Self-Supervised Video Object Segmentation via Cutout Prediction and Tagging

Jyoti Kini
jyoti.kini@knights.ucf.edu
University of Central Florida
USA

Fahad Shahbaz Khan
fahad.khan@liu.se
MBZ University of Artificial Intelligence
UAE

Salman Khan
salman.khan@mbzuai.ac.ae
MBZ University of Artificial Intelligence
UAE

Mubarak Shah
shah@crcv.ucf.edu
University of Central Florida
USA

ABSTRACT
We propose a novel self-supervised Video Object Segmentation (VOS) approach that strives to achieve better object-background discriminability for accurate object segmentation. Distinct from previous self-supervised VOS methods, our approach is based on a discriminative learning loss formulation that takes into account both object and background information to ensure object-background discriminability, rather than using only object appearance. The discriminative learning loss comprises cutout-based reconstruction (cutout region represents part of a frame, whose pixels are replaced with some constant values) and tag prediction loss terms. The cutout-based reconstruction term utilizes a simple cutout scheme to learn the pixel-wise correspondence between the current and previous frames in order to reconstruct the original current frame with added cutout region in it. The introduced cutout patch guides the model to focus as much on the significant features of the object of interest as the less significant ones, thereby implicitly equipping the model to address occlusion-based scenarios. Next, the tag prediction term encourages object-background separability by grouping tags of all pixels in the cutout region that are similar, while separating them from the tags of the rest of the reconstructed frame pixels. Additionally, we introduce a zoom-in scheme that addresses the problem of small object segmentation by capturing fine structural information at multiple scales. Our proposed approach, termed CT-VOS, achieves state-of-the-art results on two challenging benchmarks: DAVIS-2017 and Youtube-VOS. A detailed ablation showcases the importance of the proposed loss formulation to effectively capture object-background discriminability and the impact of our zoom-in scheme to accurately segment small-sized objects.

KEYWORDS
self-supervision, video object segmentation, cutout, tagging, pretext tasks, attention

1 INTRODUCTION
Video object segmentation (VOS) aims to segment an object of interest in a video, given its segmentation mask in the first frame. VOS plays an important role in numerous real-world applications, such as interactive video editing, autonomous driving, augmented-reality, and smart surveillance systems. The problem is challenging since the target object is only provided with a reference segmentation in the first frame, and determining the object segmentation mask in all other frames involves: a. accounting for drastic changes in the appearance of the object of interest, over time, due to occlusion, b. distinguishing the object from highly similar surrounding distractor elements, and c. estimating finer object details, especially in case of a small object or object diminishing in size across the video sequence.

Most existing VOS approaches [4, 9, 33, 34, 42, 45, 54] typically rely on extensive supervision during training in the form of ground-truth human annotations (see Figure 1). Specifically, a common strategy is to fine-tune a network backbone, conventionally pretrained on ImageNet [5], using additional data (e.g. MS COCO [26], Pascal VOC [11], DAVIS [35], YouTube-VOS [55]) with pixel-level supervision in the form of segmentation masks. Different from

Figure 1: Comparison in terms of overall accuracy \( J&F(\text{Mean}) \) and pixel-level annotations required for different VOS methods on DAVIS-2017. The proposed CT-VOS outperforms existing self-supervised VOS methods. Different VOS methods shown here in comparison are: DINO [3], STC [17], MAST [21], UVC [24], CorrFlow [22], CycleTime [51], LCM [16], RMNet [53], SST [10], STM [33], PReMVOS [27], FEELVOS [44], OSVOS S [28], OnAVOS [46], OSVOS [2], SiamMask [50], OSMN [57].
these supervised approaches, recent self-supervised VOS methods [21, 22, 48, 51, 56, 60] perform segmentation without utilizing additional supervised training in the form of pixel-level ground-truth annotations. Several of these self-supervised approaches learn pixel-wise correspondence between frames by reducing the reconstruction error [21, 48], whereas others focus on matching pixels across frames using inherent cyclic consistency in videos [22, 51]. While promising results have been achieved by these self-supervised approaches, the performance still remains far from their supervised counterparts.

Typically, self-supervised VOS [21] approach employs robust pixel matching strategies to learn prominent representations of the target region while paying minimal focus on learning the feature embedding of the background region. Our methodology, however, is based on introducing feature diversity by taking into account both object and background information for object-background discriminability. The proposed cutout and tagging learning scheme encourages the object prediction model to focus on both prominent and non-prominent features of the target object as well as the background appearance, providing superior discriminative abilities. In addition, we alleviate a common problem encountered in several existing VOS methodologies that of loss of details, especially in small objects.

