# Semantic-Aware Dynamic Parameter for Video Inpainting Transformer

Eunhye Lee[1*], Jinsu Yoo[2*], Yunjeong Yang[2], Sungyong Baik[2,3], Tae Hyun Kim[1†]
{dldms1345, jinsuyoo, yunjeongyang, dsybaik, taehyunkim}@hanyang.ac.kr
[1]Dept. of Computer Science, [2]Dept. of Artificial Intelligence, [3]Dept. of Data Science, Hanyang University

## Abstract

*Recent learning-based video inpainting approaches have achieved considerable progress. However, they still cannot fully utilize semantic information within the video frames and predict improper scene layout, failing to restore clear object boundaries for mixed scenes. To mitigate this problem, we introduce a new transformer-based video inpainting technique that can exploit semantic information within the input and considerably improve reconstruction quality. In this study, we use the mixture-of-experts scheme and train multiple experts to handle mixed scenes, including various semantics. We leverage these multiple experts and produce locally (token-wise) different network parameters to achieve semantic-aware inpainting results. Extensive experiments on YouTube-VOS and DAVIS benchmark datasets demonstrate that, compared with existing conventional video inpainting approaches, the proposed method has superior performance in synthesizing visually pleasing videos with much clearer semantic structures and textures.*

## 1. Introduction

A constant increase in demand for video content in our daily lives (*e.g*., YouTube or TikTok) has led to the development of video inpainting methods, which aim to complete a missing region or erase unwanted areas such as watermarks and captions from a given input video. In particular, recent convolutional neural networks (CNNs) for video inpainting tasks have shown promising results [34, 15, 2, 43, 37, 9]. In addition, transformer-based inpainting networks [39, 26, 23, 29] significantly escalate the quality of inpainting results through their large network capacity and superior local/global connectivity based on attention mechanisms.

However, conventional approaches are unable to fully utilize the semantic information within the input video or distinguish the class-specific characteristics of objects with different semantics. As a result, they frequently fail to recover proper object structure, texture, and scene layout. Several studies [32, 24, 19, 25] have suggested that utilizing semantic information can lead to better results and produce visually more plausible images when filling missing regions. However, research on video inpainting has yet to explore the potential of incorporating semantic maps.

Video inpainting is a challenging task that requires to render temporally consistent video frames. To address this issue, optical flow has been extensively explored, although it is computationally intensive [15, 37, 9, 23, 41, 40, 13]. In this study, we demonstrate that semantic information can also be used to enforce temporal consistency, and we introduce a new technique that further facilitates the use of semantic cues in a video by dynamically mixing multiple class-specific experts to handle objects with distinct semantics adaptively.

Although previous semantic-guided inpainting networks used the predicted segmentation map as additional input [32] or developed a dedicated semantic-aware inference module [24] to learn semantic-aware parameters, the restoration performance is limited, given the use of shared network parameters for the different semantics. This problem has been addressed in a recent study [25] by using a semantic-aware attention module that enables the features to attend solely to regions with identical semantic labels. However, the computational cost increases proportionally to the number of classes because the attention mechanism is separately carried out for each category.

To mitigate these problems, we propose a novel semantic-aware dynamic parameter selection approach that effectively utilizes the semantic information within input video frames while retaining the number of operations during the inference phase. Specifically, inspired by CondConv [38], we introduce conditionally parameterized linear operations to learn semantic-aware experts, and produce locally varying (dynamic) parameters by leveraging the given semantic cues based on the notion of mixture-of-experts [31, 30, 38]. Our linear operations with dynamically determined parameters replace the standard feed-forward operation within a conventional transformer block. Notably, we can keep the number of parameters for inference by mixing the expert parameters before performing linear operations rather than aggregating the output features cal-

[*] Equal contribution.
[†] Corresponding author.

culated by each expert. Moreover, in contrast to the original CondConv [38] that produces conditional, but locally uniform parameters, ours can generate conditional and locally (token-wise) varying parameters, thus improving each token's representation power.

Our extensive experiments witness the outstanding performance of the proposed **S**emantic-**A**ware **V**ideo **I**npainting **T**ransformer (**SAVIT**) in improving video inpainting results. Specifically, SAVIT elevates quantitative performance on conventional video inpainting benchmark datasets (YouTube-VOS [36] and DAVIS [28]) and produces visually more superior results against state-of-the-art methods. We summarize our contribution as follows:

- We tackle leveraging semantic information within the given input video frames for the video inpainting task.
- We introduce a novel semantic-aware video inpainting transformer based on a mixture-of-experts scheme.
- We propose a semantic-aware dynamic linear operation to exploit local semantic cues effectively.
- Extensive experiments demonstrate the superiority and efficacy of our method, especially in recovering semantic structures and textures.

## 2. Related Work

**Video inpainting.** An active research topic in video inpainting is generating a temporally consistent video. Various approaches, including 3D-CNN-based, flow-based, and transformer-based methods, have been proposed to fill in a missing region inside a video plausibly. Despite their ability to extract spatio-temporal information, 3D-CNN methods [34, 2, 3] fail to fully utilize global correspondence due to limited temporal receptive field. Flow-based methods [15, 37, 9, 23, 41, 40, 13] first estimate motion flow in the missing region and use the flow to predict the missing pixels. However, these methods highly rely on pre-trained inpainting networks and often require manual operations [37, 9]. Recently, transformer-based methods [39, 26, 23, 29] have attracted research attention due to their superiority in capturing information across long-ranged frames. Built upon the vision transformer architecture [8], FuseFormer [26] proposed soft tokenization with spatial overlapping to interact information among neighboring tokens. Followup studies have improved the performance with dedicated modules such as focal attention [23] or discrete latent mapping [29]. In our study, we develop our method on top of transformers to capture long-range dependencies and advocate the effectiveness of leveraging semantic information.

