Short-Term and Long-Term Context Aggregation Network for Video Inpainting

Ang Li[1], Shanshan Zhao[2], Xingjun Ma[3], Mingming Gong[4], Jianzhong Qi[1], Rui Zhang[1], Dacheng Tao[2], and Ramamohanarao Kotagiri[1]

[1] School of Computing and Information Systems, The University of Melbourne, Australia {angl4@student., jianzhong.qi@, rui.zhang@, kotagiri@}unimelb.edu.au

[2] UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering, The University of Sydney, Darlington, NSW 2008, Australia {szha4333@uni., dacheng.tao@}sydney.edu.au

[3] School of Information Technology, Deakin University, Geelong, Australia daniel.ma@deakin.edu.au

[4] School of Mathematics and Statistics, The University of Melbourne, Australia mingming.gong@unimelb.edu.au

Abstract. Video inpainting aims to restore missing regions of a video and has many applications such as video editing and object removal. However, existing methods either suffer from inaccurate short-term context aggregation or rarely explore long-term frame information. In this work, we present a novel context aggregation network to effectively exploit both short-term and long-term frame information for video inpainting. In the encoding stage, we propose **boundary-aware short-term context aggregation**, which aligns and aggregates, from neighbor frames, local regions that are closely related to the boundary context of missing regions into the target frame[5]. Furthermore, we propose **dynamic long-term context aggregation** to globally refine the feature map generated in the encoding stage using long-term frame features, which are dynamically updated throughout the inpainting process. Experiments show that it outperforms state-of-the-art methods with better inpainting results and fast inpainting speed.

Keywords: Video Inpainting, Context Aggregation

1 Introduction

Video inpainting aims to restore missing regions in a video with plausible contents that are both spatially and temporally coherent [7, 15]. It can benefit a wide range of practical video applications such as video editing, damage restoration, and undesired object removal. Whilst significant progress has been made in image inpainting [8, 14, 17, 25, 28, 30, 31], it is challenging to extend image inpainting methods to solve the video inpainting problem. Directly applying image

[5] The target frame refers to the current input frame under inpainting.

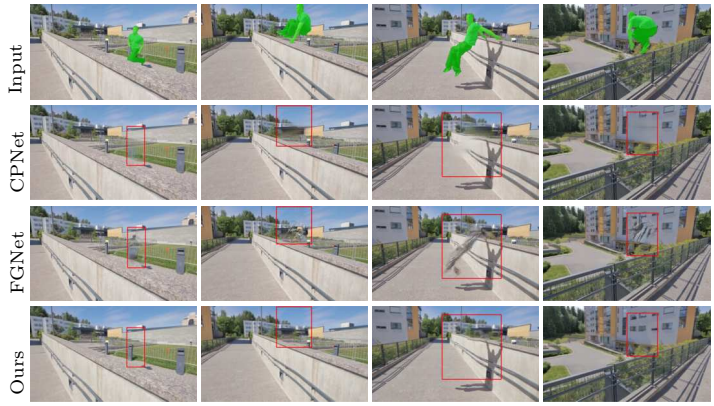

Fig. 1. Comparison with state-of-the-art CPNet [13] and FGNet [27]. The green areas in the input frames are the missing regions. Best viewed at zoom level 400%.

Fig. 2. An example of missing regions' negative effects on flow-warping-based aggregation. Given a target frame X_t and a reference frame X_i, we get the optical flow $F_{t\to i}$ between them using a pretrained flow estimator. Then we warp X_i onto X_t and get the warped frame $W(F_{t\to i}, X_i)$. Heavy distortions can be found in $F_{t\to i}$ and $W(F_{t\to i}, X_i)$ within the missing regions. Our network alleviates this problem, producing more accurate aggregation.

inpainting methods on individual video frames may lose the inter-frame motion continuity and content dependency, which causes temporal inconsistencies and unexpected flickering artifacts.