We introduce a self-supervised VOS approach, CT-VOS, that predicts the segmentation mask for the current frame based on the information in previous frames. Instead of learning trivial solutions from standard self-supervised color-based reconstruction loss between the current frame and previous frames, we utilize a two-fold discriminative learning loss formulation. First, instead of simply reconstructing the current frame, we also predict cutout in the current frame based on cutout in previous frames. Here, cutout represents a part of an image, whose pixels are replaced with some constant values e.g., zero. Intuitively, cutouts effectively simulate occlusion-based discontinuities and thereby aid the model to deal with occlusions during inference. Second, the model predicts tags in the current frame, such that the tags of all pixels in the cutout region are similar, and also considerably different from tags of non-cutout pixels. Here, we generate a heatmap of per-pixel identity tags for the current frame, where each tag could be any arbitrary value between 0 and 1. Our tagging loss enables separating the cutout pixels from the reconstructed image pixels, thereby improving the VOS performance by mimicking foreground-background separation. These complementary formulations are designed to allow the model to learn a rich representation, rather than learning a trivial solution of assigning the exact same color to pixels. Furthermore, we utilize zoom-in views of a random frame that facilitates the model to deal with objects at different scales. To summarize, our contribution is multifold:

• We present an object-background discriminative loss formulation comprising a cutout-based reconstruction element and tagging loss component. While cutout-based reconstruction forces the model to focus on both salient as well as less salient aspects of the object of interest by simulating controlled spatio-temporal occlusion-based discontinuities, tagging loss effectively separates the object of interest from the background pixels by mimicking foreground-background separation.
• We introduce zoom-in views in our VOS architecture that enables processing small spatial regions and thereby captures fine-grained object details. The zoom-in scheme not only assists with tracking smaller objects but also improves the segmentation mask details generated for larger objects.
• Comprehensive qualitative and quantitative experiments are performed on two challenging benchmarks: DAVIS-2017 [35] and Youtube-VOS [55]. Our results reveal the benefits of object-background discriminative loss formulation and zoom-in scheme, outperforming all published self-supervised VOS methods on both benchmarks.

2 RELATED WORK

Hand-crafted features based on appearance, motion, spatial proximity, and neighborhood grouping constraints have been used in earlier works [1, 12, 18, 30] on VOS. The need for extensive manual annotations in a supervised setting has encouraged researchers to venture beyond the supervised paradigm into the realm of self-supervised VOS. Based on the nature of supervision, existing VOS approaches can be roughly categorised into following three groups:

Supervised VOS Methods: A considerable volume of work on video object segmentation (VOS) task [4, 9, 33, 45, 54] focuses on segmenting each object instance separately. Several recent works [9, 42, 54], employ sequential modelling to achieve temporal consistency. Lately, a number of solutions have been motivated by embedding based approaches [34, 45], which deal with learning pixel-mappings between reference and target frames. One such prior work by Oh et al. [33], deals with binary-classification of each pixel to generate a single-object segmentation mask for a given frame by applying non-local matching with previously predicted frames. Another idea of the region based segmentation approach is used to extract multiple object proposals in a single feedforward path, followed by associating objects across frames [23, 27]. However, the dependence of these approaches on region proposal networks [14, 38] and ground-truth annotations introduces additional complexities.

Few-Shot Learning Based Segmentation: Few-shot segmentation deals with learning to segment objects based on a limited set of annotated examples. Initial literature [7, 40, 41, 58] in this area focused on using labeled segmentation masks for few examples in the support set. Some of the latest methods [36, 49, 59] have worked with weaker annotations based on scribble or bounding boxes. Raza et al. [37] recently proposed using image-level labels to solve the supervision requirement in a few-shot segmentation task. Although this is a promising direction, these formulations are labor-intensive, especially in robotic applications that require learning online.