**Semantic-guided inpainting.** Several works have been proposed to utilize the semantic map for inpainting [32, 24, 25], and its related fields [14, 4, 22, 12]. Among them, SPGNet [32] first fills the missing segmentation map, then synthesizes the inpainted image based on the estimated semantic results. Moreover, SGENet [24] introduced iterative feature refinement using semantic map-based normalization [27] to handle more complex holes in mixed scenes, including objects with different semantics. Recent work [25] proposed to calculate semantic-wise attention to learn the corresponding semantic information. However, most of the previous works have developed upon image inpainting, and leveraging semantic cues within a video has yet to be explored due to the lack of annotated data. Our study mitigates the data acquisition problem by gathering pseudo-annotated videos with pre-trained segmentation network [17], and demonstrates that semantic information can elevate video inpainting performance.

**Dynamic filter.** As an alternative to fixed filter, scene-adaptive dynamic filter networks [11, 7, 42] adapt their parameters based on a given input image and/or features. These methods allow flexibility in the network in treating various inputs with different characteristics. Notably, CondInst [33] dynamically generates instance-aware parameters for instance segmentation. In this study, we adapt the transformer parameters in reasoning each token conditioned to its features and semantic information.

**Mixture-of-experts.** By combining the results from multiple expert models, the network can further improve its representation power [30, 10, 31]. Instead of increasing the computational cost during the inference proportional to the number of experts, recent works [38, 5] parameterized the mixed experts to combine the knowledge efficiently. Our work is motivated by CondConv [38]; we mix the experts' parameters before the feature calculation. Specifically, we divide the experts' roles by semantic categories, then combine semantic-aware experts' knowledge at each token.

## 3. Proposed Method

In this section, we first introduce the annotation procedure to acquire the semantic maps for videos, followed by a detailed description of the proposed framework **SAVIT**, which leverages semantic information for video inpainting.

### 3.1. Semantic data acquisition

Unlike previous semantic-aware approaches [32, 25, 24], which simultaneously estimate the segmentation maps for inpainting, we assume that semantic segmentation maps are predictable from the given clean video frames, which include unmasked regions to remove, using off-the-shelf segmentation methods. However, currently available video segmentation datasets, such as DAVIS [28] or CityScape [6], are limited to specific categories and/or particular objects, leading acquiring ground-truth segmentation maps for general video frames difficult. Alternatively, we utilize a fully pre-trained panoptic segmentation network [17] to predict the foreground and background semantics in more general scenes.

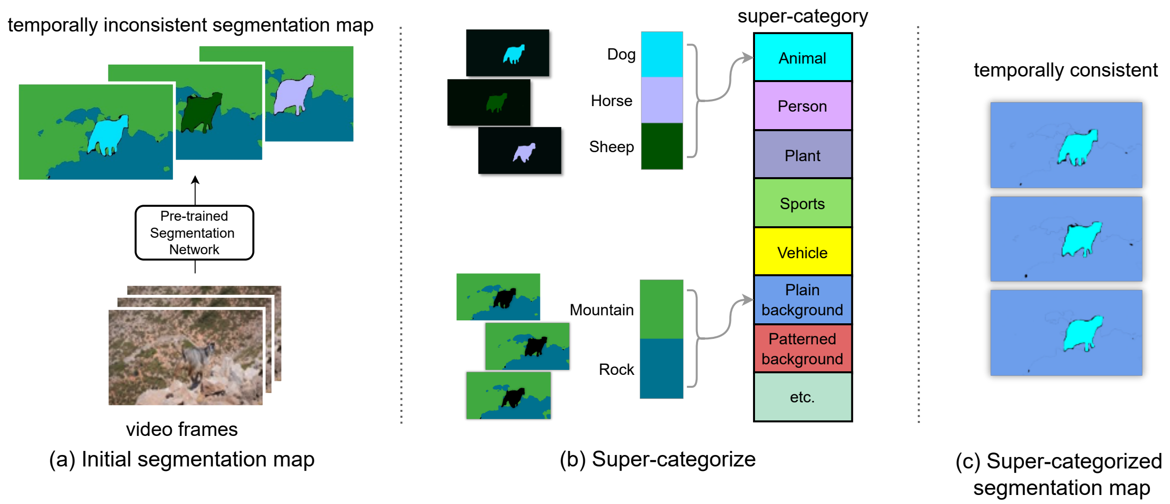


Figure 1: Overview of the semantic data preparation: (a) Temporally inconsistent rough prediction using a pre-trained panoptic segmentation model; (b) Grouping raw predictions into corresponding super-categories; (c) Super-categorized segmentation map produces temporally consistent labels.

Fig. 1 illustrates the overall semantic data acquisition process used for training and inference. For training, we first apply the panoptic segmentation network [17] to collect estimated segmentation maps for video frames in conventional video inpainting datasets (*e.g*., YouTube-VOS [36]). The initial segmentation results consist of 53 background and 80 foreground classes [1]. However, these initial segmentation maps with a large number of classes are erroneous and temporally inconsistent, as shown in Fig. 1 (a). Thus, we enforce the temporal consistency of semantics by grouping the predicted labels to the corresponding super-category of the class. For example, dog, horse, and sheep regions are grouped into a super-category of animal (Fig. 1 (b)). Although we can lose semantic details to some extent by the super-categorization, such constraint helps the training of our semantic-aware inpainting network through the enforced temporal semantic consistency (Fig. 1 (c)).

