This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

# Frequency-Aware Spatiotemporal Transformers for Video Inpainting Detection

Bingyao Yu[1,3], Wanhua Li[1,2], Xiu Li[1,3], Jiwen Lu[1,2,*], Jie Zhou[1,2]
[1]Department of Automation, Tsinghua University, China
[2]Beijing National Research Center for Information Science and Technology, China
[3]Shenzhen International Graduate School, Tsinghua University, China
{yby18,li-wh17}@mails.tsinghua.edu.cn; li.xiu@sz.tsinghua.edu.cn; {lujiwen,jzhou}@tsinghua.edu.cn

## Abstract

*In this paper, we propose a Frequency-Aware Spatiotemporal Transformer (FAST) for video inpainting detection, which aims to simultaneously mine the traces of video inpainting from spatial, temporal, and frequency domains. Unlike existing deep video inpainting detection methods that usually rely on hand-designed attention modules and memory mechanism, our proposed FAST have innate global self-attention mechanisms to capture the long-range relations. While existing video inpainting methods usually exploit the spatial and temporal connections in a video, our method employs a spatiotemporal transformer framework to detect the spatial connections between patches and temporal dependency between frames. As the inpainted videos usually lack high frequency details, our proposed FAST synchronously exploits the frequency domain information with a specifically designed decoder. Extensive experimental results demonstrate that our approach achieves very competitive performance and generalizes well.*

## 1. Introduction

Video inpainting has attracted much attention over the past years [18, 17, 35, 23, 6], which is a task of repairing the missing or corrupter regions in a video sequence with visually plausible pixels. Video inpainting has been widely used as a video editing technique in multiple applications such as video completion and virtual reality. However, the increasing progress and rapid development of video inpainting also result in enticing malicious attackers to forge video sequences to release some fake news, aiming to mislead the direction of public opinion. Recently, as a consequence of advances in deep learning, a variety of studies [15, 24] have shown spectacular progress in video inpainting, which enables editing the special area of a video, e.g. removing objects that could be key evidence. Inpainted videos have become more and more difficult to be distinguished even by eyes in pace with the remarkable success in video inpainting methods. Furthermore, the misuse of video inpainting techniques may pose potential threats and cause legal issues in society. Therefore, there is a justified demand for effective video inpainting detection methods, which attempt to detect whether the videos presented are pristine or inpainted.

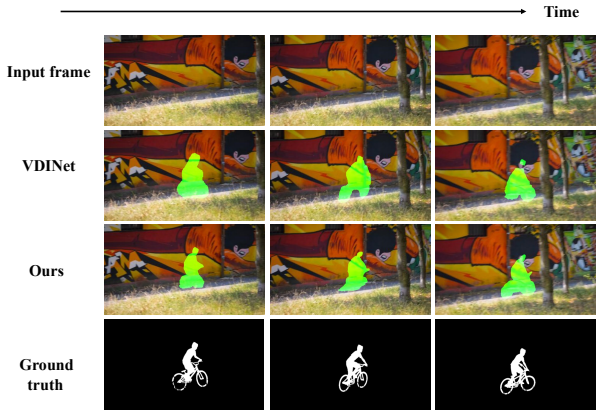



Figure 1. Compared with the first video inpainting detection method VDINet [39], we can observe that our FAST approach can preserve more detailed information of prediction masks when guaranteeing the temporal consistency.

There have been a variety of studies about video inpainting [2, 15, 23, 17], which fall into two main classes: patch-based methods and learning-based methods. However, there is an important problem with the above two methods – the video inpainting turns out to obtain corresponding pixels from similar regions or frames, or learn related distributions from similar scenes. Therefore, these methods inevitably leave clues and artifacts such as the inconsistency between regional pixels, sharp change at the edge of the regions and the blurred area caused by the failure to acquire complete distribution. Accordingly, more recent approaches have been developed for inpainting detection, but most existing inpainting detection methods are frame-level based on the single input image. Further, [39] first proposed the LSTM-

* Corresponding author

based framework combining both RGB image and ELA information to extract both spatial and temporal features for video inpainting detection. In general, there still leaves a clear gap in the experimental performance.

In this paper, we propose to learn frequency-aware spatiotemporal transformers for video inpainting detection. Actually, one of the most important things in video inpainting detection is to discover the associations between patches and frames. The exiting methods usually employ attention models and memory mechanism, where the direct and hard combination will lead to inconsistent prediction results. Motivated by this, we construct the spatiotemporal transformer including encoder and decoder to capture the spatial and temporal artifacts using multihead self-attention mechanism. Furthermore, we incorporate the frequency-aware features as the auxiliary supervised information into the prediction process so that the upsampling operation for prediction masks is regularized to improve the generalizability. Afterwards, we optimize the FAST framework guided by the hybrid loss function which is directly related to the evaluation metric and diminish the effect of class imbalance existing in the datasets. Finally, we evaluated our framework both in-domain and cross-domain to investigate both the performance and generalization.

To summarize, the contributions of our paper are threefold: (i) We first introduce the transformer-based framework for video inpainting detection which can explore the spatial and temporal information inside the inpainted videos. (ii) We present the frequency-aware features and augment the extracted features with frequency domain information to find tamper artifacts and manipulation clues well-hidden in the RGB frames. (iii) Experimental results on the Davis Video Inpainting dataset and Free-form Video Inpainting dataset show that our proposed framework achieves very competitive performance even when confronted with unseen approaches.