Traditional video inpainting methods [7, 15, 24] utilize patch-based optimization strategies to fill missing regions with sampled patches from known regions. These methods often suffer from limited effectiveness and vulnerability to complex motions. Recently, deep learning-based video inpainting methods [11, 13, 16, 21] have improved the inpainting performance by a large margin. Most of them use encoder-decoder structures following a frame-by-frame inpainting pipeline to borrow information from reference frames[6] and perform different types of context aggregation to restore the target frame.

In spite of the encouraging results, deep learning-based methods still need to overcome the following limitations. First, they fail to make effective usage of short-term and long-term reference information in the input video. Studies [11, 21] restrict the range of reference frames to nearby (short-term) frames of

[6] The reference frames refer to other frames from the same video.

the target frame so as to maintain temporal consistency. When dealing with diverse motion patterns in videos (eg., slowly moving views or objects), short-term frames alone cannot provide sufficient information to restore the target frame. Other methods [13, 16] often sample a set of fixed frames from the input video (eg., every 5-th frame) as the reference frames. Although this can exploit some long-term information, it tends to include irrelevant contexts, reduce temporal consistency, and increase the computation time. Second, how to achieve accurate context aggregation remains challenging. Since missing regions contain no visual information, it is difficult to find the most related local regions in reference frames for accurate context aggregation. For example, a recent method [11] uses estimated optical flows to warp reference frames onto the target frame and further aggregate them together. As shown in Figure 2, the flow information within the missing region is inaccurate and is distorted. This will cause unexpected artifacts when using the distorted flow to do warping and context aggregation, and the distortion artifacts will be propagated and accumulated during the encoding process. While using more fixed reference frames from the input video may help mitigate the negative effects of missing regions, it inevitably brings more irrelevant or even noisy information into the target frame, which inadvertently does more harm than good.

In this paper, we aim to address the challenges above in a principled manner from the following three aspects: (1) We propose a novel framework for video inpainting that integrates the advantages of both short-term and long-term reference frames. Different from existing methods that only conduct context aggregation at the decoding stage, we propose to start context aggregation at the encoding stage with short-term reference frames. This can help provide more temporally consistent local contexts. Then, at the decoding stage, we refine the encoding-generated feature map with a further step of context aggregation on long-term reference frames. This refinement can deal with more complex motion patterns. (2) To better exploit short-term information, we propose *boundary-aware short-term context aggregation* at the encoding stage. Different from existing methods, here, we pay more attention to the boundary context of missing regions. Our intuition is that, in the target frame, the boundary area around the missing regions is more related to the missing regions than other areas of the frame. Considering the spatial and motion continuity of videos, if we can accurately locate and align the boundary context of missing regions with the corresponding regions in the reference frames, it would improve both the spatial and temporal consistency of the generated contents. This strategy can also alleviate the impact of missing regions at context aggregation. (3) To better exploit long-term information, we propose a *dynamic long-term context aggregation* at the decoding stage. Since different videos have different motion patterns (eg., slow moving or back-and-forth moving), they have different contextual dependency between frames. Therefore, it is necessary to eliminate frames that are largely irrelevant with the target frame. Our dynamic strategy aims for the effective usage of long-term frame information. Specifically, instead of simply using fixed reference frames, we propose to dynamically update the long-term

reference frames used for inpainting, according to similarities of other frames to the current target frame.

In summary, our main contributions are:

- We propose a novel framework for video inpainting that effectively integrates context information from both short-term and long-term reference frames.
- We propose a *boundary-aware short-term context aggregation* to better exploit the context information from short-term reference frames, by using the boundary information of the missing regions in the target frames.
- We propose a *dynamic long-term context aggregation* as a refinement operation to better exploit the context information from dynamically updated long-term reference frames.
- We empirically show that our proposed network outperforms the state-of-the-art methods with better inpainting results and fast inpainting speed.

2 Related Work

2.1 Image Inpainting

Traditional image inpainting methods [1, 2] mostly perform inpainting by finding pixels or patches outside missing regions or from the entire image database. These methods often suffer from low generation quality, especially when dealing with complicated scenes or large missing regions [8, 17].

