

Video Instance Segmentation via Multi-Scale Spatio-Temporal Split Attention Transformer

Omkar Thawakar¹, Sanath Narayan², Jiale Cao³, Hisham Cholakkal¹,
Rao Muhammad Anwer¹, Muhammad Haris Khan¹, Salman Khan¹,
Michael Felsberg⁴, Fahad Shahbaz Khan^{1,4}

¹MBZUAI, UAE ²IIAI, UAE

³Tianjin University, China ⁴Linköping University, Sweden

omkar.thawakar@mbzuai.ac.ae

Abstract. State-of-the-art transformer-based video instance segmentation (VIS) approaches typically utilize either single-scale spatio-temporal features or per-frame multi-scale features during the attention computations. We argue that such an attention computation ignores the multi-scale spatio-temporal feature relationships that are crucial to tackle target appearance deformations in videos. To address this issue, we propose a transformer-based VIS framework, named MS-STs VIS, that comprises a novel multi-scale spatio-temporal split (MS-STs) attention module in the encoder. The proposed MS-STs module effectively captures spatio-temporal feature relationships at multiple scales across frames in a video. We further introduce an attention block in the decoder to enhance the temporal consistency of the detected instances in different frames of a video. Moreover, an auxiliary discriminator is introduced during training to ensure better foreground-background separability within the multi-scale spatio-temporal feature space. We conduct extensive experiments on two benchmarks: Youtube-VIS (2019 and 2021). Our MS-STs VIS achieves state-of-the-art performance on both benchmarks. When using the ResNet50 backbone, our MS-STs achieves a mask AP of 50.1%, outperforming the best reported results in literature by 2.7% and by 4.8% at higher overlap threshold of AP₇₅, while being comparable in model size and speed on Youtube-VIS 2019 val. set. When using the Swin Transformer backbone, MS-STs VIS achieves mask AP of 61.0% on Youtube-VIS 2019 val. set. Source code is available at <https://github.com/OmkarThawakar/MSSTs-VIS>.

1 Introduction

Video instance segmentation (VIS) is a challenging computer vision problem with numerous real-world applications, including intelligent video analysis and autonomous driving. Given a video sequence, the task is to simultaneously segment and track all object instances from a set of semantic categories. The problem is particularly challenging since the target object needs to be accurately segmented and tracked despite appearance deformations due to several real-world issues such as, target size variation, aspect-ratio change and fast motion.

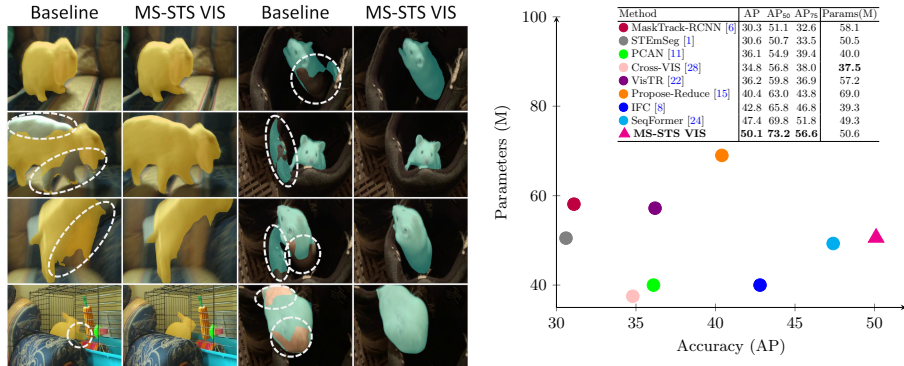


Fig. 1: **On the left:** Qualitative comparison of our MS-STs VIS with the baseline in the case of target appearance deformations on two example videos. For each method, four sample video frames are shown along the columns. The baseline struggles to accurately predict the *rabbit* class instance undergoing appearance deformations due to scale variation, aspect-ratio change and fast motion. As a result, the predicted mask quality is hampered (marked by white dotted region). Similarly, the mask quality is also deteriorated for the *mouse* class. Our MS-STs VIS addresses these issues by capturing multi-scale spatio-temporal feature relationships, leading to improved mask quality. Best viewed zoomed in. **On the right:** Accuracy (AP) *vs.* model size (params) comparison with existing methods using a single model and single-scale inference on YouTube-VIS 2019 val. set. We also report performance at AP₅₀ and AP₇₅. All methods here utilize the same ResNet50 backbone. Compared to the best reported results in literature [24], our MS-STs VIS achieves absolute gains of 2.7% and 4.8% in terms of overall mask AP and at higher overlap threshold of AP₇₅, respectively, while being comparable in model size and speed ([24]: 11 FPS *vs.* **Ours:** 10 FPS).

Recently, transformers [20] have shown promising results on several vision tasks, including VIS [22,24]. The recent transformer-based VIS approaches [22,24] are built on DETR [4] and Deformable DETR [29] frameworks, utilizing an encoder-decoder architecture along with instance sequence matching and segmentation mechanisms to generate final video mask predictions. These approaches typically employ either single-scale spatio-temporal features [22] or per-frame multi-scale features [24] during attention computations at the encoder and decoder. However, such an attention computation ignores the *multi-scale spatio-temporal feature relationships*, which are crucial towards handling target appearance deformations due to real-world challenges such as, scale variation, change in aspect-ratio and fast motion in videos.

In this work, we investigate the problem of designing an attention mechanism, within the transformer-based VIS framework, to effectively capture multi-scale spatio-temporal feature relationships in a video. With this aim, we introduce a multi-scale spatio-temporal attention mechanism, which learns to aggregate

the necessary attentions performed along the spatial and temporal axes without losing crucial information related to target appearance deformations in both the spatial and temporal axes. In addition to target appearance deformations, another major challenge in VIS is the accurate delineation of the target object in the presence of cluttered background. Surprisingly, existing transformer-based VIS approaches do not employ an explicit mechanism to enforce foreground-background (fg-bg) separability. Here, we introduce a loss formulation that improves fg-bg separability by emphasizing the fg regions in multi-scale spatio-temporal features while simultaneously suppressing the bg regions.