## 2. Related Work

**Video Inpainting Detection:** Existing video inpainting methods can be primarily divided into two classes: patch-based methods and learning-based methods. Patch-based methods aim to exploit the connection and similarity between patches and frames. For example, Barnes *et al*. [2] proposed to search for approximate nearest-neighbor patches in the surrounding region recurrently to complete the masked region. To handle dynamic scenes, Huang *et al*. [15] adopted a non-parametric optimization-based method to match patches and jointly utilize optical flow and color as regularization. For the second class, learning-based methods aim to utilize the deep network to learn semantic representations. Recently, Kim *et al*. [17] utilized the flow loss and warping loss as additional constraints for inpainting the missing regions. To attend to the invisible information from the reference images, Oh *et al*. [23] proposed an asymmetric attention block to compute similarities in a non-local manner. Lee *et al*. [18] intended to copy and paste reference frames to complete missing details.

At the same time, there have been several methods developed for the forensics of inpainting approaches. To reduce the high false alarm rates, Chang *et al*. [5] adopted a two-stage searching approach to search for the suspicious regions and corresponding multi-region relations. Afterwards, Zhu *et al*. [40] adopted CNNs to detect inpainting patches within $256 \times 256$ images. Recently, Li *et al*. [19] employed high-pass pre-filtering as the initialization of CNNs to distinguish high frequency residual of real images from inpainted ones. To improve the generalization and robustness, Zhou *et al*. [39] combined both RGB image and ELA [31] information with Convolutional LSTM to guarantee temporally consistent prediction. However, since such approaches can not formulate consistent attention results along both spatial and temporal dimensions, there still is a clear gap in the experimental performance.

**Tranformers:** Transformers have been successfully applied in natural language processing and machine translation [29, 8, 10, 34, 36]. Due to the core self-attention mechanism of transformers, researchers tend to utilize transformers to model long-range dependencies [3, 14, 32, 30, 27]. Recently, transformers begin to achieve a series of breakthroughs in computer vision tasks. DETR [4] utilized a transformer encoder-decoder architecture for object detection. In [11], transformers were directly applied to sequences of image patches embedding to conduct image classification, which achieved excellent performance compared to state-of-the-art convolutional networks. Furthermore, SETR [37] rethought semantic segmentation from a sequence-to-sequence perspective with transformers. Moreover, IPT [7] developed a new pre-trained transformer model from low-level computer vision tasks. Meanwhile, transformers have also draw significantly growing attention in video processing. VisTR [33] viewed the video instance segmentation task as a direct end-to-end sequence decoding problem and accomplished sequence instance segmentation with transformers. The above studies reveal the effectiveness of transformers in computer vision tasks. However, to the best of our knowledge, so far there exist no previous applications of transformers to video inpainting detection. Motivated by the fact that the transformers can both model long-range dependencies to learn temporal information across multiple frames and utilize self-attention mechanisms to explore spatial features between patches. Thus, we propose the FAST method for video inpainting detection.

## 3. Proposed Approach

In this section, we detail the frequency-aware spatiotemporal transformers for video inpainting detection. We first

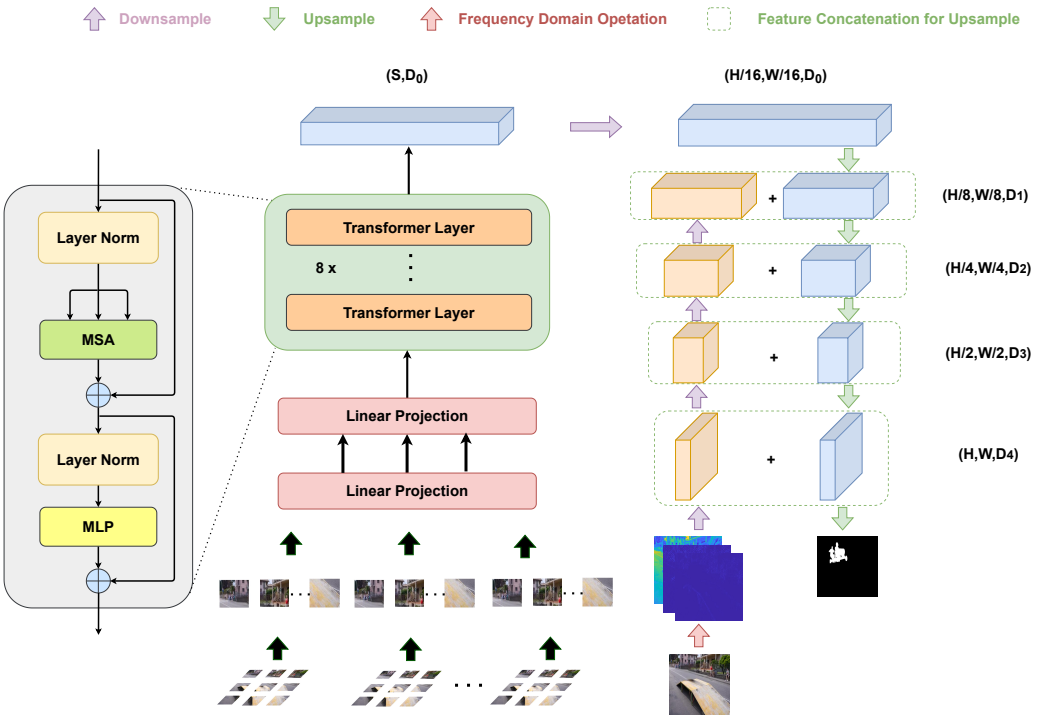



Figure 2. The overall network architecture of our FAST framework. For a series of video frames, we firstly adopt two linear projection layers to map the image patches to vector embeddings along both spatial and temporal dimensions. Then we utilize transformer encoders to obtain the hidden feature consist of spatial and temporal information. Meanwhile, we incorporate the frequency-aware features as an auxiliary signal to assist the encoder to predict video inpainting detection results.

show how to construct the spatiotemporal transformer including encoder and decoder. Then, we present the motivation of frequency-aware features and propose the combination of RGB frames and spectrum signals. Finally, we introduce how to optimize our FAST framework guided by the hybrid loss function. Figure 2 shows the overall network architecture of our proposed approach.