In our work, we define eight different super-categories based on the texture similarities and the frequencies of categories, which are measured on the entire training dataset. Specifically, we have five super-categories for foreground objects: ‘animal,’ ‘person,’ ‘vehicle,’ ‘plants,’ and ‘sports.’ Moreover, we have two super-categories for background according to their texture; ‘plain’ and ‘patterned.’ Finally, we group the rest of the remaining sub-categories into ‘etc.’ During the test phase, we also obtain the semantic maps in the same fashion.

### 3.2. Semantic-aware dynamic transformer

To handle diverse scenes with various objects across different classes in videos, we propose a semantic-aware dynamic transformer. In particular, our transformer employs a mixture-of-experts scheme, in which each expert is a single fully connected layer with its parameters reserved for each corresponding semantic class (*i.e*., super-category) obtained from Section 3.1. Conditioned on the semantic features of each token, we create a token-wise fully connected layer, where its parameters are determined by weighted summing the class-specific parameters, and the weights for the mixture are produced by our semantic router as in the following.

**Learning semantic router.** To produce locally (*i.e*., token-wise) varying parameters conditioned on the semantic information, the network requires a routing function that predicts blending weights to mix the class-specific expert parameters. Our router computes routing weights (*i.e*., mixture coefficients) for each token by:

$$r_i = MLP(f_i) + hardmax(s_i), \tag{1}$$

where $r_i \in \mathbb{R}^C$ is a produced mixture weight vector; $f_i$ is the feature of an input token; $s_i \in \mathbb{R}^C$ is a histogram of class distribution on the segmentation map for the given $i$-th token; $C$ is the number of semantic experts ($C = 8$ in this work unless stated otherwise); and $MLP$ is a standard multi-layer perceptron (MLP) composed of two fully-connected layers with Leaky ReLU activation and sigmoid output. Our router uses the most dominant semantic label from the input histogram $s_i$ by using a hard selection strategy ($hardmax$ operation) rather than a softmax result. In addition, we model our router to be lightweight (*i.e*., two layers) to minimize the overhead, keeping our framework to have similar computational costs to the standard transformers during inference.

**Semantic-aware dynamic parameterization.** By using the routing weights $r_i$ generated by the router for the $i$-th token with features $f_i$, we produce semantic-aware dynamic parameters $W_i$ for linear (fully-connected) layers of a transformer block. The formulation of semantic-aware dynamic

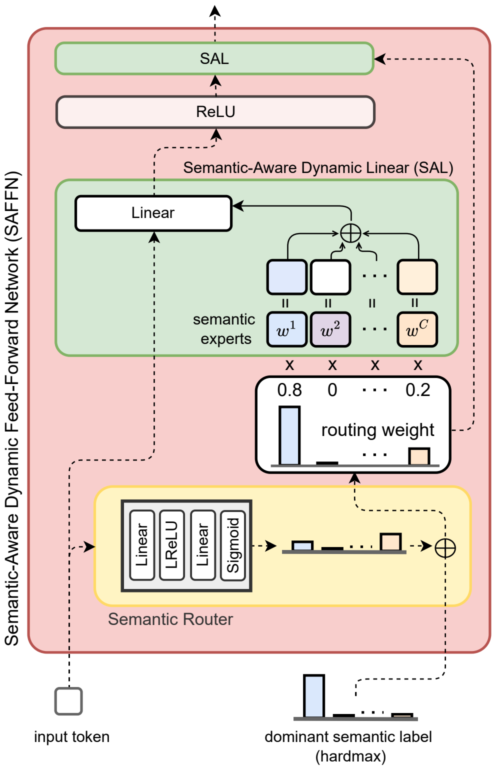

Semantic-Aware Dynamic Feed-Forward Network (SAFFN)
SAL
ReLU
Semantic-Aware Dynamic Linear (SAL)
Linear
semantic experts
$w^1$
$w^2$
$w^C$
x
0.8
0
0.2
routing weight
Linear
LReLU
Linear
Sigmoid
Semantic Router
input token
dominant semantic label (hardmax)

Figure 2: Illustration of **SAFFN**. With the histogram of semantic categories (*e.g*., hardmax) and additional linear layers, the semantic router outputs a routing weight for a specific token. Using the routing weight, **SAL** first acquires token-specific parameters by the weighted sum of experts, then operates the linear transformation of the token with the parameters. Here, we illustrate the operational flow of a single token. ⊕ indicates element-wise summation.

layer (**SAL**) is given by:

$$\mathrm{SAL}(f_i) = f_i \cdot W_i, \tag{2}$$

where the mixed expert parameter $W_i$ for token $f_i$ yields:

$$W_i = \sum_{c=1}^{C} r_i(c) \cdot w^c. \tag{3}$$

Here, $w^c$ indicates the parameters of each class-specific expert, while $r_i(c)$ is the $c$-th element of routing weight vector $r_i$ and means the mixture coefficient for the corresponding expert $w^c$. Consequently, our semantic-aware dynamic parameterization produces locally varying (token-wise) parameters. For example, a token corresponding to a goat in the input frame can be handled by parameters specialized for animal super-category, and another token on the rock can be dealt with specific parameters for the plain background. Notably, our framework can also handle a token of complex scenes where semantically different objects are coexisting. This is because our router is designed to produce locally varying weights and give high routing weights for experts corresponding to coexisting objects, rather than choosing a single expert.

