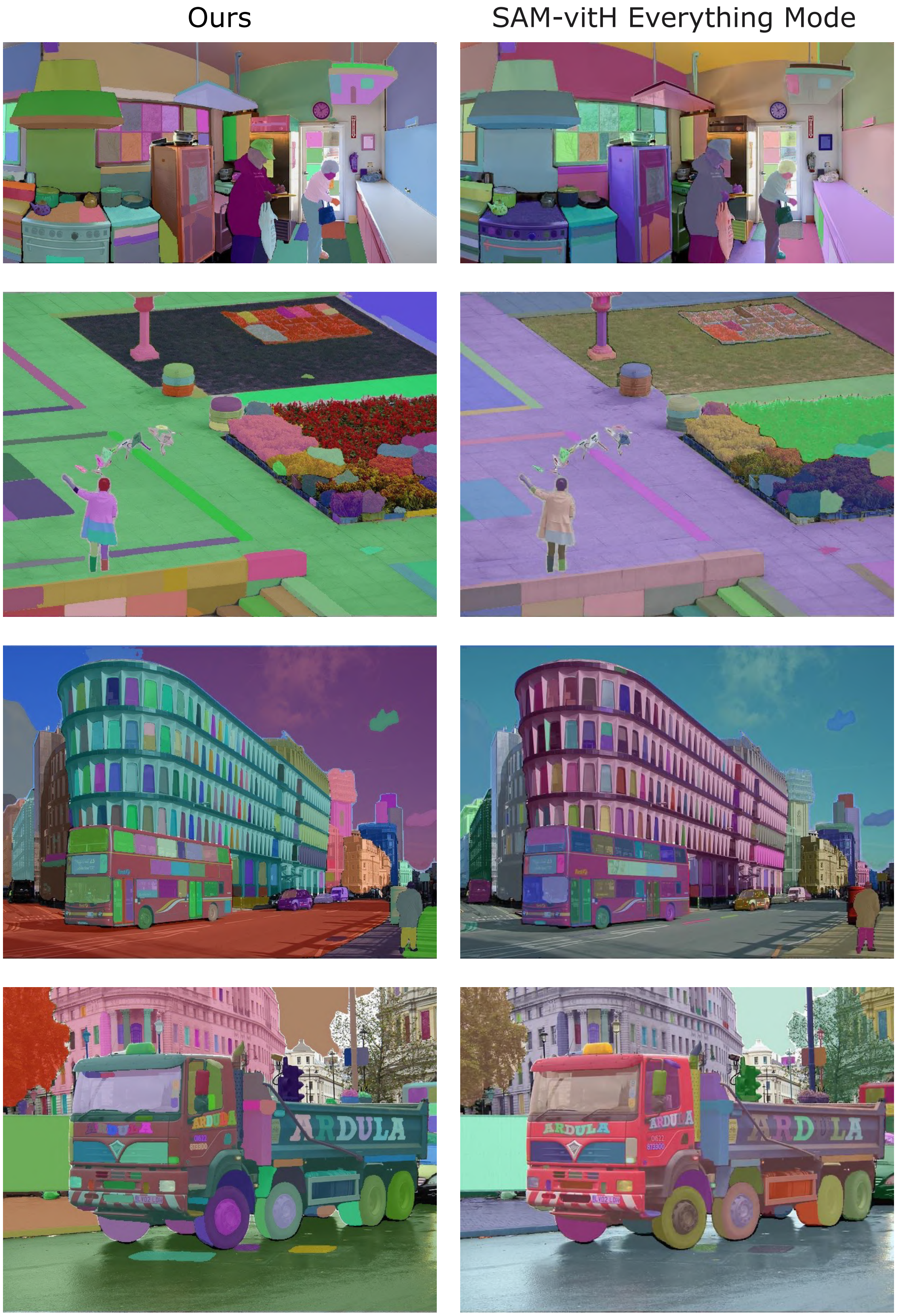


Figure 9. Qualitative comparison between our fast proposal generation and the original SAM everything mode on images from COCO validation set. The results show that our fast proposal generation can achieve similar segmentation quality to the everything mode of SAM, despite using much less time.