### 3.1. Spatiotemporal Transformer Networks

**Images to Sequence:** Following the typical transformer encoder-decoder architecture, we first divide the input image $\mathbf{I}$ into $N$ patches, where $N = \frac{H}{S} \times \frac{W}{S}$ (*i.e.*, the input sequence length) and the patch size is $S$. Additionally, the patch size $S$ is usually set to 16. Then, we obtain a sequence of flattened 2D patches $\{\mathbf{I}_i \in \mathbb{R}^{S^2 \cdot C} | i = 1, .., N\}$, where $C$ denotes the image channels. As for video inpainting detection task, we normally input the video clips to the network so that there are several input images. We choose $T + 1$ frames, $\frac{T}{2}$ frames are in front of the $(\frac{T}{2} + 1)$-th input image, and the rest are behind. Thus, we can perform image division for all the frames as above and the $t$-th flattened 2D patches are $\{\mathbf{I}_i^t \in \mathbb{R}^{S^2 \cdot C} | i = 1, .., N, t = 1, ..., T + 1\}$.

We first utilize a trainable linear projection to map the vectorized patches $\mathbf{I}_i^t$ into a latent $D_0$-dimensional embedding space along the spatial dimension. Then we repeat the similar operation along the temporal dimension. To encode the patch spatial and temporal information, we adopt learnable position embeddings which are directly added to the above patch embeddings to preserve positional information as follows:

$$\mathbf{z}_0 = \mathbf{E}(\mathbf{I}_i^t)\mathbf{E_0} + \mathbf{E}_{pos}, \tag{1}$$

where $\mathbf{E_0} \in \mathbb{R}^{(S^2 \cdot C) \times D_0}$ denotes the spatial patch embedding projection, $\mathbf{E} \in \mathbb{R}^{(N \cdot (T+1) N)}$ denotes the temporal patch embedding projection and $\mathbf{E}_{pos} \in \mathbb{R}^{N \times D}$ is the position embedding.

**Transformer Encoder:** There exist $L$ layers of Multi-head Self-Attention (MSA) module and Multi-Layer Perceptron (MLP) blocks in the transformer encoder. Consequently, the output of the $\ell$-th layer can be formulated as follows:

$$\hat{\mathbf{z}}_\ell = MSA(LN(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \tag{2}$$

$$\mathbf{z}_\ell = MLP(LN(\hat{\mathbf{z}}_\ell)) + \hat{\mathbf{z}}_\ell, \tag{3}$$

where $LN(\cdot)$ is the layer normalization operator, $\hat{\mathbf{z}}_\ell$ is the intermediate output variable of the MSA module, $\mathbf{z}_{l-1}$ and $\mathbf{z}_l$ denotes encoded image representations served as the input and output. We show the structure of a transformer layer in Figure 2.

**Decoder Designs:** Follow SETR [37], for the decoder part, we also adopt the simple progressive upsampling

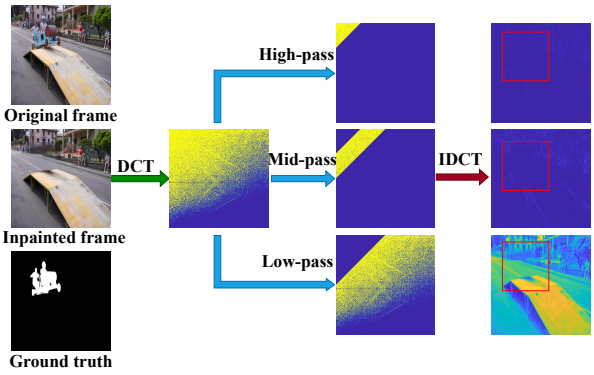



Figure 3. The process to obtain our proposed frequency-aware features. We can observe that the decomposed frequency-aware images reveal the inpainting artifacts having different frequency domain distribution with the untouched regions (best viewed digitally, in color and with zoom).

(PUP) approach. Moreover, we first reshape the final encoder output $\mathbf{z}^l$ back to $\mathbf{x}^0 \in \mathbb{R}^{\frac{H}{16}\times\frac{W}{16}\times D_0}$, which denotes a 2D feature map with $\frac{H}{16} \times \frac{W}{16}$ size and $D_0$ channels. Afterwards, we utilize three sequential standard upsampling-convolution layers to increase the feature map resolution, where we obtain $\mathbf{x}^1 \in \mathbb{R}^{\frac{H}{8}\times\frac{W}{8}\times D_1}$, $\mathbf{x}^2 \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times D_2}$ and $\mathbf{x}^3 \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times D_3}$, respectively. These feature maps of various scales $\mathbf{x}^0$, $\mathbf{x}^1$ and $\mathbf{x}^2$ are reshaped to the same size for late combination with the frequency-aware features. We show the structure of the decoder in Figure 2.