Finally, we devise a semantic-aware dynamic feed-forward network (**SAFFN**) as depicted in Fig. 2. SAFFN includes two SAL using the dynamically generated semantic-aware parameters as:

$$\begin{aligned} \mathrm{SAFFN}(f_i) &= \mathrm{SAL}(\mathrm{ReLU}(\mathrm{SAL}(f_i; W_i^l)); W_i^{l+1}) \\ &= \max(\mathbf{0}, f_i \cdot W_i^l) \cdot W_i^{l+1}. \end{aligned} \tag{4}$$

In our SAFFN, each SAL is parameterized by our dynamic parameters $W_i^l$, where $l$ denotes the layer index of the linear transformation, and ReLU activation is included between two layers. Bias parameters are omitted for brevity.

### 3.3. Overall architecture

Fig. 3 illustrates the overall pipeline of the proposed video inpainting network **SAVIT**. The network is composed of encoder, decoder, and multiple transformer blocks between the encoder and the decoder. First, the encoder produces output features by taking $N$ consecutive video frames with masks and the corresponding semantic segmentation maps as input. We employ the encoder in FuseFormer [26] and additionally concatenate corresponding semantic maps to input. Next, we carry out the tokenization procedure [26], then forward to several transformer blocks to extract powerful features. Finally, the tokenized features are rearranged into the image-like shape, which is then fed into the decoder that predicts $N$ inpainted video frames as previous works [26, 23].

Note that our semantic-aware mixture-of-experts schemes can be plugged into any conventional transformer-based approach (*e.g*., FuseFormer [26], E2FGVI [23]), where a conventional transformer block is just replaced with our semantic-aware transformer block. In our experiments, we embed our dynamic transformer within FuseFormer [26]. Specifically, SAVIT contains eight FuseFormer blocks, each of which consists of a multi-head self-attention (MHSA) operation and fusion feed-forward network (F3N). We replace one of the eight FuseFormer blocks with our dynamic transformer block. Precisely, within the F3N in the fourth FuseFormer block, we modify the two standard linear layers with our dynamic ones and add a semantic router.

**Semantic-aware dynamic discriminator.** To facilitate each expert's learning of the corresponding category's representation and texture details, we employ a semantic-aware discriminator (**SAD**) that predicts the real/fake score for each token. Fig. 4 illustrates the architecture of SAD, which has similar structures to the encoder of SAVIT, except for the last layer, where a single SAL is added. Notably, we utilize the same mixture coefficients for dynamic parameters for both SAVIT (generator) and SAD (discriminator), encouraging each expert to learn detailed and discriminative category-specific representation.

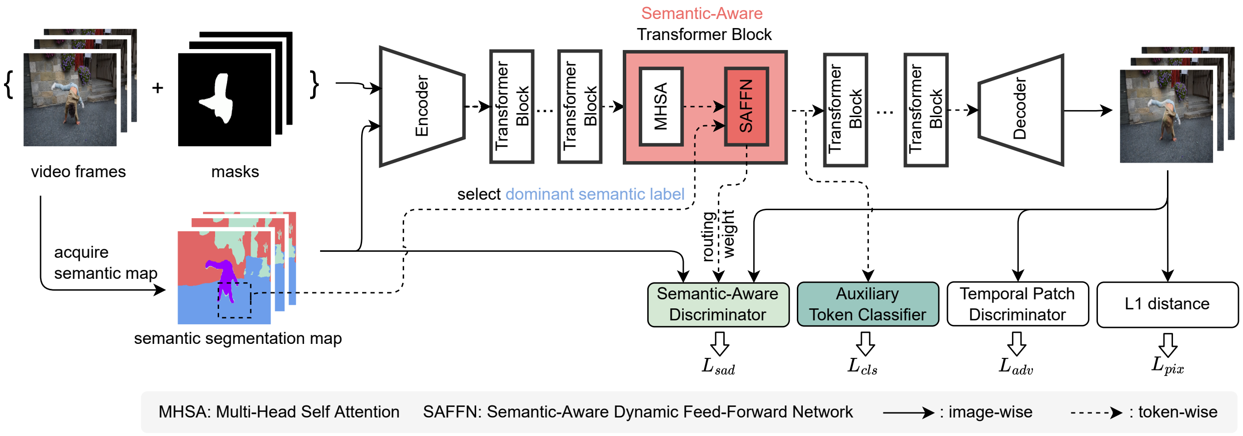


Figure 3: Overall flow of **SAVIT**. Our method utilizes additional semantic cues with a mixture-of-experts scheme. The transformer parameters are dynamically adapted to the local semantic information for each token region. With these token-wise dynamic parameters, our network produces more precise inpainting results, generating an object's boundary or synthesizing detailed textures.

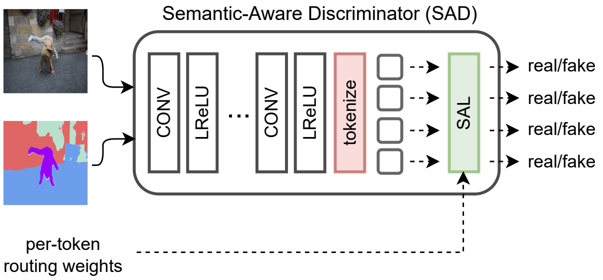


Figure 4: Schematic illustration of **SAD**. With the specific mixture-of-experts parameters produced by using the same routing weight from the generator, SAD discriminates whether the inpainted results are realistic or not.

### 3.4. Loss function

**Auxiliary token classification loss.** In the mixture-of-experts scheme, it is crucial that each expert is assigned with distinct role [18, 30]. If our semantic experts have been assigned properly to each corresponding category, the output tokens from the semantic-aware dynamic block should be helpful for classification. We drive collaborative role division across our semantic experts. Specifically, we regularize the learning of experts during training by performing auxiliary classification tasks on the output tokens from the semantic-aware dynamic block. Such token-wise classification is performed by an auxiliary single linear layer classifier. The classifier predicts the semantic labels of the output tokens, as illustrated with the green box in Fig. 3. As a result, each expert can effectively learn the feature representation corresponding to a specific semantic label.