Deep neural networks have been used to improve inpainting results [8, 14, 17, 19, 25, 28–31]. Pathak *et al.* [17] introduce the *Context Encoder* (CE) model where a convolutional encoder-decoder network is trained with the combination of an adversarial loss [6] and a reconstruction loss. Iizuka *et al.* [8] propose to utilize global and local discriminators to ensure consistency on both entirety and details. Yu *et al.* [30] propose the contextual attention module to restore missing regions with similar patches from undamaged regions in deep feature space.

2.2 Video Inpainting

Apart from spatial consistency in every restored frame, video inpainting also needs to solve a more challenging problem: how to make use of information in the whole video frame sequence and maintain temporal consistency between the restored frames. Traditional video inpainting methods adopt patch-based strategies. Wexler *et al.* [24] regard video inpainting as a global optimization problem by alternating between patch search and reconstruction steps. Newson *et al.* [15] extend this and improve the search algorithm by developing a 3D version of PatchMatch [1]. Huang *et al.* [7] introduce the optical flow optimization in spatial patches to enforce temporal consistency. These methods require heavy computations, which limit their efficiency and practical use.

Recently, several deep learning-based methods have been proposed [3, 11, 13, 16, 21, 27, 32]. These works can be divided into two groups. The first group mainly relies on short-term reference information when inpainting the target frame.

For example, VINet [11] uses a recurrent encoder-decoder network to collect information from adjacent frames via flow-warping-based context aggregation. Xu *et al.* [27] propose a multi-stage framework for video inpainting: they first use a deep flow completion network to restore the flow sequence, then perform forward and backward pixel propagation using the restored flow sequence, and finally use a pretrained image inpainting model to refine the results. The second group uses a fixed set of frames from the entire video as reference information. Wang *et al.* [21] propose a two-stage model with a combination of 3D and 2D CNNs. CPNet [13] conducts context aggregation by predicting affine matrices and applying affine transformation on fixedly sampled reference frames.

Although these video inpainting methods have shown promising results, they still suffer from ineffective usage of short-term and long-term frame reference information, and inaccurate context aggregation as discussed in Section 1.

3 Short-Term and Long-Term Context Aggregation Network

Given a sequence of continuous frames from a video $X \coloneqq \{X_1, X_2, ..., X_T\}$ annotated with binary masks $M \coloneqq \{M_1, M_2, ..., M_T\}$, a video inpainting network outputs the restored video $\hat{Y} \coloneqq \{\hat{Y}_1, \hat{Y}_2, ..., \hat{Y}_T\}$. The goal is that $\hat{Y}$ should be spatially and temporally consistent with the ground truth video $Y \coloneqq \{Y_1, Y_2, ..., Y_T\}$.

3.1 Network Overview

Our network is built upon a recurrent encoder-decoder architecture and processes the input video frame by frame in its temporal order. An overview of our proposed network is illustrated in Figure 3. Different from existing methods, we start inpainting (context aggregation) at the encoding stage. Given current target frame X_t, we choose a group of neighboring frames $\{X_i\}$ with $i \in \{t-6, t-3, t+3, t+6\}$) as the short-term reference frames for X_t. During encoding, we have two sub-encoders: encoder a for the stream of target frame and encoder b for the other four streams of reference frames. The encoding process contains three feature spatial scales $\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}\}$. At each encoding scale, we perform **Boundary-aware Short-term Context Aggregation (BSCA)** between feature maps of the target frame and those of the short-term reference frames, to fill the missing regions in the *target feature map*. This module can accurately locate and aggregate relevant bounding regions in short-term reference frames and at the same time avoid distractions caused by missing regions in the target frame. At the decoding stage, our **Dynamic Long-term Context Aggregation (DLCA)** module refines the encoding-generated feature map using dynamically updated long-term features. This module stores long-term frame features selected from previously *restored* frames, and updates them according to their contextual correlation to the current target frame. Intuitively, it only keeps those long-term frame features that are more contextual relevant to the current target frame. We also adopt a convolutional LSTM (Conv-LSTM) layer