Contributions: We propose a transformer-based video instance segmentation framework, MS-STS VIS, with the following contributions.

- We propose a novel multi-scale spatio-temporal split (MS-STS) attention module in the transformer encoder for effectively capturing spatio-temporal feature relationships. Our MS-STS module first attends to features across frames at a given spatial scale via an intra-scale temporal attention block, and then progressively attends to neighboring spatial scales across frames via an inter-scale temporal attention block to obtain enriched feature representations. To further improve the video mask prediction, we introduce an attention block in the decoder that enhances the temporal consistency of the detected instances in different frames of a video.
- We introduce an auxiliary discriminator network during training to enhance the fg-bg separability within the multi-scale spatio-temporal feature space. Here, the discriminator network is trained to distinguish between ground-truth and predicted mask, while the encoder is learned to fool the discriminator by generating features that result in better mask predictions.
- Comprehensive experiments are performed on Youtube-VIS 2019 and 2021 datasets. Our proposed MS-STS VIS sets a new state-of-the-art on both datasets. When using the ResNet50 backbone, our MS-STS VIS outperforms all existing methods with an overall mask AP of 50.1% on Youtube-VIS 2019 val. set, while being on par in terms of model size and speed compared to the state-of-the-art method (see Fig. 1 (right)). Specifically, the proposed MS-STS VIS achieves a significant performance improvement over the baseline, in case of target appearance deformations due to scale variation, aspect-ratio change and fast motion (see Fig. 1 (left)).

2 Baseline Framework

We base our approach on the recently introduced SeqFormer [24]. SeqFormer is built on the Deformable DETR [29] framework, which comprises a CNN backbone followed by a transformer encoder-decoder with deformable attention. We choose SeqFormer as our base framework to demonstrate the impact of our proposed contributions on a *strong baseline*. Here, a video clip \mathbf{x} consisting T frames of spatial size $H^0 \times W^0$ with a set of object instances is input to the backbone. Latent feature maps for each frame are obtained from the backbone at L multiple scales 2^{-l-2} with $1 \leq l \leq L$ and passed through separate convolution filters

for each scale. The output feature dimension for each of these convolution filters is set to C . The resulting multi-scale feature maps of each frame are then input to a transformer encoder comprising multi-scale deformable attention blocks. For each frame, the transformer encoder outputs multi-scale feature maps with the same size as its input. These encoder output features maps from each frame along with n learnable instance query embeddings $\mathbf{I}^Q \in \mathbb{R}^C$ are then input to the transformer decoder comprising a series of self- and cross-attention blocks. The n instance queries are further decomposed into n box queries \mathbf{B}^Q per-frame and are used to query the box features from the encoder feature maps of the corresponding frame. The learned box queries across T frames are then aggregated temporally to obtain n instance features $\mathbf{I}^O \in \mathbb{R}^C$. These instance features output by the decoder are then used for video instance mask prediction. We refer to [24] for additional details.

Limitation: As discussed above, the aforementioned SeqFormer framework independently utilizes *per-frame* multi-scale features during attention computations. As a result, it ignores the spatio-temporal feature relationships during attention computation that is crucial for the VIS problem. Different from SeqFormer that utilizes *per-frame* spatial features at multiple scales, our approach performs *multi-scale spatio-temporal attention* computation. Such a multi-scale spatio-temporal attention is especially desired in cases when the target object undergoes appearance deformations due to real-world challenges such as, scale variation, aspect-ratio change and fast motion in videos (see Fig. 1 and 3). Furthermore, distinct from the baseline SeqFormer, our approach employs an explicit mechanism to ensure accurate delineation of foreground objects from the cluttered background by enhancing fg-bg separability.

3 Method

3.1 Overall Architecture

Fig. 2(a) shows the overall architecture of the proposed MS-STC VIS approach built on the baseline framework described above. Our MS-STC VIS comprises a backbone network, a transformer encoder-decoder and a sequence matching and segmentation block. The focus of our design is the introduction of a novel multi-scale spatio-temporal split (MS-STC) attention module (Fig. 2(b)) in the transformer encoder to effectively capture spatio-temporal feature relationships at multiple scales across frames in a video. The MS-STC attention module comprises intra- and inter-scale temporal attention blocks (Fig. 2(c) and (d)). The intra-scale block takes C -dimensional backbone features \mathbf{F} as input and enriches features across frames in a video within a given scale, whereas the inter-scale block then progressively attends to multiple spatial scales across frames to generate C -dimensional spatio-temporally enriched features $\hat{\mathbf{Z}}$. These features $\hat{\mathbf{Z}}$ are fused in each encoder layer with the base features output by the standard deformable attention. While backbone features \mathbf{F} are used as input to the first encoder layer, the subsequent layers utilize the outputs from the preceding layer