### 3.2. Frequency-aware Features

Recently, the video inpainting approaches usually inpaint the specific regions with their surrounding patches to appear realistic. As a result, it is difficult to acquire a mapping directly from the inpainted RGB frames to corresponding binary ground truth masks. Therefore, a variety of studies tend to learn multimodal features for general video inpainting detection. For example, [39] proposes to combine RGB features with error level analysis (ELA) information [31] which is designed to disclose regions containing inconsistent compression artifacts. Besides, several attempts have been made for image forgery detection using frequency domain clues [13]. To mitigate the limits of RGB frames, we augment the extracted features with frequency domain information.

Normally, the researchers would utilize DFT or DCT to transfer an image to frequency domain, and here we choose the Discrete Cosine Transform (DCT) [1] considering its wide applications in computer vision tasks and regular distribution of spectrum. Afterwards, as for an input image $\mathbf{I}$, we can get the frequency domain map $M$: $M = \mathcal{T}(\mathbf{I})$, where $\mathcal{T}$ denotes the DCT. To obtain more detailed and meticulous frequency domain information, we employ $n$ frequency band-pass filter $\{f_1, f_2, ..., f_n\}$ to decompose frequency domain map into a series of independent parts. Subsequently, we design $f_i$ as a binary map so that we can achieve the decomposed spectrum by doing dot-product of $f_i$ and $M$. Actually, the hand-crafted transformations DCT fails to handle the shift-invariance and explore local consistency which are very significant in the inpainted images. Finally, we can obtain the decomposed frequency-aware images as follow:

$$\mathbf{I}_i = \mathcal{T}^{-1}(\mathcal{T}(\mathbf{I}) \odot f_i) \tag{4}$$

where $\mathbf{I}_i$ is the $i$-th decomposed frequency-aware image and $i = \{1, 2, ..., n\}$. $\mathcal{T}^{-1}$ denotes the Inversed Discrete Cosine Transform (IDCT) and $\odot$ is the element-wise product.

We choose to set $n$ to 3 and there are two reasons for this: on the one hand, we will stack these components later along the channel axis to keep consistent with the input RGB frames. On the other hand, we can decompose frequency domain information into common high-pass, mid-pass and low-pass signals which are regularly distributed in the spectrum. Similar to [26], from low frequency to high frequency, we split the spectrum into 3 bands following the equal energy principle. After we inversely transformed the decomposed frequency components to the spatial domain, we finally obtain a series of decomposed frequency-aware images. Then, we stack these components along the channel axis and input the stacked feature map into a CNN backbone. The feature map is progressively downsampled to $\frac{H}{8} \times \frac{W}{8}$ and we utilize the CNN to explore enriched discriminative information. We take three outputs from first ($x_1 \in \mathbb{R}^{\frac{H}{2}\times\frac{W}{2}\times C_1}$), second ($x_2 \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times C_2}$) and third ($x_3 \in \mathbb{R}^{\frac{H}{8}\times\frac{W}{8}\times C_3}$) blocks to match with the output from transformer decoder.

### 3.3. Loss Function

As for video inpainting methods, the user tends to remove some objects or repair local missing regions. Therefore, the inpainted regions are usually much smaller than the natural ones. Thus, when we train the network to predict mask, there will exist a class imbalance which the standard CE loss fails to handle. The CE loss tends to focus on the majority of negative samples and leads to a low true positive rate to misclassify inpainted regions. Consequently, we adopt the focal loss proposed in [20] to lessen the effect of class imbalance. The Focal loss is a kind of general CE loss and we can regard CE loss as a special case of Focal loss. The Focal loss assigns an extra factor to the original cross entropy term, so the loss can control the gradient of different imbalanced samples. We use Focal loss which is formulated as:

$$\begin{aligned} L_{\text{Focal}}(y, \hat{y}) = &- \sum \alpha\,(1-\hat{y})^{\gamma} * y \log(\hat{y}) \\ &- \sum (1-\alpha)\hat{y}^{\gamma} * (1-y) \log(1-\hat{y}) \end{aligned} \tag{5}$$

where $y$ denotes pixel from the binary ground truth mask and $\hat{y}$ denotes the corresponding prediction pixel. $\alpha$ and $\gamma$

are hyperparameters.

In addition, we employ the mean Intersection of Union (mIoU) as our evaluation metric for video inpainting detection. As a result, to foster more intersection between the prediction mask and the binary ground truth, we adopt IoU score [28] as our loss function :

$$L_{\text{IoU}}(y, \hat{y}) = 1 - \frac{\sum y * \hat{y}}{\sum (y + \hat{y} - y * \hat{y}) + \epsilon} \tag{6}$$

where we set a hyperparameter $\epsilon$ which is just a small number to evade zero division.

Finally, the hybrid loss function for supervising the predictions is defined as follows:

$$L(y, \hat{y}) = \lambda_1 L_{\text{Focal}}(y, \hat{y}) + \lambda_2 L_{\text{IoU}}(y, \hat{y}) \tag{7}$$

There are two loss functions playing a significant role in the optimization, the Focal loss assists the network to alleviate class imbalance and pay attention to hard samples. Furthermore, the IoU loss directly measures the evaluation metric and guides the framework to predict inpainted regions more and more accurately.

# 4. Experiment

In this section, we evaluated our proposed method compared with previous image/video inpainting detection approaches. Then, we conducted experiments on video inpainting datasets from various widely used methods. Moreover, we investigated the robustness analysis and ablation study of our approach. Finally, we presented both quantitative and qualitative results.