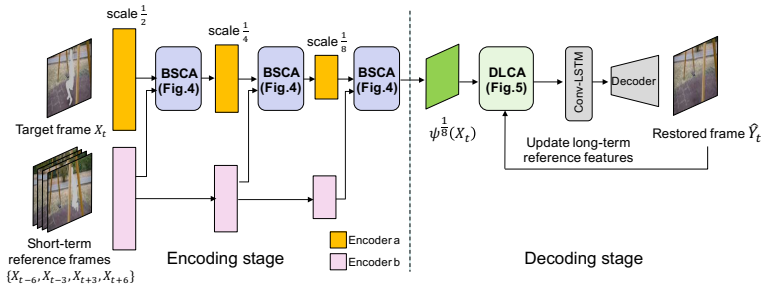

Fig. 3. Overview of our proposed network. In the encoding stage, we conduct Boundary-aware Short-term Context Aggregation (BSCA) (Sec. 3.2) using short-term frame information from neighbor frames, which is beneficial to context aggregation and generating temporally consistent contents. In the decoding stage, we propose the Dynamic Long-term Context Aggregation (DLCA) (Sec. 3.3), which utilizes dynamically updated long-term frame information to refine the encoding-generated feature map.

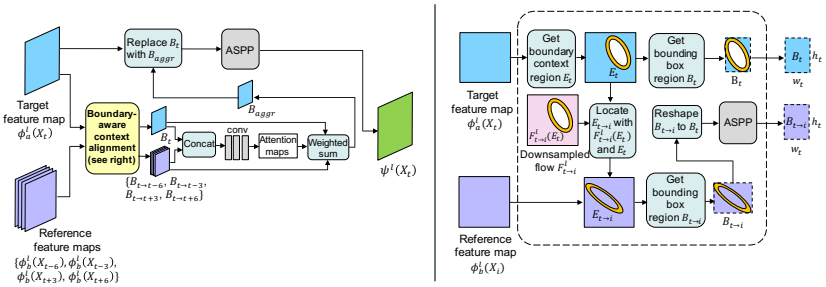

Fig. 4. Left: Boundary-aware Short-term Context Aggregation (BSCA) module. **Right**: The boundary-aware context alignment operation in BSCA. Here, $l \in \{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}\}$ refers to the encoding scale.

to increase temporal consistency as suggested by Lai *et al.* [12]. Finally, the decoder takes the refined latent feature to generate the restored frame $\hat{Y}_t$. Since the missing regions are now filled with contents, we replace the target frame X_t by the restored frame $\hat{Y}_t$, which provides more accurate information for the following iterations.

3.2 Boundary-aware Short-term Context Aggregation

Optic flows between frames have been shown to be essential for alignment with short-term reference frames. Previous optic-flow-based works [11, 27] conduct context aggregation by warping the reference frames onto the target frame. However, missing regions in the target frame become occlusion factors and may lead to incorrect warping, as we have shown in Figure 2. To alleviate this problem, we propose to utilize optic flows in a novel way: instead of using optic flows to do warping, we only use them to locate the corresponding bounding regions in the

reference frame feature map that match the surrounding context of the missing regions in the target frame feature map. Here, we define the surrounding context region as the non-missing pixels that are within a Euclidean distance d ($d = 8$ in our experiments) to the nearest pixels from the missing regions.

The structure of BSCA is illustrated in the left subfigure of Figure 4. At a certain encoding scale $l \in \{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}\}$, we have the target feature map $\phi_a^l(X_t)$ from encoder a and the reference feature maps $\{\phi_b^l(X_i)\}$ from encoder b as input for boundary-aware context aggregation. We first obtain the bounding region B_t of the missing region in $\phi_a^l(X_t)$ and its corresponding bounding regions $\{B_{t\to i}\}$ in $\{\phi_b^l(X_i)\}$. Then, we apply an attention-based aggregation to combine B_t and $\{B_{t\to i}\}$ as B_{aggr}. We replace B_t in $\phi_a^l(X_t)$ with B_{aggr} and obtain the restored target feature map $\psi^l(X_t)$, which, together with the original reference feature maps $\{\phi_b^l(X_i)\}$, is passed on to the next encoding scale (see Figure 3). Two essential operations in this process, i.e., 1) boundary-aware context alignment and 2) attention-based aggregation, are detailed below.