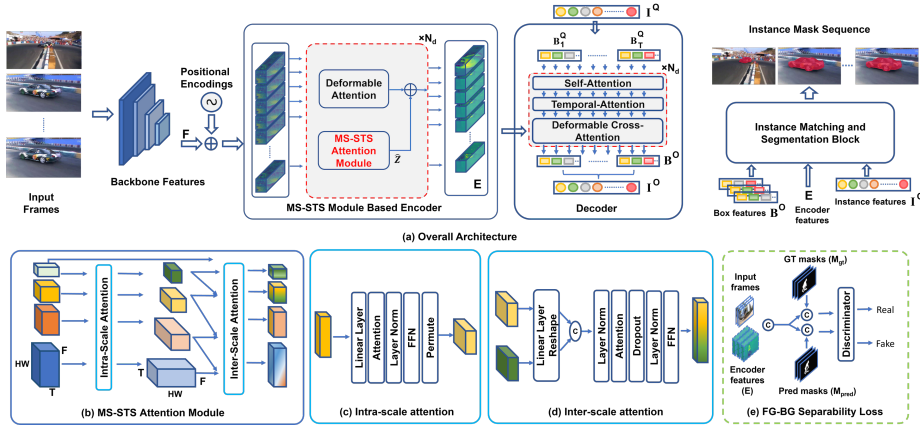


Fig. 2: (a) Our MS-STs VIS architecture comprises a backbone, a transformer encoder-decoder and a instance matching and segmentation block. Here, our key contributions are: (i) a novel MS-STs attention module in the encoder to capture spatio-temporal feature relationships at multiple scales across frames, (ii) a temporal attention block in the decoder for enhancing the temporal consistency of the box queries and (iii) an adversarial loss for enhancing foreground-background (fg-bg) separability. Specifically, the MS-STs attention module (b) attends to the input backbone features by employing intra- and inter-scale temporal attentions to generate enriched features which are then fused with standard baseline features. While the intra-scale temporal attention block (c) enriches features across frames within a given scale, its inter-scale counterpart (d) progressively attends to multiple spatial scales across frames to obtain spatio-temporally enriched features. These enriched features are further improved using an adversarial loss (e) to enhance fg-bg separability. The resulting encoder features along with the temporally consistent instance features from the decoder are used within the matching and segmentation block for the video instance mask prediction.

as input. As a result, the encoder outputs C -dimensional multi-scale spatio-temporally enriched features \mathbf{E} that are further input to the transformer decoder. To achieve temporal consistency among box queries from different frames, we introduce a temporal attention block within the transformer decoder. Next, the encoder features \mathbf{E} along with the instance features \mathbf{I}^O (aggregated temporally attended box queries) from the decoder are utilized within the instance matching and segmentation block to obtain the video instance mask prediction. To further improve the predicted video instance mask quality, we introduce an adversarial loss during training to enhance foreground-background (fg-bg) separability. The adversarial loss (Fig. 2(e)) strives to enhance the encoder features by discriminating between the predicted and ground-truth masks, utilizing the encoder features \mathbf{E} , the input frames \mathbf{x} and the binary object mask \mathbf{M} . Next, we describe our MS-STs attention module-based encoder.

3.2 MS-STTS Attention Module Based Encoder

Within the proposed MS-STTS VIS framework, we introduce a novel multi-scale spatio-temporal split (MS-STTS) attention module in the transformer encoder to effectively capture spatio-temporal feature relationships at multiple scales across frames. To this end, the MS-STTS module takes the backbone features as input and produces multi-scale spatio-temporally enriched features, which are then fused with the standard features within the base framework. The MS-STTS module (see Fig. 2(b)) comprises an intra-scale and an inter-scale attention block (see Fig. 2(c) and (d)) described next.

Intra-scale Temporal Attention Block: Given the backbone features as input, our intra-scale block independently attends to each scale (spatial resolution) temporally across frames (see Fig. 2(c)). Let $\mathbf{z}_{s,t}^l$ be the feature at spatial scale l , position s and frame t , where $s \in [0, S^l - 1]$, $t \in [0, T - 1]$, and $S^l = H^l \cdot W^l$. The intra-scale temporal attention block processes the features $\mathbf{Z}^l \in \mathbb{R}^{S^l \times T \times C}$ using intra-scale self-attention (SA_{intra}), layer normalization (LN), and MLP layers as:

$$\mathbf{Y}^l = \text{SA}_{\text{intra}}(\text{LN}(\mathbf{Z}^l)) + \mathbf{Z}^l, \quad \tilde{\mathbf{Z}}^l = \text{MLP}(\text{LN}(\mathbf{Y}^l)) + \mathbf{Y}^l, \quad (1)$$

$$\text{where } \text{SA}_{\text{intra}}(\text{LN}(\mathbf{z}_{s,t}^l)) = \sum_{\hat{t}=0}^{T-1} \text{Softmax}\left(\mathbf{q}_{s,t}^l \cdot \mathbf{k}_{s,\hat{t}}^l / \sqrt{C}\right) \mathbf{v}_{s,\hat{t}}^l, \quad (2)$$

where $\mathbf{q}_{s,t}^l, \mathbf{k}_{s,t}^l, \mathbf{v}_{s,t}^l \in \mathbb{R}^D$ are the query, key, and value vectors obtained from $\mathbf{z}_{s,t}^l$ (after LN) using the embedding matrices $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v \in \mathbb{R}^{C \times C}$. The intra-scale temporal attention operates on each spatial scale l across frames and produces temporally relevant intermediate features $\{\tilde{\mathbf{Z}}^l\}_{l=1}^L$.

Inter-scale Temporal Attention Block: The inter-scale temporal attention block takes the intermediate features output by the intra-scale block and aims to learn similarities between the spatio-temporal features across two neighbouring spatial scales (see Fig. 2(e)). Let $\tilde{\mathbf{Z}}^l \in \mathbb{R}^{S^l \times T \times C}$ and $\tilde{\mathbf{Z}}^{l+1}$ be the intra-scale attended features at spatial scales l and $l + 1$. To compute inter-scale temporal attention between the two neighbouring scale features, we first upsample the lower resolution features by $\times 2$ using bilinear interpolation, concatenate along the feature dimensions and project it into C dimensions using $\mathbf{W}_p \in \mathbb{R}^{2C \times C}$ as $\mathbf{H}^l = \mathbf{W}_p(\text{CONCAT}(\tilde{\mathbf{Z}}^l, \text{UPSAMPLE}(\tilde{\mathbf{Z}}^{l+1})))$. We reshape $\mathbf{H}^l \in \mathbb{R}^{S^l \times T \times C}$ to $\mathbb{R}^{(S^l \cdot T) \times C}$ and then compute the joint spatio-temporal attention to obtain enriched features $\hat{\mathbf{Z}}^l$ given by

$$\mathbf{Y}^l = \text{SA}_{\text{inter}}(\text{LN}(\mathbf{H}^l)) + \mathbf{H}^l, \quad \hat{\mathbf{Z}}^l = \text{MLP}(\text{LN}(\mathbf{Y}^l)) + \mathbf{Y}^l, \quad (3)$$

where the inter-scale self-attention (SA_{inter}) computation is given by

$$\text{SA}_{\text{inter}} = \sum_{\hat{s}=0}^{S-1} \sum_{\hat{t}=0}^{T-1} \text{Softmax}\left(\mathbf{q}_{s,t}^l \cdot \mathbf{k}_{\hat{s},\hat{t}}^l / \sqrt{C}\right) \mathbf{v}_{\hat{s},\hat{t}}^l. \quad (4)$$