## 4.1. Dataset and Metrics

We evaluated our framework both in-domain and cross-domain to investigate both the performance and generalization. For this reason, following [39], we chose Davis Video Inpainting dataset (DVI) and Free-form Video Inpainting dataset (FVI) to conduct various experiments, where there existed different approaches performing state-of-the-art for video inpainting task. Here we provide a brief description of these two datasets:

- DVI: Considering that DAVIS 2016 [25] is one of the most famous benchmarks for video inpainting, which contains 50 videos in total, we evaluated our proposed FAST on DVI dataset for video inpainting detection. We obtained inpainted videos utilizing three SOTA video inpainting approaches — VI [17], OP [23] and CP [18], regarding the ground truth mask as reference. We chose two out of all the three kinds of inpainted DAVIS videos for both training and testing. After that, we conducted additional cross-domain testing using the left kind of videos to test the generalization. We followed the original training/test set split.
- FVI: We conducted an additional evaluation on FVI dataset to investigate the generalization on different datasets. FVI dataset [6] consists of 100 test videos, which are used for multi-instance object removal and closer to the real-world scenario. We directly applied the approach proposed in [6]to acquire the corresponding 100 inpainted videos. To present the generalization of our proposed approach, we utilized the model which was trained on VI and OP inpainted DAVIS videos to directly test on FVI dataset.

We adopted the $F_1$ score and mean Intersection of Union (IoU) between the prediction masks the ground truth as evaluation metrics. Further, we reported the Area Under the Receiver Operating Characteristic Curve (AUC) as an additional evaluation metric.

## 4.2. Implementation Details

We implemented our FAST framework using the PyTorch package. For the specific transformer-based encoder, we directly adopted ViT [11] network. Moreover, all the Transformer backbones (*i.e.*, ViT) were pretrained on the ImageNet [9] dataset. Unless otherwise specified, the input image resolution and patch size $S$ were set as 224×224 and 16. As a result, we needed to cascade four consecutive $2\times$ upsampling blocks in PUP approach to recover the full resolution. Furthermore, we set the length of our video clips to 3 frames during training due to GPU memory limitation. We utilized SGD as the optimizer and set learning rate, momentum and weight decay as 0.01, 0.9 and 1e-4. The default batch size was set to 16 for both Davis Video Inpainting dataset and Free-form Video Inpainting Dataset.

## 4.3. Results on Davis Video Inpainting Dataset

We first evaluated our proposed method on DVI dataset and compared our proposed framework with existing video inpainting detection method VIDNet[39], video segmentation method COSNet [21] and manipulation detection methods consisting of *NOI* [22], *CFA* [12], *HPF* [19] and *GSR-Net* [38]. As for the various network architectures in [39], we adopted the VIDNet-IN framework which performs best in general. To explore the effect of different video inpainting methods, we tested all the models on one video inpainting method and trained on the other two.

Table 1 shows the results of different video inpainting detection methods, where bold numbers represent the best results. First, most existing manipulation detection methods are designed to obtain tamper artifacts in the images. Moreover, video segmentation method COSNet tends to obtain the flow difference between sequential frames to predict segmentation of objects. Therefore, these methods fail to achieve good performance. Recently, the first video inpainting detection framework VIDNet can learn temporal information between the frames and yields better performance.

Table 1. Comparison results on DVI dataset. We trained the model on DVI dataset inpainted by VI and OP methods, OP and CP methods, and VI and CP methods respectively (denoted as '*').

| Methods | **VI*** mIoU/F1 | **OP*** mIoU/F1 | **CP** mIoU/F1 | **VI** mIoU/F1 | **OP*** mIoU/F1 | **CP*** mIoU/F1 | **VI*** mIoU/F1 | **OP** mIoU/F1 | **CP*** mIoU/F1 |
|---|---|---|---|---|---|---|---|---|---|
| NOI [22] | 0.08/0.14 | 0.09/0.14 | 0.07/ 0.13 | 0.08/0.14 | 0.09/0.14 | 0.07/0.13 | 0.08/0.14 | 0.09/0.14 | 0.07/ 0.13 |
| CFA [12] | 0.10/0.14 | 0.08/0.14 | 0.08/0.12 | 0.10/0.14 | 0.08/0.14 | 0.08/0.12 | 0.10/0.14 | 0.08/0.14 | 0.08/0.12 |
| COSNet [21] | 0.40/0.48 | 0.31/0.38 | 0.36/0.45 | 0.28/0.37 | 0.27/0.35 | 0.38/0.46 | 0.46/0.55 | 0.14/0.26 | 0.44/0.53 |
| HPF [19] | 0.46/0.57 | 0.49/0.62 | 0.46/0.58 | 0.34/0.44 | 0.41 /0.51 | 0.68/0.77 | 0.55/0.67 | 0.19/ 0.29 | 0.69/0.80 |
| GSR-Net [38] | 0.57/0.69 | 0.50/0.63 | 0.51/0.63 | 0.30 /0.43 | 0.74/0.82 | 0.80/0.85 | 0.59 /0.70 | 0.22/0.33 | 0.70/0.77 |
| VIDNet [39] | 0.59/0.70 | 0.59/ 0.71 | 0.57/0.69 | **0.39 /0.49** | 0.74/0.82 | 0.81/0.87 | **0.59/ 0.71** | **0.25/0.34** | **0.76/0.85** |
| FAST (ours) | **0.61/0.73** | **0.65/0.78** | **0.63/0.76** | 0.32/**0.49** | **0.78/0.87** | **0.82/0.90** | 0.57/ 0.68 | 0.22/**0.34** | **0.76**/0.83 |

Table 2. Frame-level results for inpainting classification AUC comparison. We trained the models on VI and OP inpainted DAVIS videos (denoted as '*') and tested the models on all the three inpainting approaches.