Self-supervised Segmentation: Self-supervised methods enable models to learn without manually generated supervisory signals, thereby eliminating the overhead of large annotated datasets. Generally, computer vision pipelines that employ self-supervised learning employ two tasks: a pretext task and a downstream task. Although the downstream task is the ultimate goal, the pretext task is the primary task that the self-supervised method seeks to solve. In the
Figure 2: Overview of the proposed CT-VOS training. Firstly, the current frame $I_t$ is used to generate query $Q_t$, and the past frames $I_{t−1},...,I_{t−7}$ are employed to compute a comprehensive key set $K_t$ (shown on the left). Next, the past frames with cutout $I'_{t−1},...,I'_{t−7}$ are processed to yield the value set $V_t$ (shown on the right). Then, the generated query, key and value are employed to predict the reconstructed frame with cutout $I^\text{pred}_t$ and tag channel $P_t$ (shown on the bottom right). In order to learn finer object details, we introduce zoom-based strategy by choosing a random frame $I_t$ (shown on the left). Here, we introduce 3 different zoom levels in a chosen random frame. Therefore, with these three frames, the total number of frames, $T$, corresponds to 10 frames. ⊗ implies dot product.

3 APPROACH

Given the first frame segmentation mask of an object, the proposed network segments the object throughout the video clip based on the context provided by the previous frames. Figure 2 illustrates the overall network architecture. We exploit the automatically available supervisory signals from data by framing VOS as a self-supervised learning problem. To this end, we introduce object-background discriminative loss formulation comprising cutout-based reconstruction and tag (cutout vs. non-cutout) prediction. While one of the components helps mimic occlusion-based spatio-temporal discontinues, the other encourages the model to generate well-separated foreground-background segmentation masks. We also utilize zoom-in views of a random frame that enriches the overall representation by capturing the segmentation details at different scales.

Initially, the current frame is processed through an encoder setup to generate current frame embedding - query (Figure 2). Next, previous frames are encoded to past frame embeddings - keys. The query interacts with keys to generate similarities between the current and previous frame content. Then, the values are extracted using previous frames with introduced cutout patches. Finally, the relative matching scores and values are fed to the decoder to generate the reconstructed image with cutout prediction and a tag map. During inference stage, the values are generated using previously predicted object segmentation masks, and the decoder yields object segmentation mask along with the tags of the current frame.

3.1 Architecture

Here, we elaborate the preliminaries of our network presented in Figure 2 and Figure 3. Using a ResNet-18 [15] backbone network $Enc_{cur}$, we encode the current frame $I_t$ into current frame embedding or query $Q_t$. Next, the past frames $I_{t−1}...I_{t−7}$ are encoded by $Enc_{key}$ to generate keys $K_{t−1}...K_{t−7}$. These keys are concatenated to form a comprehensive key $K_t$. Using the query and the key, attention is computed, which is used to determine the strength of the correspondence between pixels from the current and the past frames. During training, past frames introduced with cutouts $I'_{t−1}...I'_{t−7}$ are passed through $Enc_{cut}$. The resultant generated values $V_{t−1}...V_{t−7}$ (collectively termed as $V_t$) are then combined with the query and the key. We utilize this resultant representation as
input to the decoder $Dec_{cur}$, which yields the reconstructed image with cutout $i_t^{pred}$ and tags $P_t$.

The model comprises ResNet-18 as the backbone network for current frame encoder $Enc_{cur}$, past frames encoder $Enc_{key}$ as well as value encoder $Enc_{val}$. The base features are computed using stage-4 (res4) of the ResNet-18 model. For each input frame $I_t$, we have a triplet $(Q_t, K_t, V_t)$ referring to Query, Key, and Value. During training, reconstruction of the cutout-based current frame ($I_t^{pred}$) is accomplished by combining pixels from the current frame ($I_t$), with pixels of past frames through attention.

To implement cutout, we apply a fixed-size zero mask to a random location in the frame. The selected random location for the cutout is retained for each image within a 10-frame clip passed through $Enc_{val}$, as shown in Figure 2. In addition, we apply a cutout size constrain such that at least 50% of the image remains unmodified. Concurrently, we generate multi-scale feature embeddings for one of the random past frames. This helps in dealing with multiple object sizes and finer deals, thereby enhancing the robustness of the method.

During inference, however, as illustrated in Figure 3, the approach focuses on generating object segmentation mask $M_t$ for time frame $t$ (current frame). While the network is trained on $V_t$ comprising 3-channel cutout-based RGB frames, we generate 3-channel $V_t$ from 1-channel segmentation masks $M_{t-1}...M_{t-7}$ during inference by stacking each object mask 3 times. Also, during training, the resultant output $O_t$ is a cutout-based reconstructed RGB frame, therefore inference involves computing resultant 1-channel segmentation output $M_t$ by averaging 3-channel output from the decoder $Dec_{cur}$ normalized to the range [-1, 1] and further thresholded at 0. Additionally, we do not use the tags generated during inference.