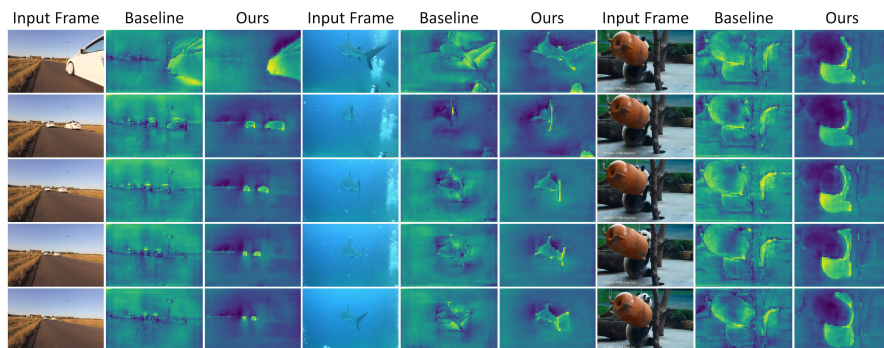


Fig. 3: Example attention map visualizations obtained at the output of the baseline encoder and our MS-STS attention module-based encoder. Here, we show the comparison on example videos from Youtube-VIS 2019 val. set. The baseline struggles to accurately focus on the *car* instance undergoing significant scale variation, where the size becomes extremely small in the later frames. Similarly, it fails to accurately focus on the *shark* instance undergoing aspect-ratio change and the *person* instance partially visible only in the first two frames of the middle video. In the last video, the baseline inaccurately highlights the irrelevant object (in orange) occluding the target *panda* instance. Our MS-STS attention module-based encoder, which strives to capture spatio-temporal feature relationships at multiple scales, successfully focuses on these challenging targets despite scale variations (*car*), aspect-ratio change (*shark*), partial visibility (*person*) and appearance deformations due to occlusion (*panda*). Best viewed zoomed in.

To summarize, our MS-STS attention module utilizes an intra-scale temporal attention block to first attend to features across frames at a given spatial scale. It then employs an inter-scale temporal attention block to progressively attend to neighboring spatial scales across frames for obtaining enriched feature representations $\hat{\mathbf{Z}}^l$. The resulting features $\hat{\mathbf{Z}}^l$ are fused with the standard baseline features through a convolution operation. Finally, the encoder outputs enriched multi-scale spatio-temporal features \mathbf{E} after processing the input features through N_d attention layers. Fig. 3 shows example attention maps obtained at the output of our MS-STS attention module-based encoder. The attention maps are computed as the average activation strength across the C features at a particular spatial position for the largest scale considered, *i.e.*, $H/s \times W/s$. The attention maps are shown for example videos from Youtube-VIS 2019 val. set depicting target appearance deformations. We also compare our encoder output to that of the baseline encoder. Compared to the baseline, our MS-STS module-based encoder better focuses on the *car* instance (left video) undergoing scale variations, the *shark* instance (middle video) undergoing aspect-ratio change, the *person* instance (middle video) partially visible only in the first two frames and the *panda* instance (right video) exhibiting target deformations due to the irrelevant object (in orange) occluding the target *panda* instance.

3.3 Enhancing Temporal Consistency in Decoder

As discussed earlier, the transformer decoder in the base framework comprises a series (layers) of alternating self- and cross-attention blocks, operating on the box queries \mathbf{B}^Q of individual frames. Although the video-level instance features \mathbf{I}^O are obtained through an aggregation of the corresponding frame-level box features, the temporal consistency of the detected instances is hampered likely due to the per-frame attention computation of the box queries. To alleviate this issue, we introduce a temporal attention block in between the self- and cross-attention blocks of a decoder layer. Our temporal attention block attends to the sequence of box queries $\{\mathbf{B}_i^Q\}_{i=1}^T$ from T frames and produces temporally consistent queries at its output. Such a temporal attention between the per-frame box queries of an instance enables information flow between the probable box locations across frames, thereby improving the temporal consistency of the video mask predictions. The resulting temporally-attended box queries are then utilized to query the multi-scale spatio-temporally enriched features \mathbf{E} , output by the encoder, for computing the box features \mathbf{B}^O in a per-frame manner. Afterwards, these box features are aggregated to obtain video-level instance features \mathbf{I}^O . Finally, the resulting instance features along with the box features and multi-scale spatio-temporally enriched features \mathbf{E} are input to an instance matching and segmentation block for generating the video mask predictions.

3.4 Enhancing Foreground-Background Separability

Both our MS-STTS attention in encoder and temporal attention in decoder promote multi-scale spatio-temporal feature enrichment and temporal consistency, respectively, across frames. This helps to address the critical issue of appearance deformations in the target object due to challenges such as, scale variation, aspect-ratio change and fast motion. In addition to addressing target appearance deformations across frames, another common challenge in the VIS problem is the accurate delineation of foreground objects from the cluttered background. To this end, we introduce an adversarial loss during training of our MS-STTS VIS framework for enhancing foreground-background (fg-bg) separability. To the best of our knowledge, we are the first to explore an adversarial loss within a transformer-based VIS framework for enhancing fg-bg separability.