**Overall objective function.** Our overall objective function $\mathcal{L}_{total}$ is the weighted sum of the following four different loss terms:

$$\mathcal{L}_{total} = \mathcal{L}_{pix} + \lambda_{adv}\mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls} + \lambda_{sad}\mathcal{L}_{sad}, \quad (5)$$

where $L_{pix}$ is a pixel-wise reconstruction loss, $\mathcal{L}_{adv}$ is an adversarial loss, $\mathcal{L}_{cls}$ is token classification loss, and $\mathcal{L}_{sad}$ is semantic adversarial loss, while $\{\lambda_{adv}, \lambda_{cls}, \lambda_{sad}\}$ control the weight of associated loss term. Given the predicted video frame $\hat{Y}$ and its ground truth counterpart $Y$, we employ the L1 pixel-wise reconstruction loss as:

$$\mathcal{L}_{pix} = ||\hat{Y} - Y||_1. \quad (6)$$

As for the adversarial loss, we train our inpainting network to fool a discriminator $D$ as follows:

$$\mathcal{L}_{adv} = -E_{\hat{Y}}[D(\hat{Y})]. \quad (7)$$

Based on a temporal patch discriminator [26, 23], the discriminator $D$ is trained to optimize

$$\mathcal{L}_D = E_Y[\text{ReLU}(1 - D(Y))] + E_{\hat{Y}}[\text{ReLU}(1 + D(\hat{Y}))]. \quad (8)$$

The token classification loss term is given as

$$\mathcal{L}_{cls} = -\sum_i \sum_{c=1}^{C} y_i^c \log(softmax(\hat{y}_i)), \quad (9)$$

where $\hat{y}_i \in \mathbb{R}^C$ is the predicted score from the single-layer classifier conditioned on the output tokens from the semantic-aware dynamic block, while $y_i^c$ is the target probability for class $c$ at the $i$-th token. For the ground truth target $y_i^c$, we use the input semantic segmentation map (*i.e.*, $y_i^c = s_i(c)$). The semantic-aware adversarial loss $\mathcal{L}_{sad}$ is similar to $\mathcal{L}_{adv}$ in that (8) and (7) are used, but with the semantic-aware discriminator (SAD) instead of $D$.

## 4. Experiments

In this section, we quantitatively and qualitatively demonstrate the effectiveness of SAVIT. Please refer to the supplementary material for more experimental results.

|  | YouTube-VOS [36] |  |  |  | DAVIS [28] |  |  |  |
|---|---|---|---|---|---|---|---|---|
| Method | PSNR$^{\uparrow}$ | SSIM$^{\uparrow}$ | VFID$_{\downarrow}$ | $E_{warp}(\%)_{\downarrow}$ | PSNR$^{\uparrow}$ | SSIM$^{\uparrow}$ | VFID$_{\downarrow}$ | $E_{warp}(\%)_{\downarrow}$ |
| VINet [15] | 29.20 | 0.9434 | 0.072 | 0.1490 | 28.96 | 0.9411 | 0.199 | 0.1785 |
| LGTSM [3] | 29.74 | 0.9504 | 0.070 | 0.1859 | 28.57 | 0.9409 | 0.170 | 0.1640 |
| CAP [21] | 31.58 | 0.9607 | 0.071 | 0.1470 | 30.28 | 0.9521 | 0.182 | 0.1533 |
| STTN [39] | 32.34 | 0.9655 | 0.053 | 0.0907 | 30.67 | 0.9560 | 0.149 | 0.1449 |
| E2FGVI [23] | 33.71 | 0.9700 | 0.046 | 0.0864 | 33.01 | 0.9721 | 0.116 | 0.1315 |
| ISVI [41] | 31.19 | 0.9569 | 0.053 | 0.1173 | 32.28 | 0.9669 | 0.129 | 0.1589 |
| FuseFormer [26] | 33.16 | 0.9673 | 0.051 | 0.0900 | 32.54 | 0.9700 | 0.138 | 0.1362 |
| **SAVIT** (Ours) | **33.97** | **0.9727** | **0.043** | **0.0436** | **33.14** | **0.9748** | **0.107** | **0.0673** |

Table 1: Quantitative comparison of SAVIT against state-of-the-art methods. SAVIT outperforms the existing approaches in terms of various evaluation metrics. Notably, SAVIT significantly elevates the inpainting performance of FuseFormer [26], which is our baseline architecture, witnessing the superiority of leveraging semantic cues within the input video.

|  | Configuration | PSNR$^{\uparrow}$ | VFID$_{\downarrow}$ |
|---|---|---|---|
| A | FuseFormer$_{small}$ [26] | 30.51 | 0.182 |
| B | + super-categorized segmentation map | 30.90 | 0.161 |
| C | + SAL | 30.92 | 0.164 |
| D | + token classification loss ($\lambda_{cls}$) | **31.19** | 0.165 |
| E | + SAD | 31.01 | **0.157** |

Table 2: Ablation experiments on various network configurations under DAVIS [28] dataset. Starting from the performance of the baseline architecture [26], the inpainting performance gradually increases as the configurations are added since the network can benefit from the semantic cues within a video.