| Methods | **VI*** | **OP*** | **CP** |
|---|---|---|---|
| HPF [19] | 0.718 | 0.640 | 0.845 |
| GSR-Net [38] | 0.762 | 0.758 | 0.834 |
| VIDNet [39] | 0.778 | 0.768 | 0.884 |
| FAST (ours) | **0.795** | **0.787** | **0.898** |

For all the three experimental settings, our FAST outperforms other approaches on all the untrained video inpainting approaches which presents the powerful generalization of our approach. Furthermore, our FAST achieves very competitive performance compared with other approaches on all the trained video inpainting approaches which presents the advantages of our approach to acquire inpainting artifacts distributed in the videos. Nevertheless, our FAST only achieves the second best result, which may be due to the significant difference between VI inpainting videos and CP inpainting ones.

Following [16] and [39], we also try to investigate the ability of our proposed approach to distinguish between original videos frames and inpainted ones. Same as above, we conducted the experiment with models trained on VI and OP inpainting approaches. Specially, we gave the inpainted frames positive labels and added the natural uninpainted frames to test sets as negative samples for additional evaluation. Moreover, we obtained a frame-level score by averaging the prediction result of all frames. Finally, we acquired the AUC classification performance of all the models. From table 2, we can observe that our model achieves the best performance compared to other models for all the three inpainting approaches. This suggests that our FAST framework successfully learn how to acquire discriminative information between inpainted and original videos.

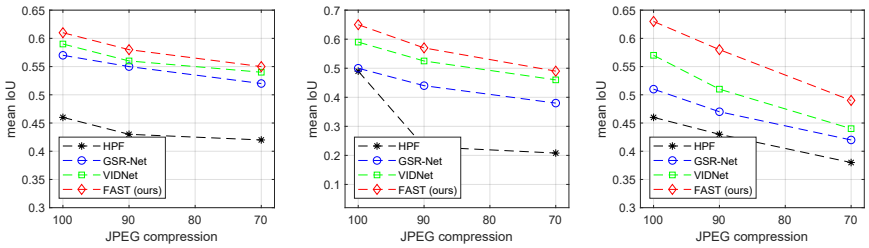
### 4.4. Ablation Study

We conducted four ablation studies on DVI dataset to investigate the effects of different individual components in our framework. Also, we conducted the experiment with models trained on VI and OP inpainting approaches. The ablation study consists of five settings which respectively adopt different network architectures. First, we only input one frame to our FAST framework to conduct experiments. Second, we dropped the Focal loss $L_{\text{Focal}}$ in the hybrid loss function. Third, we dropped the IoU loss $L_{\text{IoU}}$ in the hybrid loss function. Then, we gave up the frequency-aware features (FAF) and generate the prediction map directly using output of the transformer encoder. Finally, the last model was just our proposed FAST approach. The ablation study results are shown in Table 3.

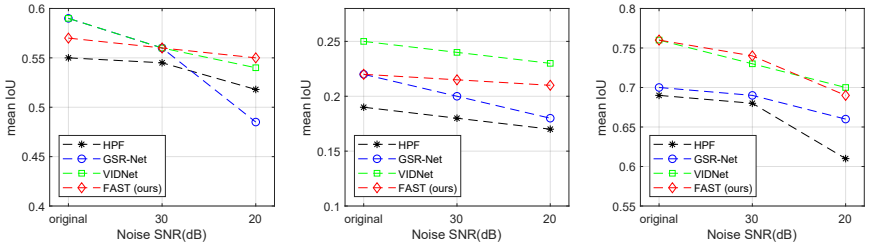
Table 3. Evaluation of different components of our proposed framework on DVI dataset. We trained the models on VI and OP inpainted DAVIS videos (denoted as '*').

| Methods | **VI*** mIoU/F1 | **OP*** mIoU/F1 | **CP** mIoU/F1 |
|---|---|---|---|
| Ours one frame | 0.57/0.68 | 0.53/0.60 | 0.51/0.61 |
| Ours w/o $L_{\text{Focal}}$ | 0.57/0.66 | 0.58/0.74 | 0.59/0.70 |
| Ours w/o $L_{\text{IoU}}$ | 0.58/0.68 | 0.59/0.73 | 0.59/0.72 |
| Ours w/o FAF | 0.55/0.64 | 0.51/0.62 | 0.48/0.54 |
| FAST (ours) | 0.61/0.73 | 0.65/ 0.78 | 0.63/0.76 |

From table 3, we can observe that our proposed FAST without frequency-aware features perform worse than other models. Similarly, we conducted the experiment with models trained on VI and OP inpainting approaches. This is perhaps because the frequency-aware features can explore the inpainting artifacts hidden in the RGB frames and the combination of frequency domain features and spatial domain information improve the discriminative ability of the model. Moreover, we can observe that our model does not perform well without any loss function, and it suggests that these two loss functions both play an important role in optimizing the FAST model and promoting the evaluation performance. In addition, the IoU loss $L_{\text{IoU}}$ occupies a slightly more important position because it is directly related to the evaluation metric. Obviously, the FAST with one frame input fails to perform well. This model can only formulate the spatial attention between image patches, ignoring the important temporal connections between video frames. Furthermore, we believe that the performance of FAST will get better and better as the number of frames increases within a certain range.




(a) JPEG perturbation (VI*, OP*, CP)




(b) Noise perturbation (VI*, OP*, CP)

Figure 4. Comparison results under different perturbations. We chose the quality factor with 90 and 70 for perturbation in JPEG compression. We chose SNR 30dB and 20dB for perturbation in noise. From the left column to right, the results are respectively on VI, OP and CP inpainting. We adopt '*' to denote the inpainting methods which the model was trained on.