### 3.2 Proposed Loss Formulation

To train our network, we introduce two new losses, namely self-supervised cutout-based reconstruction loss and tagging loss, detailed below:

$$L_{overall} = L_{reconstruction} + \lambda L_{tagging}. \quad (1)$$

The reconstruction loss with cutout ensures that the model accounts for the less significant features within the frames as much as it focuses on the salient features. This better equips the VOS model with occlusion-based spatio-temporal discontinuities that it would encounter during inference. In addition, the self-supervised tagging loss enforces the separation of the object from the background to boost the segmentation performance. We use a combination of cutout-based reconstruction loss and $\lambda$ weighted self-supervised tagging loss to train our network for segmentation (equally weighted losses give us the best results).

**Reconstruction Loss with Cutout:** The objective $L_{reconstruction}$ to distinguish between the reconstructed frame $i_t^{pred}$ and input frame $I_t$ (we are dropping subscript $t$ below for convenience), is defined in terms of Huber Loss as follows:

$$L_{reconstruction} = \frac{1}{N} \sum_{j=1}^{N} z_j. \quad (2)$$

where $z_j = \begin{cases} 0.5 |i_j^{pred} - I_j|^2 & \text{if } |i_j^{pred} - I_j| < 1 \\ |i_j^{pred} - I_j| - 0.5 & \text{otherwise} \end{cases} \quad (3)$

Here, $N$ corresponds to total number of pixels in the image, and we assume pixel values in $I$ and $i^{pred}$ are normalized between -1 and 1.

**Self-Supervised Cutout-Based Tagging Loss:** We introduce the self-supervised cutout-based tagging loss that comprises a cutout pixel pull loss, $L_{cp}$, and cutout pixel-to-reconstructed image pixel push loss, $L_{cp}$, for the separation of pixels in the VOS task. Suppose, $h_{w'}$ is the predicted tag (or the regressed value, i.e. continuous values between 0 and 1) for a given $w^{th}$ random pixel in the cutout region or the reconstructed image region, then let us define $h_{w'}$, that is the mean of tags for the group of $W$ random pixels as:

$$h_{w'} = \frac{1}{W} \sum_{w=1}^{W} h_w. \quad (4)$$

In order to define $L_{cp}$, we only work with randomly selected subset of cutout pixels, say $m$ where $m = 1, ... M$. Here, $h_m$ corresponds to predicted tag for $m^{th}$ cutout pixel, while $h_M$ refers to mean of tags of $M$ cutout pixels. Formally, we define $L_{cp}$ below which pulls the tags of all cutout pixels closer as:

$$L_{cp} = \frac{1}{M} \sum_{m=1}^{M} (h_m - h_M')^2. \quad (5)$$

In addition to pulling the tags of cutout pixels together, we also try to push apart the cutout pixels from the reconstructed image pixels. We introduce a margin $G$ to provide a permissible gap on the difference in the embedding space of cutout pixels $C$ and reconstructed image pixels $B$. Given the mean of cutout pixel tags $h_c'$ and
the mean of reconstructed image pixel tags $h'_p$, the complementary margin-based penalty loss $L_{rp}$ is given by:

$$L_{rp} = \max (0, G - ||h'_p - h'c||).$$  \hspace{1cm} (6)

The self-supervised cutout-based tagging loss $L_{\text{tagging}}$ that helps with separation of cutout pixels and reconstructed image pixels is defined as:

$$L_{\text{tagging}} = L_{cp} + L_{rp}.$$  \hspace{1cm} (7)

4 EXPERIMENTS

Datasets: We evaluate our approach on two VOS benchmarks: DAVIS-2017 [35] and Youtube-VOS [55]. DAVIS-2017 consists of 120 videos with 60 training, 30 validation, and 30 testing sequences. Youtube-VOS comprises 4,453 annotated videos of 91 object classes.