### 4.1. Implementation details

**Datasets and evaluation metrics.** We use two conventional video inpainting datasets, YouTube-VOS [36] and DAVIS [28], for our experiments. YouTube-VOS dataset consists of 3471 training, 474 validation, and 508 test videos from mixed scenes. Meanwhile, DAVIS dataset comprises 150 video clips, 60 of which are used for training and the remaining 90 clips are used for evaluation. Following previous works [26, 23], we use only the training videos in the YouTube-VOS for training, while the evaluation is conducted on YouTube-VOS testset and 50 selected clips of DAVIS testset. During training, we create random mask shapes for input masked frames as in [39, 26, 23] and use the method discussed in Section 3.1 to obtain a segmentation map for the entire dataset.

The inpainting performance is evaluated under two standard scenarios: fixed region inpainting and foreground object removal. For the fixed region inpainting, we use free-form fixed masks for each video provided by FuseFormer [26]. As for the foreground object removal, we remove the main objects using frame-by-frame annotations provided by the stakeholders. Notably, we employ a simple k-nearest neighbors algorithm to complete the missing object region in the segmentation maps for the object removal task (please refer to supplementary material for detail).

We compare the results with respect to four evaluation metrics: PSNR, SSIM, temporal warping error $E_{warp}$ [20], and Video-based Fréchet Inception Distance (VFID) [35]. PSNR and SSIM are traditional metrics used to measure the average quality of produced frames. VFID and $E_{warp}$ measure the video's perceptual quality and temporal stability (consistency), respectively, and low values indicate better quality.

**Training settings.** We follow the training settings of our baseline network FuseFormer [26]. Specifically, we randomly sample five frames from a video, resize the frames to 240×432, and utilize random horizontal flipping augmentation. The network is trained for 500k iterations with Adam optimizer [16] and a batch size of eight. The initial learning rate is 1e-4 and decreases to 1e-5 after 400k iterations. The coefficients for our loss functions, $\lambda_{adv}$, $\lambda_{cls}$, and $\lambda_{sad}$ are 0.01, 0.01, and 0.001, respectively. We implement our network on the PyTorch framework and use eight NVIDIA RTX A6000 GPUs for training.

### 4.2. Video inpainting results

**Quantitative results.** Table 1 quantitatively compares our video inpainting results on fixed region inpainting setting against seven state-of-the-art video inpainting networks: VINet [15], LGTSM [3], CAP [21], STTN [39], E2FVGI [23], ISVI [41], and FuseFormer [26]. The result shows that our method remarkably elevates the performance of FuseFormer [26], on which we base our architecture, and achieves superior performances on both YouTube-VOS and DAVIS datasets compared to the state-of-the-art methods. Notably, the warping error has also significantly improved, indicating that semantic information benefits in generating temporally consistent video. These results show the effectiveness of employing mixture-of-experts to dynamically leverage the semantic cues within given frames for video inpainting.

**Qualitative results.** Fig. 5 shows that SAVIT has visually pleasing inpainting results, especially in synthesizing semantic structure, boundary, and texture of objects. Notably, unlike the baseline methods that rely solely on inherent video clues and cannot accurately restore the shape of the human body, SAVIT effectively completes the region

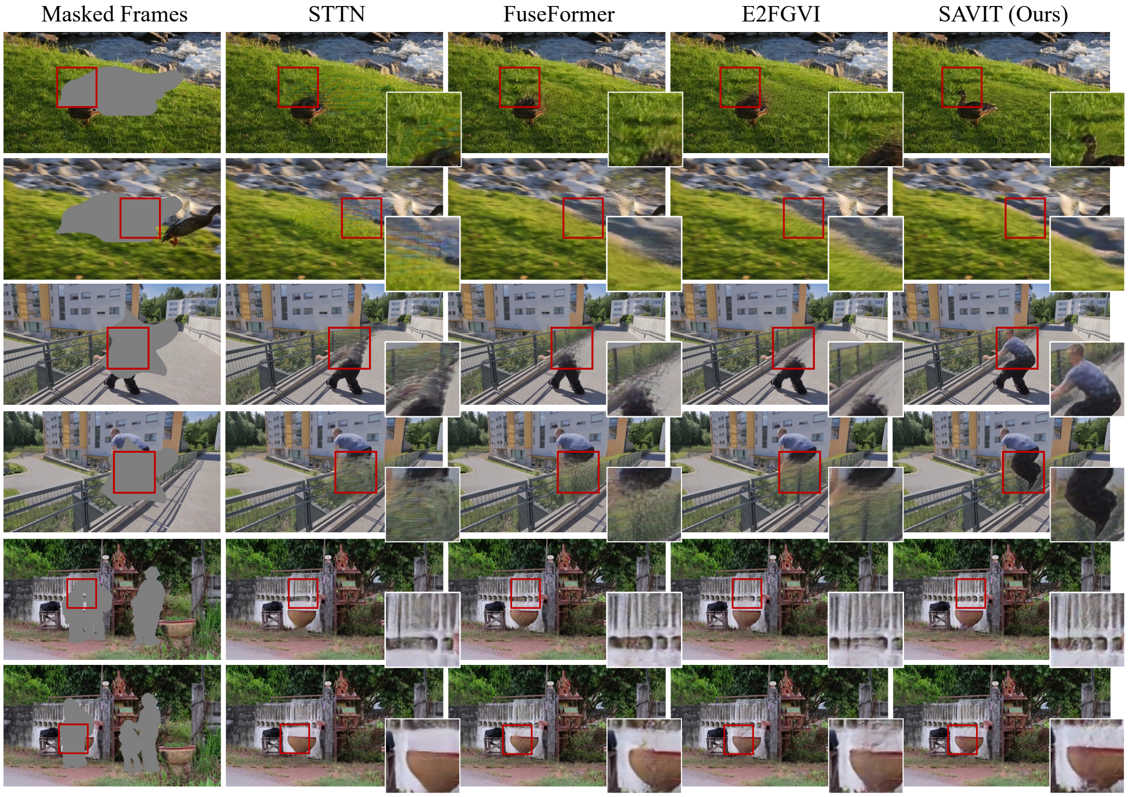


Figure 5: Qualitative comparison of SAVIT with baseline video inpainting networks in fixed region inpainting and object removal. SAVIT more accurately recovers the object's boundaries and appropriate textures.