### 4.5. Robustness Analysis

We conducted experiments under various perturbations to study the robustness of our FAST approach under JPEG and noise perturbation. First, as for JPEG perturbation, we compressed the input frames with 70 and 90 JPEG quality factor. Further, for noise perturbation, we added Gaussian noise to the test video frames with Signal-to-Noise Ratio (SNR) 20 and 30 dB.

The robustness analysis results suggest that our FAST approach performs well for the robustness under different perturbations. Because we introduce many high frequency noises, there is a dramatic drop in HPF for perturbation compared to other methods. However, VIDNet suffers more from JPEG perturbation than Noise perturbation, because VIDNet utilizes the ELA features which are very sensitive to the JPEG compression. Finally, our FAST approach achieves the best robustness and only performs a small degradation under different perturbations, because we introduce the frequency-aware features focused on enriched information with different decomposed frequency components.

### 4.6. Results on Free-form Video Inpainting Dataset

To further study the problem of generalization, we evaluated our proposed method on FYI dataset. Moreover, we investigated the generalization between different datasets instead of various video inpainting approaches. The models were all trained on VI and OP inpainting methods and the generalization analysis results are shown in Table 4. All the proposed approaches suffer from the performance degradation when applied for cross-datasets testing because there are significant differences between the two datasets and inpainting approaches. However, our method still achieves competitive generalization compared to existing methods due to the combination with frequency-aware features and utilization of temporal information.

Table 4. Comparison results on FVI dataset. We trained the model on DVI dataset inpainted by VI and OP methods and directly tested it on FVI dataset.

| | **FVI** |
|---|---|
| Methods | mIoU/F1 |
| NOI [22] | 0.062/0.107 |
| CFA [12] | 0.073/0.122 |
| HPF [19] | 0.205/0.285 |
| GSR-Net [38] | 0.195/0.288 |
| VIDNet [39] | 0.257/**0.367** |
| FAST (ours) | **0.285**/0.359 |

### 4.7. Qualitative Results

Figure 5 indicates the visualization results of our proposed FAST compared with other approaches under the same setting. We can observe that our approach can predict the masks closest to the ground truth, because our frequency-aware features extract discriminative information and the spatiotemporal transformer formulate the temporal connections. Specifically, HPF tends to misclassify the real regions due to limits of single input modality. Furthermore, GSR-Net conducts frame-level inpainting detection so that the results are not temporally consistent. The VIDNet utilizes the temporal information to maintain consistency, but the prediction results miss some details.