In our adversarial formulation, the objective is to discriminate between the ground-truth mask and the predicted mask output by our MS-STTS VIS. With this objective, we introduce an auxiliary discriminator network during training. The discriminator takes the input frames along with the corresponding encoder features and binary masks as its input. Here, the binary mask \mathbf{M} is obtained either from the ground-truth or predictions, such that all object instances (category-agnostic) within a frame are marked as foreground. While the discriminator D attempts to distinguish between ground-truth and predicted binary masks (\mathbf{M}_{gt} and \mathbf{M}_{pred} , respectively), the encoder learns to output enhanced features \mathbf{E} such that the predicted masks \mathbf{M}_{pred} are close to ground-truth \mathbf{M}_{gt} . Let $\mathbf{F}_{gt} = \text{CONCAT}(\mathbf{x}, \mathbf{E}, \mathbf{M}_{gt})$ and $\mathbf{F}_{pr} = \text{CONCAT}(\mathbf{x}, \mathbf{E}, \mathbf{M}_{pred})$ denote

the real and fake input, respectively, to the discriminator D . Similar to [9], the adversarial loss is then given by

$$\min_{Enc} \max_D \mathbb{E}[\log D(\mathbf{F}_{gt})] + \mathbb{E}[\log(1 - D(\mathbf{F}_{pr}))] + \lambda_1 \mathbb{E}[|D(\mathbf{F}_{gt}) - D(\mathbf{F}_{pr})|_1]. \quad (5)$$

Since the mask prediction depends on the quality of the encoder features that are decoded by the queries, we treat our encoder Enc as a generator in our adversarial formulation above. As a result, the encoder learns to better delineate foreground and background regions leading to improved video instance mask prediction. Note that the discriminator network is utilized only during training.

4 Experiments

Datasets: The *YouTube-VIS 2019* [26] dataset contains 2883 high-quality videos with 131K annotated object instances belonging to 40 different categories. The *YouTube-VIS 2021* [25] dataset contains 3,859 high-quality videos with 232K annotated object instances belonging to 40 different improved categories. YouTube-VIS 2021 [25] has a 40-category label set by merging *eagle* and *owl* into *bird*, *ape* into *monkey*, deleting *hands*, and adding *flying disc*, *squirrel* and *whale*, maintaining the same number of categories as YouTube-VIS 2019 [26] set.

Evaluation Metrics: We follow the standard protocol, where the evaluation metrics, Average Precision (AP) and Average Recall (AR), are adapted from image instance segmentation with the video Intersection over Union (IoU) of the mask sequences as the threshold.

Implementation Details: We choose ResNet-50 [7] as the default backbone, unless otherwise specified, for our MS-STs VIS framework. Outputs from conv_3 , conv_4 and conv_5 of the backbone are utilized to obtain multi-scale feature inputs to our transformer encoder-decoder, as in [29]. Both encoder and decoder layers are set to $N_d=6$. The feature dimension C is set to 256, while the number of instance queries is set to 300 and length of video clip $T=5$, as in [24]. We use the AdamW optimizer with a base learning rate (LR) of 2×10^{-4} , $(\beta_1, \beta_2)=(0.9, 0.999)$ and a weight decay of 10^{-4} . LR of linear projections of deformable attention modules and the backbone are scaled by 0.1. The model is first pretrained on COCO [16] for 24 epochs with a batch size of 2. Similar to [24], the pretrained weights are then used to train the model on Youtube-VIS and COCO dataset for 12 epochs with $T=5$ and batch size set to 2. The LR is scaled by a factor of 0.1 at 4th and 10th epochs. The framework is trained on 8 Nvidia V100 GPUs using PyTorch-1.9 [18].

4.1 State-of-the-art Comparison

Tab. 1 presents the state-of-the-art comparison on the YouTube-VIS 2019 val. set. When using the ResNet-50 backbone, the recent one-stage PCAN [11] and CrossVIS [28] approaches achieve an overall mask accuracy (AP) of 36.1% and

Table 1: State-of-the-art comparison on **YouTube-VIS 2019 val** set. Our MR-STS VIS consistently outperforms the state-of-the-art results reported in literature. When using the ResNet-50 backbone, MS-STS VIS achieves overall mask AP score of 50.1% with an absolute gain of 2.7% over the best existing SeqFormer, while being comparable in terms of model size and speed (SeqFormer: 11 FPS *vs.* Ours: 10 FPS). Similarly, when using the ResNet-101 backbone, our MS-STS VIS achieves overall mask AP of 51.1%. Further, MS-STS VIS achieves the best accuracy reported on this dataset with a mask AP of 61.0% and outperforms SeqFormer with an absolute gain of 1.7%, using the same Swin-L backbone.

Method	Venue	Backbone	Type	AP	AP ₅₀	AP ₇₅	AR ₁	AR ₁₀
IoUTracker+ [26]	ICCV 2019	ResNet-50	-	23.6	39.2	25.5	26.2	30.9
OSMN [27]	CVPR 2018	ResNet-50	Two-Stage	27.5	45.1	29.1	28.6	33.1
DeepSORT [23]	ICIP 2017	ResNet-50	Two-stage	26.1	42.9	26.1	27.8	31.3
FEELVOS [21]	CVPR 2019	ResNet-50	Two-stage	26.9	42.0	29.7	29.9	33.4
SeqTracker [26]	ICCV 2019	ResNet-50	-	27.5	45.7	28.7	29.7	32.5
MaskTrack R-CNN [26]	ICCV 2019	ResNet-50	Two-stage	30.3	51.1	32.6	31.0	35.5
MaskProp [2]	CVPR 2020	ResNet-50	-	40.0	-	42.9	-	-
SipMask-VIS [3]	ECCV 2020	ResNet-50	One-stage	32.5	53.0	33.3	33.5	38.9
SipMask-VIS [3]	ECCV 2020	ResNet-50	One-stage	33.7	54.1	35.8	35.4	40.1
STEM-Seg [1]	ECCV 2020	ResNet-50	-	30.6	50.7	33.5	31.6	37.1
Johnander <i>et al.</i> [10]	GCPR 2021	ResNet-50	-	35.3	-	-	-	-
CompFeat [5]	AAAI 2021	ResNet-50	-	35.3	56.0	38.6	33.1	40.3
CrossVIS[28]	ICCV 2021	ResNet-50	One-stage	36.3	56.8	38.9	35.6	40.7
PCAN [11]	NeurIPS 2021	ResNet-50	One-stage	36.1	54.9	39.4	36.3	41.6
VisTR [22]	CVPR 2021	ResNet-50	Transformer	35.6	56.8	37.0	35.2	40.2
SeqFormer [24]	Arxiv 2021	ResNet-50	Transformer	47.4	69.8	51.8	45.5	54.8
MS-STS VIS (Ours)		ResNet-50	Transformer	50.1	73.2	56.6	46.1	57.7
MaskTrack R-CNN [26]	ICCV 2019	ResNet-101	Two-stage	31.9	53.7	32.3	32.5	37.7
MaskProp [2]	CVPR 2020	ResNet-101	-	42.5	-	45.6	-	-
STEM-Seg [1]	ECCV, 2020	ResNet-101	-	34.6	55.8	37.9	34.4	41.6
CrossVIS [28]	ICCV 2021	ResNet-101	One-stage	36.6	57.3	39.7	36.0	42.0
PCAN [11]	NeurIPS 2021	ResNet-101	One-stage	37.6	57.2	41.3	37.2	43.9
VisTR [22]	CVPR 2021	ResNet-101	Transformer	38.6	61.3	42.3	37.6	44.2
SeqFormer [24]	Arxiv 2021	ResNet-101	Transformer	49.0	71.1	55.7	46.8	56.9
MS-STS VIS (Ours)		ResNet-101	Transformer	51.1	73.2	59.0	48.3	58.7
SeqFormer [24]	Arxiv 2021	Swin-L	Transformer	59.3	82.1	66.6	51.7	64.4
MS-STS VIS (Ours)		Swin-L	Transformer	61.0	85.2	68.6	54.7	66.4