Training: The backbone network constitutes the ResNet-18 module and additional layers that are randomly initialized. We use random crops as well as spatial and temporal flips to augment the training data. Next, the generated single clip of 8 frames is split into past frames (initial 7 frames) and the current frame ($8^{th}$ frame). In addition, we have 2 zoomed-in frames for one of the randomly chosen past frames, resulting in overall 10 frames per batch in a given epoch. The objective function used to optimize the network is discussed in detail in Section 3.2. For fair comparison with existing methods, we use the same training strategy as the previous self-supervised VOS works. We present the quantitative results on DAVIS-2017 validation set using backbone networks trained only on datasets mentioned under the 'Dataset' column in Table 1, and the model used to report results on YouTube-VOS validation (Table 2) is trained only using the Youtube-VOS. Our method converges on the Youtube-VOS dataset in around 400 epochs. Additionally, the model is trained using a learning rate of 0.0001 and Adam Optimizer on a Nvidia V100 GPU.

Inference: Figure 3 summarizes the testing stage. Similar to the training process, we process 10 frames at-a-time during inference. Also, during inference, similar to training phase, our model generates two outputs: 3-channel output and 1-channel of tags. Tags are primarily used for enforcing object-background separation during training and not used for mask generation during inference, which is handled by a dedicated 3-channel output (2nd row Figure 5). The predicted segmentation mask (final mask as shown in Figure 5) is determined by simply averaging the generated labels from the 3-channel output. The final segmentation generated for each current frame is, then, used as input for the subsequent clip.

Evaluation Metrics: We use the following metrics to evaluate the predicted multi-object video segmentation masks in comparison to its ground truth:

- Region similarity $J$: Defined by the intersection-over-union between the predicted segmentation and ground-truth mask. It provides a measure of the mislabeled pixels.

Table 2: State-of-the-art comparison on Youtube-VOS validation set.

| Method          | Backbone | Dataset (Size) | $J$ & $F$ | $\text{mean}_{\text{seen}}$ | $\text{mean}_{\text{unseen}}$ |
|-----------------|----------|----------------|-----------|------------------------------|-------------------------------|
|                 |          |                | $\text{G} (%)$ | $\text{J} (%)$ | $\text{F} (%)$ |
| **Unsupervised**|          |                |           |                          |                              |
| Vid. Color [48] | ResNet-18 | Kinetics (800 hrs) | 34.0 | 34.6 | 32.7 |
| CycleTime [51]  | ResNet-50 | VLOG (344 hrs)  | 48.7 | 46.4 | 50.0 |
| CorrFlow [22]   | ResNet-18 | OxUvA (14 hrs)  | 50.3 | 48.4 | 52.2 |
| UVC [24]        | ResNet-18 | Kinetics (800 hrs) | 59.5 | 57.7 | 61.3 |
| MAST [21]       | ResNet-18 | YT-VOS (5.58 hrs) | 65.5 | 63.3 | 67.6 |
| **CT-VOS (Ours)**| ResNet-18 | YT-VOS (5.58 hrs) | 68.2 | 67.1 | 69.2 |
| **CT-VOS (Ours)**| ResNet-18 | Kinetics (800 hrs) | 68.3 | 65.5 | 71.0 |
| **CT-VOS (Ours)**| ResNet-18 | Kinetics (800 hrs) | 69.8 | 68.3 | 71.3 |
| DINO [3]        | ViT [8]  | ImageNet [39]  | 71.4 | 67.9 | 74.9 |
| **CT-VOS (Ours)**| ResNet-50 | Kinetics (800 hrs) | 72.6 | 69.7 | 75.4 |

Table 1: State-of-the-art comparison on DAVIS-2017 validation set. Here, reference dataset notations are: I=ImageNet, V=ImageNet-VID, C=COCO, D=DAVIS, M=Mapillary, P=PASCAL-VOC, Y=YouTube-VOS.
Figure 4: Qualitative results of our method on DAVIS-2017 (Rows 1, 3, 5) and Youtube-VOS (Rows 2, 4, 6). The first two rows depict the ability of the model to effectively handle visual discontinuities, namely occlusion; the next two rows illustrate that our model deals well with separating objects from a background even with highly similar distractor objects; the later rows demonstrate the model’s competency with generating fine-grained segmentation details for large as well as small objects.

Figure 5: Decoder output visualizations during inference
- Contour precision $F$: Measures the contour-based precision and recall between the contour points. It is an indicator of precision of the segmentation boundaries.
- Overall score $G$: Average score of $J$ and $F$ measures.