Boundary-aware Context Alignment. As illustrated in the right subfigure of Figure 4, the alignment operation takes the target feature map $\phi_a^l(X_t)$ and a reference feature map $\phi_b^l(X_i)$ as inputs. In $\phi_a^l(X_t)$, we denote the missing region with white color. Then, surrounding region E_t in $\phi_a^l(X_t)$ is obtained (the yellow elliptical ring in $\phi_a^l(X_t)$). We further obtain the bounding box region of E_t and denote it by B_t. We use a pretrained FlowNet2 [9] to extract the flow information $F_{t\to i}$ between X_t and X_i, and then we downsample $F_{t\to i}$ to $F_{t\to i}^l$ for current encoding scale l. In $F_{t\to i}^l$, the corresponding flow information of E_t is denoted as $F_{t\to i}^l(E_t)$, which has the same position with E_t. With E_t and $F_{t\to i}^l(E_t)$, we can locate the corresponding region of E_t in $\phi_b^l(X_i)$, which is $E_{t\to i}$ (the yellow elliptical ring in $\phi_b^l(X_i)$). We also obtain the bounding box region of $E_{t\to i}$ as $B_{t\to i}$, and reshape it to the shape of B_t. To ensure the context coherence, we further refine the reshaped $B_{t\to i}$ using Atrous Spatial Pyramid Pooling (ASPP) [4]. With the aligned bounding regions B_t and $B_{t\to i}$, we can alleviate the impact of missing regions and achieve more accurate context aggregation.

Attention-based Aggregation. Attention-based aggregation can help find the most relevant features from the reference feature maps, and eliminate irrelevant contents, eg., newly appeared backgrounds. We first append B_t into the set $\{B_{t\to i}\}$ to get a new set $\{B_{t\to i}, B_t\}$, denoted as $\{B_j'\}_{j=1}^5$. Then, we concatenate the elements in the new set along the channel dimension, and apply convolutional and softmax operations across different channels to obtain the attention maps $\{A_j\}_{j=1}^5$. Finally, the attention-based aggregation is performed as follows.

$$B_{aggr} = \sum_{j=1}^{5} A_j B_j^{'}, \tag{1}$$

We replace B_t with the aggregated bounding region B_{aggr} in the target feature map $\phi_a^l(X_t)$. The replaced target feature map is processed by an ASPP module to get the $\psi^l(X_t)$.

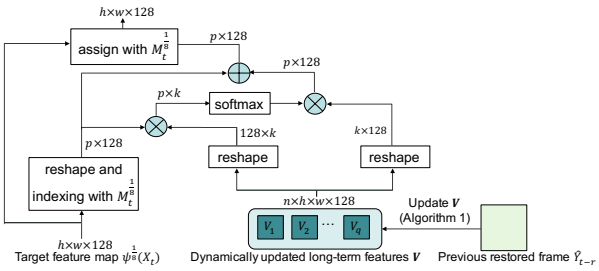

Fig. 5. The Dynamic Long-term Context Aggregation (DLCA) module.

3.3 Dynamic Long-term Context Aggregation

Fixed sampling long-term reference frames [13, 16] fail to consider the motion diversity of videos. Thus, they may inevitably bring more irrelevant or even noisy information. Since different videos have different motion patterns (eg., slow moving or back-and-forth moving), it results in different contextual dependency between frames. Therefore, it is necessary that the selected long-term reference information is contextually relevant to the current target frame. We use a dynamic strategy for the effective use of long-term reference information. The structure of this decoding-stage context aggregation module is illustrated in Figure 5. It refines the feature map generated in the above encoding stage with 1) dynamically updated long-term features and 2) non-local-based aggregation.