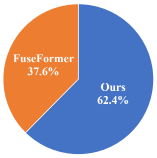


Figure 6: User study result.

| Configuration | PSNR↑ |
|---|---|
| hardmax (SAVIT) | 31.01 |
| softmax | 30.91 |
| random routing weight | 19.86 |

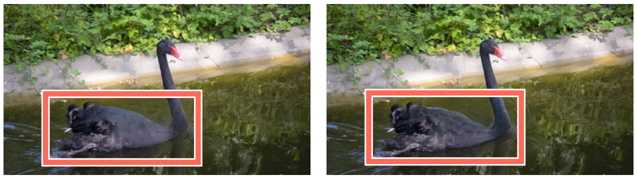
Table 3: Effect of the routing weight on the DAVIS [28] dataset.

by leveraging additional semantic information (third and fourth rows). Furthermore, SAVIT demonstrates the capability of generating more semantically plausible textures, as observed from the detailed textures of synthesized grass and wall (second and fifth rows).

**User study.** We conduct a user study on the foreground object removal task, where ground truth target video frames for evaluation are not available, to investigate the real-world applicability of our method using DAVIS [28] testset. We ask 24 raters to choose a better video between two randomly ordered videos, including the inpainted results by FuseFormer [26] and ours, fifteen times per rater. The result presented in Fig. 6 shows that raters prefer ours to the baseline, indicating the impact of exploiting semantic cues in completing missing object regions.

## 5. Ablation studies

We perform ablation experiments based on a smaller FuseFormer network (FuseFormer$_{small}$) by halving the dimension of the hidden layer.



SAVIT w/o SAD

SAVIT w/ SAD

Figure 7: Effect of semantic-aware discriminator (SAD). The texture of the black swan's body is blurry in the case without SAD.

**Network configuration.** Table 2 compares the inpainting performance for various configurations of SAVIT. First, we observe that the baseline network's performance improves with additional segmentation map inputs (rows A and B). However, a subtle additional performance improvement (0.02dB PSNR gain, rows B and C) is achieved when we naïvely add our proposed dynamic linear layer to the network. Rather, applying the proposed token-wise classification loss to the objective boosts the performance (0.2dB PSNR gain, row D). Overall performance improvement by applying mixture-of-experts (rows C and D) highlights the importance and effectiveness of employing mixture-of-experts to handle semantics dynamically. Moreover, using a semantic-aware dynamic discriminator (row E) degrades PSNR while improving VFID. However, as shown in Fig. 7, utilizing the semantic-aware discriminator produces more visually plausible results, which solidifies the advantage of

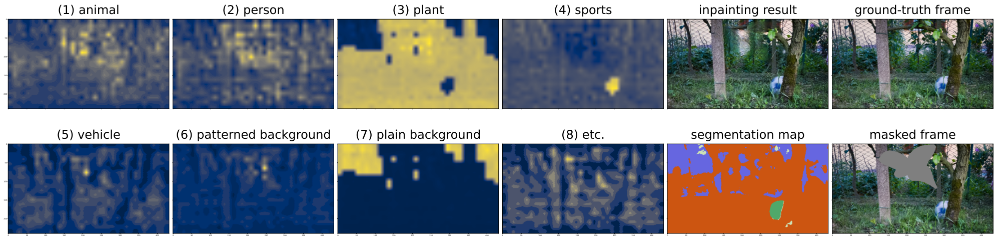


Figure 8: Visualization of the semantic router results. (1) to (8) are the visualization of routing weights of the corresponding super-category. The routing weights are rearranged to the image-like features. Brighter pixel values indicate higher routing weights.

| # Experts | PSNR$^{\uparrow}$ | VFID$_{\downarrow}$ |
|---|---|---|
| 4 | 30.83 | 0.165 |
| 6 | 30.90 | 0.168 |
| 8 | 31.01 | 0.157 |

Table 4: Inpainting results by changing the numbers of experts under the DAVIS [28] dataset. More experts render better inpainting results.

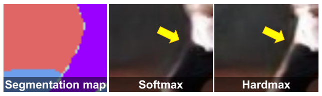


Figure 9: Qualitative comparison on different routing weight strategies.

| # blocks | # params. | PSNR$^{\uparrow}$ |
|---|---|---|
| 1 | 26.0M | 30.40 |
| 2 | 33.1M | 30.45 |

Table 5: Ablation on different number of semantic blocks.

| Method | Mask ratio | PSNR$^{\uparrow}$ | VFID$_{\downarrow}$ |
|---|---|---|---|
| FuseFormer$_{small}$ | 30% | 22.57 | 0.554 |
| | 60% | 18.10 | 1.059 |
| FuseFormer$_{small}$ + Ours | 30% | 23.57 | 0.412 |
| | 60% | 19.21 | 0.750 |

Table 6: Comparison on varying mask ratio.

the discriminator in improving perceptual quality.

**Visualization of routing weights.** Fig. 8 visualizes the routing weights of each expert from the trained semantic router. We rearrange the tokens to image-like features for visualization, and brighter values indicate higher routing weights in the corresponding region. The bright regions among different experts indicate that the experts collaborate during the inpainting procedure rather than simply utilizing a specific expert. We see that routing weights tend to change locally smoothly while the input segmentation map changes abruptly.