4.1 State-of-the-art Comparison
Table 1 presents our quantitative results on DAVIS-2017 validation set. Here, we observe that proposed method outperforms existing published self-supervised work on VOS. Similarly, the results on YouTube-VOS validation set in Table 2 further affirm its ability to generate compelling segmentations. We, also, visualize the results on DAVIS-2017 and YouTube-VOS in Figure 4. Furthermore, Figure 6 highlights the improvements in segmentation that our work exhibits on DAVIS-2017 over existing self-supervised VOS methods.

| Method                      | $J \& F$ | $J_{\text{mean}}$ | $F_{\text{mean}}$ |
|-----------------------------|----------|--------------------|--------------------|
| Reconstruction Loss W/O Cutout | 59.7     | 58.4               | 60.9               |
| Reconstruction Loss W Cutout   | 63.9     | 63.6               | 64.2               |
| + Tagging Loss               | 66.3     | 65.9               | 66.7               |
| + Zoom-in Module             | 68.2     | 67.1               | 69.2               |

Table 3: Ablation when components are added sequentially on DAVIS-2017 validation set.

| Method              | $J \& F$ | $J_{\text{mean}}$ | $F_{\text{mean}}$ |
|---------------------|----------|--------------------|--------------------|
| Circle              | 67.1     | 65.9               | 68.2               |
| Scalene Triangle    | 67.3     | 66.0               | 68.5               |
| Rectangle           | 67.9     | 66.7               | 69.1               |
| Square              | 68.2     | 67.1               | 69.2               |

Table 4: Impact of varying cutout shapes on DAVIS-2017 validation set.

4.2 Ablation Study
We demonstrate the effectiveness of individual model components by adding the components sequentially. Table 3 summarizes the experiments performed on the DAVIS-2017 dataset with our baseline model using ResNet-18 backbone. In addition, Figure 7 illustrates the significance of each proposed component based on the qualitative analysis performed on DAVIS-2017.
Figure 6: Qualitative results on DAVIS-2017 for comparison of our method (Rows 4, 8) against existing self-supervised VOS approaches, namely MAST [21] (Rows 1, 5), STC [17] (Rows 2, 6) and DINO [3] (Rows 3, 7). Row 4: Ability to separate objects from highly similar surrounding distractor elements. Row 8: Generation of fine-grained segmentation details and competency to deal with occlusion.

| Method   | J&F | Jmean | Fmean |
|----------|-----|-------|-------|
| Lcp      | 65.6| 66.8  | 64.4  |
| Lrp      | 66.7| 65.8  | 67.5  |
| Lcp + rp | 68.2| 67.1  | 69.2  |

Table 5: The role of pull and push components in cutout-based tagging loss.

| Method   | J&F | Jmean | Fmean |
|----------|-----|-------|-------|
| BCE Loss | 65.9| 66.2  | 65.5  |
| Tagging Loss | 68.2| 67.1  | 69.2  |

Table 6: Ablation for binary cross-entropy loss v/s tagging loss.

**Reconstruction Loss With Cutout:** To investigate the influence of standard colorizing loss on the overall system design, we create a base model and analyze the resulting output (see Table 3). We introduce the LAB color palette to leverage the decorrelated color space, and augment the data using random crops and spatio-temporal flips in the baseline network. The qualitative results generated by this model with conventional colorizing loss exhibits significantly deteriorated object segmentation as seen in Figure 7(a). During

![Without Cutout](image1.png)

![With Cutout](image2.png)

(a) Cutout-based reconstruction

![Without Tagging Loss](image3.png)

![With Tagging Loss](image4.png)

(b) Tagging loss

![Without Zoom-in Scheme](image5.png)

![With Zoom-in Scheme](image6.png)

(c) Zoom-in scheme

Figure 7: Our qualitative results depicting the impact of each component, on DAVIS-2017, suggest that (a) Introducing cutout-based reconstruction improves model’s ability to perceive spatio-temporal discontinuities (occlusion in our case). The rear-end of motorbike that is partially occluded is properly segmented by our model using cutouts. (b) Tagging loss helps generate accurate segmentation masks for the left most person, despite the headgear blending with the surrounding pixels, thereby validating its foreground-background decoupling capabilities. (c) Fine details of paragliding backpack are captured by our zoom-in scheme.
training, introducing cutout-based reconstruction of the images forces the model to concurrently focus on both salient as well as less salient features within the frames. Cutouts simulate occlusion-based scenarios that the model is likely to be subject to during the testing phase. It, therefore, inherently improves the generated object segmentations. Additionally, detailed analysis about shapes (refer Table 4) and sizes (refer to Table 3 from the supplementary material) of cutouts make it evident that the shape variations have inconspicuous effects on the overall performance, compared to change in the size of the cutout. The disproportionately large size of the cutout is associated with significant loss of context.