36.3%, respectively. With the same ResNet-50 backbone, the first transformer-based VIS approach, VisTR [22], built on DETR framework achieves an overall mask AP of 35.6%. Among existing methods, the recently introduced SeqFormer [24] based on Deformable DETR framework achieves the best overall accuracy with a mask AP of 47.4%. Our proposed MS-STS VIS approach outperforms SeqFormer [24] by achieving an overall mask AP of 50.1%, using the same ResNet-50 backbone. Specifically, our MS-STS VIS provides an absolute gain of 4.8% at a higher overlap threshold of AP₇₅ over SeqFormer. Similarly, our MS-STS VIS consistently outperforms SeqFormer with an overall mask AP of 51.1%, when using the ResNet-101 backbone. Finally, when using the recent Swin Transformer backbone, the proposed MS-STS VIS achieves the best accuracy reported in literature with an overall mask AP of 61.0%.

Table 2: State-of-the-art comparison on **YouTube-VIS 2021 val** set. All results are reported using the same ResNet-50 backbone. Our MS-STs VIS achieves state-of-the-art results with an overall mask AP of 42.2% and an absolute gain of 2.8% over the best existing SeqFormer at a higher overlap threshold of AP₇₅.

Method	AP	AP ₅₀	AP ₇₅	AR ₁	AR ₁₀
MaskTrack R-CNN[25]	28.6	48.9	29.6	26.5	33.8
SipMask-VIS[3]	31.7	52.5	34.0	30.8	37.8
VisTR[22]	31.8	51.7	34.5	29.7	36.9
CrossVIS[28]	34.2	54.4	37.9	30.4	38.2
IFC[8]	36.6	57.9	39.9	–	–
SeqFormer[24]	40.5	62.4	43.7	36.1	48.1
MS-STs VIS (Ours)	42.2	63.7	46.5	41.7	51.1

Table 3: **On the left:** Impact of our contributions when progressively integrating them into the baseline on the Youtube-VIS 2019 val. set. We observe a consistent performance improvement due to the integration of our proposed contributions. Our final MS-STs VIS (row 4) achieves an absolute gain of 3.7% over the baseline. **On the right:** Attribute-based performance comparison between the baseline and our MS-STs VIS on the custom set comprising 706 videos in total. Here, we present the comparison on fast motion, target size change and aspect-ratio change attributes. Our MS-STs VIS achieves consistent improvement in performance over the baseline on all attributes. See Sec. 4.2 for details.

Model	AP	Attribute Type	#Videos	Baseline	MS-STs VIS
Baseline	46.4	Custom Set	706	49.5	53.7
Baseline + MS-STs	48.4	Fast Motion	605	48.9	54.2
Baseline + MS-STs + T-Dec	49.1	Target Size Change	706	49.5	53.7
Baseline + MS-STs + T-Dec + FG-BG Loss	50.1	Aspect-Ratio Change	687	49.2	53.9

Tab. 2 reports the state-of-the-art comparison on the YouTube-VIS 2021 val. set. Among existing methods, CrossVIS [28] and IFC [8] achieve overall mask AP of 34.2% and 36.6%, respectively. SeqFormer [24] obtains a mask AP of 40.5%. Our MS-STs VIS sets a new state-of-the-art with an overall mask AP of 42.2%. Specifically, MS-STs VIS provides an absolute gain of 2.8% over SeqFormer at higher overlap threshold of AP₇₅, when using the same ResNet-50 backbone.

4.2 Ablation Study

Here, we first evaluate the merits of our three proposed contributions: MS-STs attention module-based encoder (Sec. 3.2), temporal attention in the decoder (Sec. 3.3) and the adversarial loss for enhancing fg-bag separability (Sec. 3.4). Tab. 3 (left) shows the baseline comparison on the YouTube-VIS 2019 val. set. All results reported in Tab. 3 (left) are obtained using the same ResNet50 backbone. As discussed earlier, our MS-STs VIS employs SeqFormer as its base framework. We train the baseline SeqFormer (denoted here as Baseline) using the official implementation and achieve an overall mask AP score of 46.4%. The introduction of our MS-STs attention module-based encoder within the baseline (referred as

Baseline + MS-STs) significantly improves the overall performance to 48.4% with an absolute gain of 2.0%. The overall performance is further improved to 49.1% with the integration of the temporal attention in the decoder (denoted as Baseline + MS-STs + T-Dec). Finally, the introduction of the adversarial loss during the training for enhancing the fg-bg separability provides an absolute gain of 1.0% (denoted as Baseline + MS-STs + T-Dec + FG-BG Loss). Our final MS-STs VIS achieves an absolute gain of 3.7% over the baseline.