**Designing choice on semantic router.** Table 3 compares different design choices of our semantic router. The performance drop by replacing the trained routing weights (*i.e.*, mixture coefficients) with random weights indicates the discriminative role of trained experts. Moreover, the results of softmax routing strategy show a noticeable performance degradation against hardmax strategy quantitatively. Fig. 9 also shows that softmax results tend to have blurry texture, especially on the edge area. Therefore, Hardmax was chosen for our final model to facilitate role division between experts better, as it enables the learning of distinct parameters by dividing the input for each expert before the router blends the token based on its features.

**Impact of the number of experts.** Table 4 compares the inpainting performance by changing the number of experts. Notably, as the number of experts (*i.e.*, semantic categories) increases, the network can benefit from more diverse semantic clues in terms of PSNR and VFID values, and we set the number of experts for our final model to eight.

**Impact of the number of semantic blocks.** Table 5 compares the inpainting performance by applying our semantic module on a different number of blocks. The result indicates that using more semantic blocks produces a slightly better inpainting performance. Although we apply our semantic module one time due to hardware limits for our final model, it is expected that the results will be naturally improved by increasing the number of semantic blocks.

**Input video with large corruption.** Table 6 compares our method against FuseFormer$_{small}$ under different corrupted region ratio settings. It is shown that, despite larger masks, our method still outperforms the baseline by a large margin by exploiting the semantic information.

## 6. Conclusion and Future Works

In this study, we have explored leveraging semantic information within a given video for transformer-based video inpainting. Accordingly, we propose an effective semantic-aware dynamic transformer with the notion of mixture-of-experts. Our dynamic transformer contains class-specific expert parameters where experts are efficiently combined to perform token-wise dynamic linear operations based on the token feature and its semantic information. The experiments demonstrate the effectiveness of our method, especially in recovering semantic structures. Although we have utilized pseudo labels for our study, we believe more accurate and denser annotated datasets can improve the inpainting performance further since our method relies on the quality of the segmentation label.

## Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2023-00222776) and Samsung Electronics Co., Ltd, and Samsung Research Funding Center of Samsung Electronics under Project Number SRFCIT1901-06.

## References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[2] Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[3] Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Learnable gated temporal shift module for deep video inpainting. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2019. 2, 6

[4] Cheng Chen, Jiayin Cai, Yao Hu, Xu Tang, Xinggang Wang, Chun Yuan, Xiang Bai, and Song Bai. Deep interactive video inpainting: An invisibility cloak for harry potter. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2021. 2

[5] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[7] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[9] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[10] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 1991. 2

[11] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016. 2

[12] Jun-Gyu Jin, Jaehyun Bae, Han-gyul Baek, and Sang-hyo Park. Object-ratio-preserving video retargeting framework based on segmentation and inpainting. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2023. 2

[13] Jaeyeon Kang, Seoung Wug Oh, and Seon Joo Kim. Error compensation framework for flow-guided video inpainting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[14] Bholeshwar Khurana, Soumya Ranjan Dash, Abhishek Bhatia, Aniruddha Mahapatra, Hrituraj Singh, and Kuldeep Kulkarni. Semie: Semantically-aware image extrapolation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[15] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 6

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[17] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[18] Xiangtao Kong, Hengyuan Zhao, Yu Qiao, and Chao Dong. Classsr: A general framework to accelerate super-resolution networks by data characteristic. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5

[19] Avisek Lahiri, Arnav Kumar Jain, Sanskar Agrawal, Pabitra Mitra, and Prabir Kumar Biswas. Prior guided gan based semantic inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[20] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 6

[21] Sungho Lee, Seoung Wug Oh, DaeYeun Won, and Seon Joo Kim. Copy-and-paste networks for deep video inpainting. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 6

[22] Jiacheng Li, Chang Chen, and Zhiwei Xiong. Contextual outpainting with object-level contrastive learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[23] Zhen Li, Cheng-Ze Lu, Jianhua Qin, Chun-Le Guo, and Ming-Ming Cheng. Towards an end-to-end framework for flow-guided video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 4, 5, 6

[24] Liang Liao, Jing Xiao, Zheng Wang, Chia-Wen Lin, and Shin'ichi Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 2

[25] Liang Liao, Jing Xiao, Zheng Wang, Chia-Wen Lin, and Shin'ichi Satoh. Image inpainting guided by coherence priors of semantics and textures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2

[26] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in

transformers for video inpainting. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 4, 5, 6, 7

[27] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[28] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 6, 7, 8

[29] Jingjing Ren, Qingqing Zheng, Yuanyuan Zhao, Xuemiao Xu, and Chen Li. Dlformer: Discrete latent transformer for video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[30] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 1, 2, 5

[31] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 1, 2

[32] Yuhang Song, Chao Yang, Yeji Shen, Peng Wang, Qin Huang, and C-C Jay Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 1, 2

[33] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[34] Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2019. 1, 2

[35] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 6

[36] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 2, 3, 6

[37] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[38] Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 2

[39] Yanhong Zeng, Jianlong Fu, and Hongyang Chao. Learning joint spatial-temporal transformations for video inpainting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 6

[40] Kaidong Zhang, Jingjing Fu, and Dong Liu. Flow-guided transformer for video inpainting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2

[41] Kaidong Zhang, Jingjing Fu, and Dong Liu. Inertia-guided flow completion and style fusion for video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 6

[42] Jingkai Zhou, Varun Jampani, Zhixiong Pi, Qiong Liu, and Ming-Hsuan Yang. Decoupled dynamic filter networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[43] Xueyan Zou, Linjie Yang, Ding Liu, and Yong Jae Lee. Progressive temporal feature alignment network for video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1