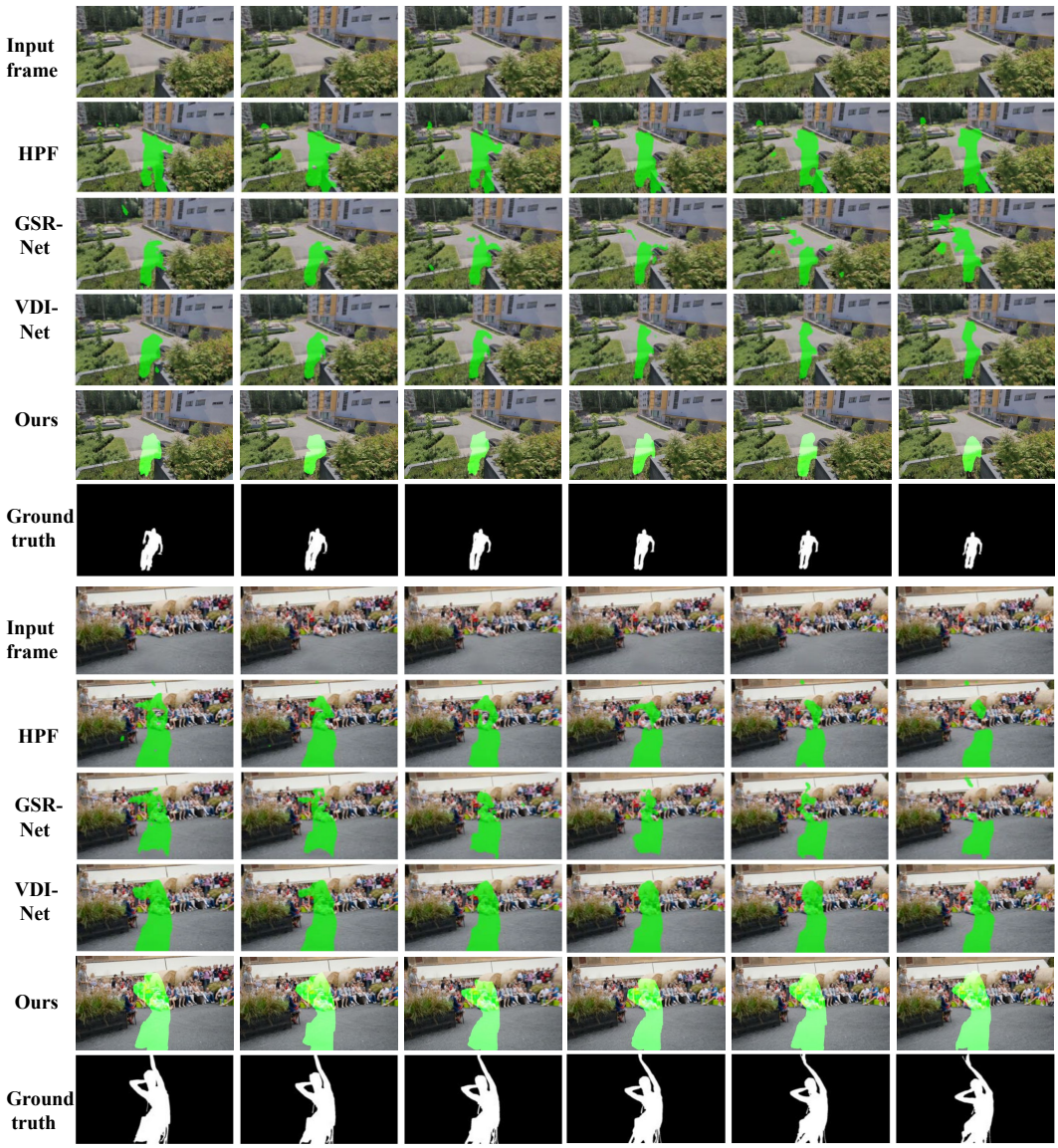



Figure 5. Qualitative visualization results on DVI dataset. From the first row, we present the inpainted video frames. From the second to the fifth row, these images show the final prediction results of different methods and we utilize the green mask to highlight the results. The sixth row is the ground truth (best viewed digitally, in color and with zoom).

## 5. Conclusions

In this paper, we have proposed to learn frequency-aware spatiotemporal transformers for video inpainting detection, aiming to simultaneously mine the spatial, temporal, and frequency-aware traces of inpainted videos. While existing deep video inpainting detection methods usually rely on hand-designed attention modules and memory mechanism, our proposed FAST possesses innate global self-attention mechanisms to capture the long-range dependency. Furthermore, we adopt a spatiotemporal transformer to detect the spatial connections between patches and temporal dependency between frames. Because the inpainted videos usually lack high frequency details, we employ a specifically designed decoder to synchronously exploit the frequency domain information. Experimental results have shown that our approach achieves very competitive performance.

## Acknowledge

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, and Grant U1713214, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI), and in part by a grant from the Institute for Guo Qiang, Tsinghua University.

## References


[1] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *TC*, 100(1):90–93, 1974.

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. In *ToG*, 2009.

[3] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *ICCV*, pages 3286–3295, 2019.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020.

[5] I-Cheng Chang, J Cloud Yu, and Chih-Chuan Chang. A forgery detection algorithm for exemplar-based inpainting images using multi-region relation. *IVC*, 31(1):57–71, 2013.

[6] Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *ICCV*, 2019.

[7] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. *arXiv preprint arXiv:2012.00364*, 2020.

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*, pages 2978–2988, 2019.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[11] 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.

[12] Pasquale Ferrara, Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva. Image forgery localization via fine-grained analysis of cfa artifacts. In *TIFS*, 2012.

[13] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *ICML*, pages 3247–3258. PMLR, 2020.

[14] Han Hu, Zheng Zhang, Zhenda Xie, and Stephen Lin. Local relation networks for image recognition. In *ICCV*, pages 3464–3473, 2019.

[15] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Temporally coherent completion of dynamic video. *TOG*, 2016.

[16] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *ECCV*, 2018.

[17] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *CVPR*, 2019.

[18] Sungho Lee, Seoung Wug Oh, DaeYeun Won, and Seon Joo Kim. Copy-and-paste networks for deep video inpainting. In *ICCV*, 2019.

[19] Haodong Li and Jiwu Huang. Localization of deep inpainting using high-pass fully convolutional network. In *ICCV*, 2019.

[20] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017.

[21] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *CVPR*, 2019.

[22] Babak Mahdian and Stanislav Saic. Using noise inconsistencies for blind image forensics. In *IMAVIS*, 2009.

[23] Seoung Wug Oh, Sungho Lee, Joon-Young Lee, and Seon Joo Kim. Onion-peel networks for deep video completion. In *ICCV*, 2019.

[24] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[25] 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 *CVPR*, 2016.

[26] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *ECCV*, pages 86–103, 2020.

[27] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models. In *NeurIPS*, 2019.

[28] Mengye Ren and Richard S Zemel. End-to-end instance segmentation with recurrent attention. In *CVPR*, 2017.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[30] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *ECCV*, pages 108–126, 2020.

[31] Wei Wang, Jing Dong, and Tieniu Tan. Tampered region localization of digital color images based on jpeg compression noise. In *IWDW*, 2010.

[32] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018.

[33] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. *arXiv preprint arXiv:2011.14503*, 2020.

[34] Felix Wu, Angela Fan, Alexei Baevski, Yann Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. In *ICLR*, 2018.

[35] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *CVPR*, 2019.

[36] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: General-

ized autoregressive pretraining for language understanding. In *NeurIPS*, pages 5753–5763, 2019.

[37] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *arXiv preprint arXiv:2012.15840*, 2020.

[38] Peng Zhou, Bor-Chun Chen, Xintong Han, Mahyar Najibi, Abhinav Shrivastava, Ser Nam Lim, and Larry S Davis. Generate, segment and refine: Towards generic manipulation segmentation. *AAAI*, 2020.

[39] Peng Zhou, Ning Yu, Zuxuan Wu, Larry S Davis, Abhinav Shrivastava, and Ser-Nam Lim. Deep video inpainting detection. *arXiv preprint arXiv:2101.11080*, 2021.

[40] Xinshan Zhu, Yongjun Qian, Xianfeng Zhao, Biao Sun, and Ya Sun. A deep learning approach to patch-based image inpainting forensics. *SPIC*, 2018.