Algorithm 1 Update Long-Term Features

Input: previous restored frame $\hat{Y}_{t-r}$, current target frame X_t, long-term features V
Output: updated V

distance = [ ]
U_{X_t} = Encoder_b(X_t)
$U_{\hat{Y}_{t-r}}$ = Encoder_b($\hat{Y}_{t-r}$)
$d_{\hat{Y}_{t-r}} = \|U_{X_t} - U_{\hat{Y}_{t-r}}\|_1$
for V_r in V **do**
    $d_r = \|U_{X_t} - V_r\|_1$
    distance.append(d_r)
end for
d_{max}, max = get_max_and_index(distance)
if $d_{\hat{Y}_{t-r}} < d_{max}$ **then**
    V.remove(V_{max})
    V.append($U_{\hat{Y}_{t-r}}$)
end if

Dynamically Updated Long-Term Features. DLCA stores the features of the previously *restored* frames that are most relevant (in feature space) to the current target frame. Specifically, $V \coloneqq \{V_1, V_2, ..., V_q\}$ stores a set of long-term feature maps with the length q, which are updated dynamically following Algorithm 1. At each inpainting iteration, it checks whether the feature map

of a long-term frame $\hat{Y}_{t-r}$ (r is the parameter that defines how far from the current target frame to look back) can be incorporated into the V set according to its L_1 distance to target frame X_t in the feature space. Let U_{X_t} and $U_{\hat{Y}_{t-r}}$ be the feature maps of X_t and $\hat{Y}_{t-r}$ respectively, if the L_1 distance between $U_{\hat{Y}_{t-r}}$ and U_{X_t} is smaller than the maximum distance between a feature map in the current V set to U_{X_t}, then $U_{\hat{Y}_{t-r}}$ will replace the corresponding feature map (that has the maximum distance) in the V set. Note that these feature maps can be obtained using our encoder b. We suggest $r \geq 7$ to exploit long-term information (as short-term information from $\hat{Y}_{t-6/t-3}$ has already been considered by our BSCA module). At the beginning when $t < r$, we simply set $r = |t - r|$ to use all restored frames so far. With this dynamic updating policy, DLCA can automatically adjust the stored long-term frame features and remove irrelevant ones, regarding each target frame.

Non-local-based Aggregation. Based on the long-term feature set V, we follow a typical approach [23] to perform non-local-based context aggregation between the target feature map $\psi^{\frac{1}{8}}(X_t)$ and feature maps stored in V, as shown in Figure 5. Softmax is applied to obtain the normalized soft attention map over feature maps in V. The attention map is then utilized as weights to compute an aggregated feature map from V via weighted summation. Finally, the aggregated feature map replaces the feature map of missing regions.

3.4 Loss Function

The loss function used for training is:

$$\mathcal{L}_{total} = \mathcal{L}_{\text{rec}} + \lambda_{\text{mre}}\mathcal{L}_{\text{mre}} + \lambda_{\text{per}}\mathcal{L}_{\text{per}} + \lambda_{\text{style}}\mathcal{L}_{\text{style}}, \tag{2}$$

Here, $\mathcal{L}_{\text{rec}}$, $\mathcal{L}_{\text{mre}}$, $\mathcal{L}_{\text{per}}$, and $\mathcal{L}_{\text{style}}$ denote reconstruction loss, reconstruction loss of mask region, perceptual loss, and style loss respectively. The balancing weights λ_{mre}, λ_{per} and λ_{style} are empirically set to 2, 0.01, and 1, respectively.