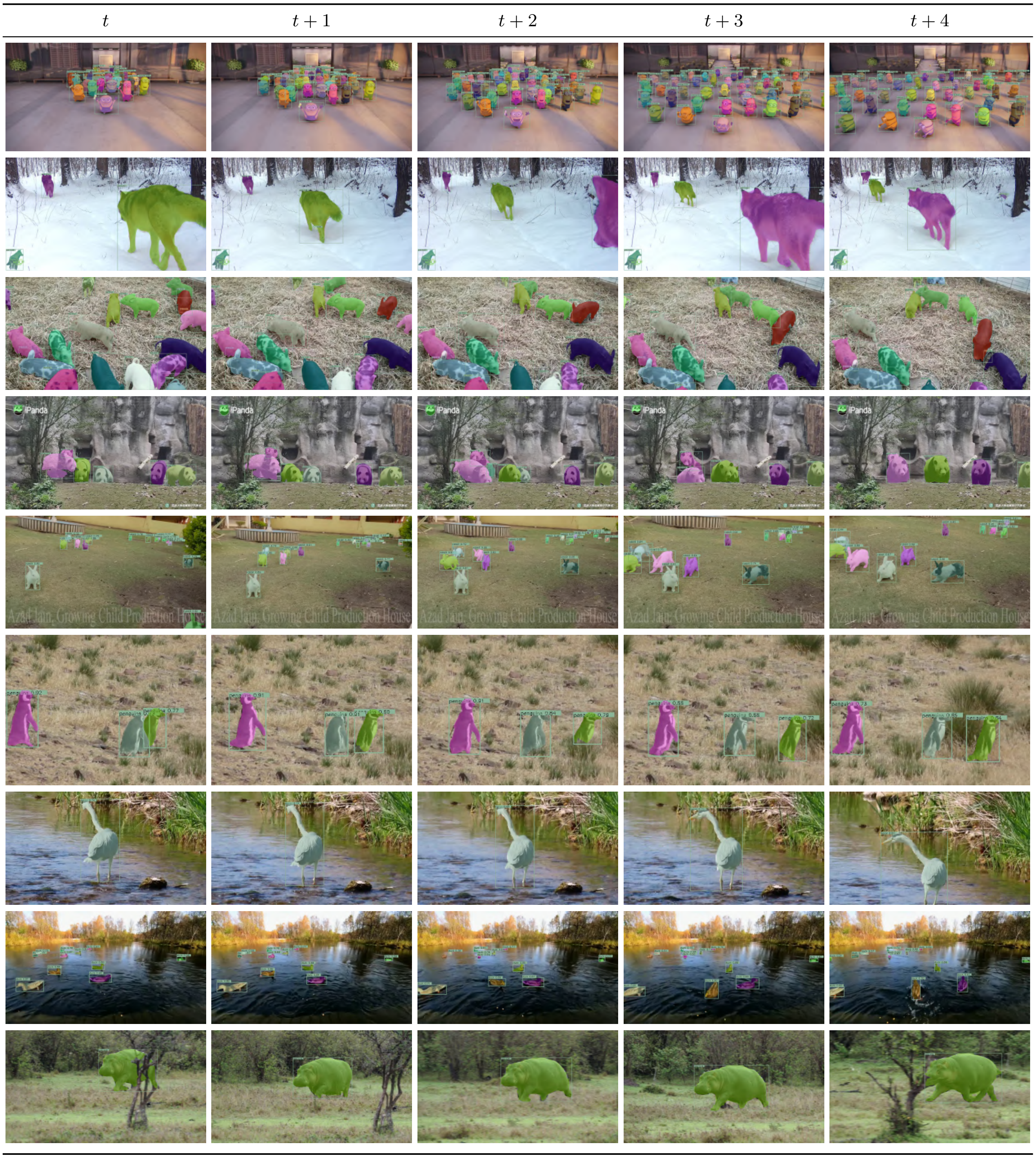


Figure 10. **Open-Vocabulary Tracking.** We condition our Grounding-DINO tracker on text prompts unseen during training and successfully track the corresponding objects in the videos. We use SAM to generate the mask from given the detected boxes. The mask color depicts the object's identity. We choose random internet videos to test our algorithm on diverse real-world scenarios. Best viewed digitally.