We further analyze the performance of our method (MS-STs VIS) under three specific challenging scenarios: fast motion, target size change (scale variation) and aspect-ratio change. To this end, we classified the videos into three categories: (i) fast motion of the object, (ii) object size changes, and (iii) aspect-ratio changes. In particular, we follow the well-established VOT-2015 benchmark [12] to label a particular video from above categories as follows: (i) **fast motion**: if object center in current frame moves by at least 30% of its size in previous frame. (ii) **change in object size**: if the ratio of the maximum size to the minimum size of an object in the video is greater than 1.5. (iii) **change in aspect-ratio**: if the ratio of the maximum to the minimum aspect (width/height) of the bounding box enclosing an object in the video is greater than 1.5.

Based on the aforementioned criteria, we first select the newly added videos in Youtube-VIS 2021 training set while retaining the same set of classes as in Youtube-VIS 2019 dataset. We refer these selected videos as custom set and classify them into the aforementioned attributes. To evaluate the performance of our MS-STs VIS and the baseline, we use the models trained on Youtube-VIS 2019 training set with the same ResNet-50 backbone. Note that we ensure that there is no overlap between the videos in Youtube-VIS 2019 training set and our custom set (only comprising the newly added videos from the Youtube-VIS 2021 training set). Tab. 3 (right) shows the comparison between the baseline SeqFormer and our MS-STs VIS. On the entire custom set, our MS-STs VIS obtains significantly improved performance over the baseline. Specifically, MS-STs VIS achieves absolute gains of 5.3%, 4.2% and 4.7% over the baseline on fast motion, target size change and aspect-ratio change attributes, respectively.

4.3 Qualitative Analysis

Fig. 4 and 5 show qualitative results obtained by our MS-STs VIS framework on example videos from the Youtube-VIS 2019 val. and 2021 val. sets, respectively. We observe our MS-STs VIS framework to obtain promising video mask prediction in various challenging scenarios involving target appearance deformations due to fast motion, aspect-ratio change and scale variation. *E.g.*, in Fig. 4, video masks are predicted accurately for *hand* in row 1 (scale change), *eagle* in row 4 (fast motion, aspect-ratio change, scale variation), *panda* in row 2 (aspect-ratio change), *etc.*. Similarly, we observe promising video masks predictions for *leopard* in row 1 (fast motion), *person* in row 3 (scale variation), *dog* in row 2 (aspect-ratio change, scale variation) in Fig. 5. These results show the efficacy of our MS-STs VIS framework under different challenges for the task of video instance segmentation. Additional results are presented in the supplementary.

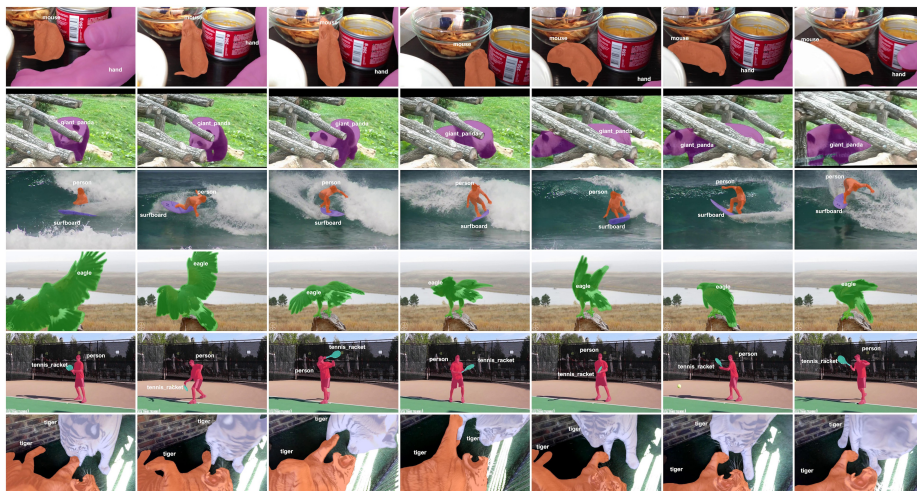


Fig. 4: Qualitative results obtained by our MS-STS VIS framework on six example videos in the Youtube-VIS 2019 val set. Our MS-STS VIS achieves promising video mask prediction in various challenging scenarios including, fast motion (*eagle* in row 4), scale change (*hand* in row 1), aspect-ratio change (*panda* in row 2, *person* in row 3), multiple instances of same class (*tiger* in row 6).

5 Related Work

Two-stage and Single-stage VIS: Several VIS methods [2,26,14] adapt the two-stage pipeline, such as Mask R-CNN [6] by introducing an additional tracking branch for target association. The work of [26] introduces MaskTrack R-CNN that jointly performs detection, segmentation and tracking. Bertasius *et al.* [2] utilize a branch in Mask R-CNN to propagate frame-level instance masks. The work of [14] introduces a modified variational autoencoder (VAE) on top of Mask R-CNN for VIS. On the other hand, several works [3,1,11,13,17] adapt the one-stage pipeline, such as FCOS [19], where a linear combination of mask bases are directly predicted as final segmentation. SipMask [3] introduces a spatial information preservation module for real-time VIS. The work of [1] introduces an approach where a short 3D convolutional spatio-temporal volume is adopted to learn pixel-level embedding. The work of [11] proposes to refine a space-time memory into a set of instance and frame-level prototypes.