The reconstruction loss and the reconstruction loss of the mask region are defined on pixels:

$$\mathcal{L}_{\text{rec}} = \sum_{t}^{T} \|\hat{Y}_t - Y_t\|_1. \tag{3}$$

$$\mathcal{L}_{\text{mre}} = \sum_{t}^{T} \|(1 - M_t) \odot (\hat{Y}_t - Y_t)\|_1, \tag{4}$$

where $\odot$ is the element-wise multiplication. To further enhance inpainting quality, we include two additional loss functions: perceptual loss [10] and style loss,

$$\mathcal{L}_{\text{per}} = \sum_{t}^{T} \sum_{s}^{S} \frac{\|\sigma_s(\hat{Y}_t) - \sigma_s(Y_t)\|_1}{S}, \tag{5}$$

$$\mathcal{L}_{\text{style}} = \sum_{t}^{T} \sum_{s}^{S} \frac{\|G_s^\sigma(\hat{Y}_t) - G_s^\sigma(Y_t)\|_1}{S}, \tag{6}$$

where σ_s is the s-th layer output of an ImageNet-pretrained VGG-16 [20] network, S is the number of chosen layers (i.e., $relu_{2\_2}$, $relu_{3\_3}$ and $relu_{4\_3}$), and G denotes the gram matrix multiplication [5].

4 Experiments

We evaluate and compare our model with state-of-the-art models qualitatively and quantitatively. We also conduct a comprehensive ablation study on our proposed model.

Table 1. Quantitative comparisons on YouTube-VOS and DAVIS datasets under three mask settings regarding 3 performance metrics: PSNR (higher is better), SSIM (higher is better) and VFID (lower is better). The rightmost column shows the average execution time to inpaint one video. The best results are in **bold**.

YouTube-VOS										
Model	Square mask			Irregular mask			Object mask			Time (sec.)
	PSNR	SSIM	VFID	PSNR	SSIM	VFID	PSNR	SSIM	VFID	
VINet [11]	26.92	0.843	0.103	27.33	0.848	0.082	26.61	0.838	0.118	**33.6**
CPNet [13]	27.24	0.847	0.087	27.50	0.852	0.051	27.02	0.845	0.087	48.5
FGNet [27]	27.71	0.856	0.082	27.91	0.859	0.056	27.32	0.849	0.083	276.3
Ours	**27.76**	**0.858**	**0.076**	**28.12**	**0.866**	**0.047**	**27.45**	**0.853**	**0.075**	35.4
DAVIS										
Model	Square mask			Irregular mask			Object mask			Time (sec.)
	PSNR	SSIM	VFID	PSNR	SSIM	VFID	PSNR	SSIM	VFID	
VINet [11]	27.88	0.863	0.060	28.67	0.874	0.043	27.02	0.850	0.068	**19.7**
CPNet [13]	27.92	0.862	0.049	28.81	0.876	0.031	27.48	0.855	0.049	28.2
FGNet [27]	28.32	0.870	0.045	29.37	0.880	0.033	**28.18**	0.864	0.046	194.8
Ours	**28.50**	**0.872**	**0.038**	**29.56**	**0.883**	**0.027**	28.13	**0.867**	**0.042**	21.5

Datasets. Following previous works [11, 13], we train and evaluate our model on YouTube-VOS [26] and DAVIS [18] datasets. For YouTube-VOS, we use the 3471 training videos for training, and the 508 test videos for testing. For DAVIS, we use the 60 unlabeled videos to fine tune a pretrained model on YouTube-VOS, and the 90 videos with object mask annotations for testing. All video frames are resized to 424×240 , and no pre-processing or post-processing is applied.
Mask Settings. To simulate the diverse and ever-changing real-world scenarios, we consider the following three mask settings for training and testing.

- Square mask: The same square region for all frames in a video, but has a random location and a random size ranging from 40×40 to 160×160 for different videos.
- Irregular mask: We use the irregular mask dataset [14] that consists of masks with arbitrary shapes and random locations.

Fig. 6. Qualitative comparison of our proposed model with baseline models on DAVIS dataset. Better viewed at zoom level 400%. More video results can be found in supplementary material.