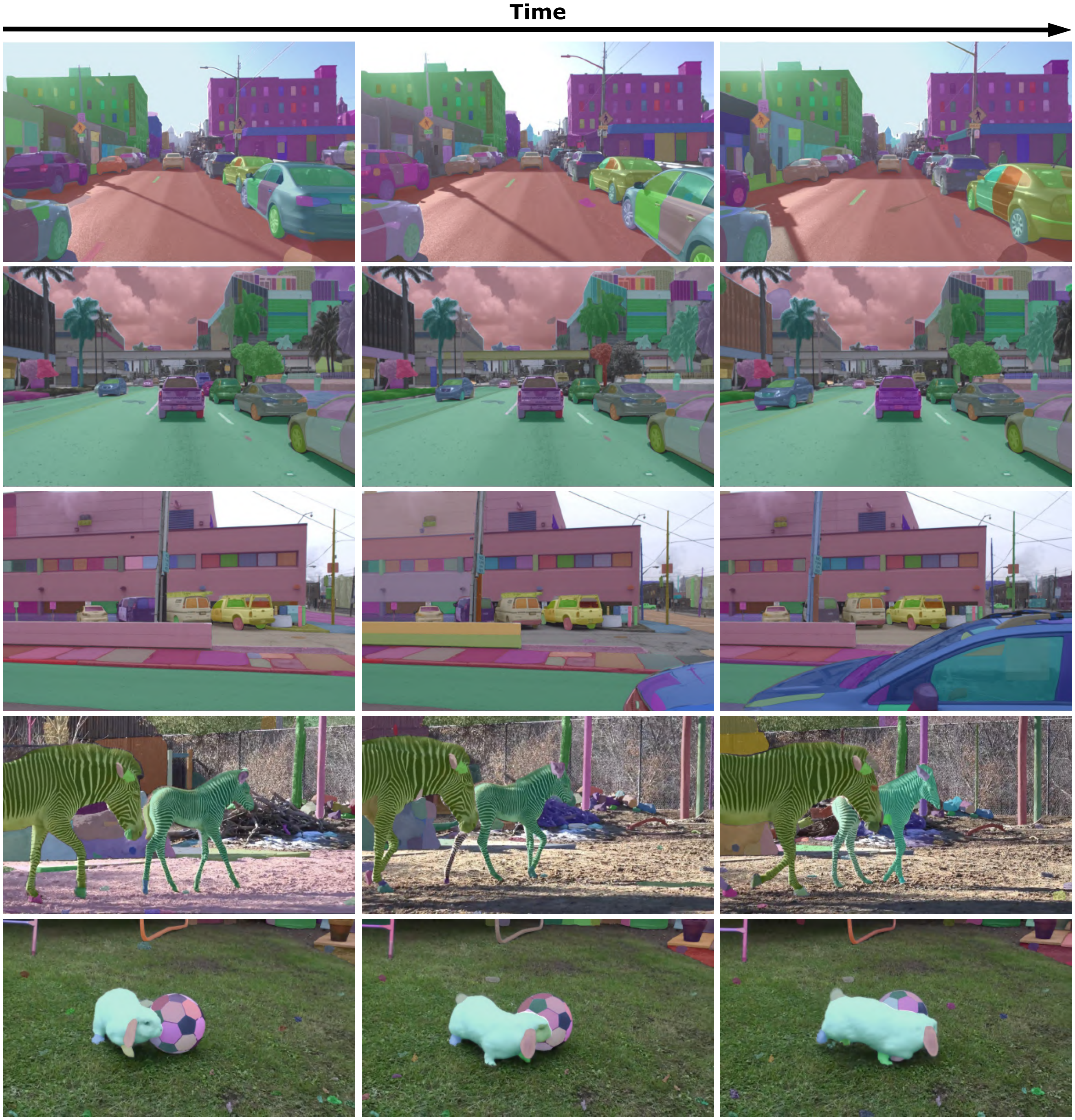


Figure 11. Qualitative results of unified proposal generation and association. The same colour indicates the same instance. We notice that although we can learn strong associations using MASA, it is still very difficult to generate consistent proposals across frames. For example, we can see the missing segmentation of the building on the left in the second row. This indicates further research efforts are needed on consistent proposal generation in videos.

References

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

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607. PMLR, 2020. 6

[4] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 4, 6

[5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 4

[6] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Alexander Schwing, and Joon-Young Lee. Tracking anything with decoupled video segmentation. In *ICCV*, 2023. 5, 6, 7, 8, 9

[7] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022. 5

[8] Yangming Cheng, Liulei Li, Yuanyou Xu, Xiaodi Li, Zongxin Yang, Wenguan Wang, and Yi Yang. Segment and track anything. *arXiv preprint arXiv:2305.06558*, 2023. 5

[9] Xiyang Dai, Yinpeng Chen, Bin Xiao, Dongdong Chen, Mengchen Liu, Lu Yuan, and Lei Zhang. Dynamic head: Unifying object detection heads with attentions. In *CVPR*, 2021. 7

[10] Achal Dave, Tarasha Khurana, Pavel Tokmakov, Cordelia Schmid, and Deva Ramanan. TAO: A large-scale benchmark for tracking any object. In *ECCV*, 2020. 2

[11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 8

[12] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[14] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation via object token association. *NeurIPS*, 2022. 2

[15] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. *NeurIPS*, 2021. 2

[16] Lei Ke, Martin Danelljan, Henghui Ding, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask-free video instance segmentation. In *CVPR*, 2023. 2

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 2, 6

[18] Siyuan Li, Martin Danelljan, Henghui Ding, Thomas E Huang, and Fisher Yu. Tracking every thing in the wild. In *ECCV*. Springer, 2022. 2

[19] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 8

[20] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 8

[21] 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

[22] Zhongdao Wang, Hengshuang Zhao, Ya-Li Li, Shengjin Wang, Philip Torr, and Luca Bertinetto. Do different tracking tasks require different appearance models? *NeurIPS*, 2021. 4

[23] J Wu, Y Jiang, S Bai, W Zhang, and X Bai. Seqformer: Sequential transformer for video instance segmentation. *ECCV*, 2021. 2

[24] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. *ECCV*, 2022. 2

[25] Jiarui Xu and Xiaolong Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *ICCV*, 2021. 2, 4, 6

[26] Bin Yan, Yi Jiang, Jiannan Wu, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Universal instance perception as object discovery and retrieval. In *CVPR*, 2023. 2

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

[28] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. In *NeurIPS*, 2022. 5, 7

[29] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 2

[30] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *ECCV*, 2022. 2

[31] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 8

[32] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9308–9316, 2019. 7