Transformers-based VIS: Wang *et al.* [22] introduce a transformer-based encoder-decoder architecture, named VisTR, that formulates VIS as a direct end-to-end parallel sequence prediction task. In the encoder, VisTR utilizes a single-scale attention that computes similarities between all pairs of features from multiple spatial locations of a low-resolution feature map, across frames. The work of [8] introduces inter-frame communication transformers, where memory tokens are used to communicate between frames. The recent SeqFormer [24], built on Deformable DETR framework [29], utilizes per-frame multi-scale fea-



Fig. 5: Qualitative results on six example videos in the Youtube-VIS 2021 val set. Our MS-STs VIS achieves favorable video mask prediction in various scenarios involving target appearance deformations: fast motion (*leopard* in row 1), scale variation (*person* and *seal* in row 4), aspect-ratio change (*dog* in row 2, *tiger* in row 6). Also see the supplementary for additional results.

tures during attention computations. While demonstrating promising results, SeqFormer struggles in case of target deformations likely due to not explicitly capturing the spatio-temporal feature relationships during attention computation. To address these issues, we proposed a framework comprising an encoder that captures multi-scale spatio-temporal feature relationships. We also introduced an attention block in decoder to enhance temporal consistency of detected instance in different frames and an adversarial loss during training that ensures better fg-bg separability within multi-scale spatio-temporal feature space.

6 Conclusions

We proposed a transformer-based VIS framework, MS-STs VIS, comprising a novel multi-scale spatio-temporal split attention (MS-STs) module to effectively capture spatio-temporal feature relationships at multiple scales across frames in a video. We further introduced an auxiliary discriminator during training that strives to enhance fg-bg separability within the multi-scale spatio-temporal feature space. Our MS-STs VIS specifically tackles target appearance deformations due to real-world challenges. Extensive experiments reveal the benefits of the proposed contributions, achieving state-of-the-art performance on two benchmarks. **Acknowledgements:** The work was partially supported by VR grants 2016-05543 and 2018-04673, WASP, ELLIIT, and SNIC funded via grant 2018-05973.

References

1. Athar, A., Mahadevan, S., Osep, A., Leal-Taixé, L., Leibe, B.: Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In: ECCV (2020) [2](#), [10](#), [13](#)
2. Bertasius, G., Torresani, L.: Classifying, segmenting, and tracking object instances in video with mask propagation. In: CVPR (2020) [10](#), [13](#)
3. Cao, J., Anwer, R.M., Cholakkal, H., Khan, F.S., Pang, Y., Shao, L.: Sipmask: Spatial information preservation for fast image and video instance segmentation. In: ECCV (2020) [10](#), [11](#), [13](#)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020) [2](#)
5. Fu, Y., Yang, L., Liu, D., Huang, T.S., Shi, H.: Compfeat: Comprehensive feature aggregation for video instance segmentation. AAAI (2021) [10](#)
6. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017) [2](#), [13](#)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) [9](#)
8. Hwang, S., Heo, M., Oh, S.W., Kim, S.J.: Video instance segmentation using inter-frame communication transformers. In: NeurIPS (2021) [2](#), [11](#), [13](#)
9. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017) [9](#)
10. Johnander, J., Brissman, E., Danelljan, M., Felsberg, M.: Learning video instance segmentation with recurrent graph neural networks. In: GCPI (2021) [10](#)
11. Ke, L., Li, X., Danelljan, M., Tai, Y.W., Tang, C.K., Yu, F.: Prototypical cross-attention networks for multiple object tracking and segmentation. In: NeurIPS (2021) [2](#), [9](#), [10](#), [13](#)
12. Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Cehovin, L., Fernandez, G., Vojir, T., Hager, G., Nebel, G., Pflugfelder, R.: The visual object tracking vot2015 challenge results. In: ICCV workshops (2015) [12](#)
13. Li, M., Li, S., Li, L., Zhang, L.: Spatial feature calibration and temporal fusion for effective one-stage video instance segmentation. In: CVPR (2021) [13](#)
14. Lin, C., Hung, Y., Feris, R., He, L.: Video instance segmentation tracking with a modified vae architecture. In: CVPR (2020) [13](#)
15. Lin, H., Wu, R., Liu, S., Lu, J., Jia, J.: Video instance segmentation with a propose-reduce paradigm. In: arXiv preprint arXiv:2103.13746 (2021) [2](#)
16. Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, L.: Microsoft coco: Common objects in context. In: ECCV (2014) [9](#)
17. Liu, D., Cui, Y., Tan, W., Chen, Y.: Sg-net: Spatial granularity network for one-stage video instance segmentation. In: CVPR (2021) [13](#)
18. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: NeurIPS (2019), <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf> [9](#)
19. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: Fully convolutional one-stage object detection. In: ICCV (2019) [13](#)
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., Polosukhin, I.: Pattention is all you need. In: NeurIPS (2017) [2](#)

21. Voigtlaender, P., Chai, Y., Schroff, F., Adam, H., Leibe, B., Chen, L.: Feelvos: Fast end-to-end embedding learning for video object segmentation. In: CVPR (2019) [10](#)
22. Wang, Y., Xu, Z., Wang, X., Shen, C., Cheng, B., Shen, H., Xia, H.: End-to-end video instance segmentation with transformers. In: CVPR (2021) [2](#), [10](#), [11](#), [13](#)
23. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: ICIP (2017) [10](#)
24. Wu, J., Jiang, Y., Zhang, W., Bai, X., Bai, S.: Seqformer: a frustratingly simple model for video instance segmentation. In: arXiv preprint arXiv:2112.08275 (2021) [2](#), [3](#), [4](#), [9](#), [10](#), [11](#), [13](#)
25. Xu, N., Yang, L., Yang, J., Yue, D., Fan, Y., Liang, Y., Huang, T.S.: Youtube-vis dataset 2021 version. <https://youtube-vos.org/dataset/vis> (2021) [9](#), [11](#)
26. Yang, L., Fan, Y., Xu, N.: Video instance segmentation. In: ICCV (2019) [9](#), [10](#), [13](#)
27. Yang, L., Wang, Y., Xiong, X., Yang, J., Katsaggelos, A.K.: Efficient video object segmentation via network modulation. In: CVPR (2018) [10](#)
28. Yang, S., Fang, Y., Wang, X., Li, Y., Fang, C., Shan, Y., Feng, B., Liu, W.: Crossover learning for fast online video instance segmentation. In: ICCV (2021) [2](#), [9](#), [10](#), [11](#)
29. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: ICLR (2021) [2](#), [3](#), [9](#), [13](#)