**Self-supervised Cutout-Based Tagging Loss:** Here, we evaluate the effectiveness of the self-supervised tagging loss in our model. Adding the self-supervised tagging loss improves the overall segmentation score of the network by \( \sim 2\% \) as depicted in Table 3. In Table 5, we test the model performance by introducing each component of the loss. Based on the analysis of qualitative results we notice that eliminating \( L_{cp} \) causes segmentation of an individual object instance to separate into smaller disjoint clumps, while \( L_{rp} \) plays a significant role in minimizing the background pixels overlapping foreground pixels of objects around edges. Here, \( L_{cp} \) serves as a pulling force to bring together all the pixels associated with the background, while \( L_{rp} \) pushes apart object-background pixels in close vicinity that are not related each other. Overall, the tagging loss is a clustering/margin maximization-based objective. For a comprehensive ablation study, we apply binary cross-entropy loss instead of tagging loss and compare the results in Table 6. We train the model using binary cross-entropy by assigning the cutout pixels as label 0 and the rest as 1. The quantitative results validate the ability of tagging loss to improve model performance by introducing intricate foreground-background separation.

**Zoom-in Views:** In the final ablation study, we test the importance of zoom-in views in the network. We find that the zoom-in views capture the finer structural information and thereby promotes the prediction of finer details for small objects. Table 3 shows the impact of the zoom-in views on the quantitative results, and we see an increase of \( \sim 2\% \) in the overall performance. Introducing the zoom-in across multiple randomly selected past frames (see Table 7), however, fails to improve the resulting output. Since the zoom-in component focuses on exploiting the spatial information, adding it across one of the single past frame or multiple past frames improves the model outcomes by a similar value (refer to Figure 8). The proposed zoom-in structure essentially forces the model to find clean signal from multi-scale correspondence, thereby enhancing its robustness to determining correlation.

### 5 CONCLUSIONS

In this work, we have proposed a self-supervised video segmentation network, which exhibits improved performance. These capabilities are driven by the self-supervised cutout-based prediction and tagging loss that ensures separation of foreground-background pixels. Additionally, the zoom-in views helps by retaining the elaborate details of objects, thereby providing the context needed for segmenting smaller objects. The overall quantitative results provide the measure of the effectiveness of the network, while the ablation studies highlight the distinct contributions of each module. Our system transforms the general reconstruction-based self-supervised approach for VOS, and has the potential to be extended to more challenging segmentation and tracking tasks.

### REFERENCES

[1] Thomas Brox and Jitendra Malik. 2010. Object Segmentation by Long Term Analysis of Point Trajectories. In Proceedings of the European Conference on Computer Vision. Springer Berlin Heidelberg, Berlin, Heidelberg, 282–295.

[2] Sergi Caelles, Kevit-Kokutsu Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. 2017. One-Shot Video Object Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging Properties in Self-Supervised Vision Transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 9650–9660.

[4] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. 2018. Blazingly Fast Video Object Segmentation With Pixel-Wise Metric Learning. In Proceedings
[54] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian L. Price, Scott Cohen, and Thomas S. Huang. 2018. YouTube-VOS: Sequence-to-Sequence Video Object Segmentation. In Proceedings of the European Conference on Computer Vision, Vol. abs/1809.00461. 603–619.

[55] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. 2018. Youtube-vos: A large-scale video object segmentation benchmark. arXiv preprint arXiv:1809.03327 (2018).

[56] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. 2021. Self-supervised Video Object Segmentation by Motion Grouping. arXiv preprint arXiv:2104.07658 (2021).

[57] L. Yang, Y. Wang, X. Xiong, J. Yang, and A. K. Katsaggelos. 2018. Efficient Video Object Segmentation via Network Modulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 1286–1295.

[58] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. 2019. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 9587–9595.

[59] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. 2019. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 5217–5226.

[60] Fangrui Zhu, Li Zhang, Yanwei Fu, Guodong Guo, and Weidi Xie. 2020. Self-supervised Video Object Segmentation. arXiv preprint arXiv:2006.12480 (2020).