– Object mask: Following [11, 27], we use the foreground object masks in DAVIS [18] which has continuous motion and realistic appearance. Note that when quantitatively testing object masks on DAVIS dataset, we shuffle its video-mask pairs for more reasonable result, as it is a dataset for object removal and the ground-truth background (after removal) is unknown (without shuffling, the original objects will become the ground truth).

Baseline Models. We compare our model with three state-of-the-art video inpainting models: 1) VINet [11], a recurrent encoder-decoder network with flow-warping-based context aggregation; 2) CPNet [13], which conducts context aggregation by predicting affine matrices and applying affine transformation on fixedly sampled reference frames; and 3) FGNet [27], which consists of three stages: first restores flow between frames, then performs forward and backward warping with the restored flow, and finally utilizes an image inpainting model for post-processing.

4.1 Quantitative Results

We consider three metrics for evaluation: 1) *Peak Signal-to-Noise Ratio* (PSNR, measures image distortion), 2) *Structural Similarity* (SSIM, measures structure similarity) and 3) the video-based Fréchet Inception Distance (VFID, a video perceptual measure known to match well with human perception) [22]. As shown in Table 1, our model outperforms all baseline models according to all three metrics across all three mask settings on both datasets, a clear advantage of using both short-term and long-term information. In terms of execution time, our model is comparable to VINet, which has the least average execution time but worse performance than all other three models. Overall, our model achieves the best trade-off between performance and execution time on the two test sets.

4.2 Qualitative Results

To further inspect the visual quality of the inpainted videos, we show, in Figure 6, three examples of the inpainted frames by our model and the compared baselines. As can be observed, frames inpainted by our models are generally of much higher quality than those by VINet or CPNet, and also perceptibly better than the state-of-the-art model FGNet. For example, in the third example (right two columns), the car structures generated by VINet are highly distorted. This is mainly caused by the occlusion effect of mask regions in the target frame, and its limited exploration of long-term information. CPNet was able to restore the rough structures of the car with more information from its fixedly sampled long-term reference frames. However, blurriness or overlapping can still be found since those fixed-term reference frames also bring in a significant amount of irrelevant contexts. FGNet in general achieves sharper results than VINet or CPNet. However, it also generates artifacts in this example. This can be ascribed to inaccurate flow inpainting in the first stage of FGNet. In contrast, our model can generate more plausible contents with high spatial and temporal consistency.

4.3 User Study

We also conduct a user study to verify the effectiveness of our proposed network. We recruited 50 volunteers for this user study. We randomly select 20 videos from DAVIS test set. For each video, the original video with object mask and the anonymized results from VINet, CPNet, FGNet and our model are presented to the volunteers. The volunteers are then asked to rank the four models with 1, 2, 3 and 4 (1 is the best and 4 is the worst) based on the perceptual quality of the inpainted videos. The result in terms of the percentage of rank scores received by different models is shown in Figure 7. Our model receives significantly more votes for rank 1 (the best) than the other three models, which verifies that our model can indeed generate more plausible results than existing models.

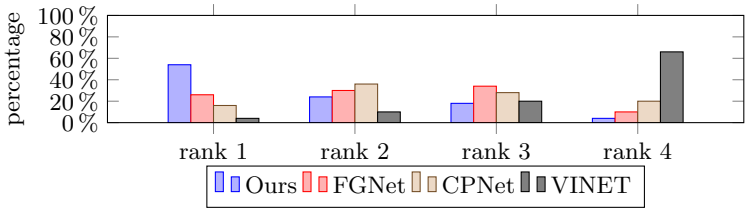

Fig. 7. User study results. For each rank (1 is the best), we collected 1000 votes (20 videos * 50 volunteers) in total. The y-axis indicates, within each rank, the percentage (out of 1000 notes) of the votes received by different models.

(a) Input (b) Ours (w/o BSCA) (c) Ours (scale $\frac{1}{8}$) (d) Ours

Fig. 8. Ablation study on BSCA. Better viewed at zoom level 400%.

4.4 Ablation Study

We investigate the effectiveness of the two components of our network: Boundary-aware Short-term Context Aggregation (BSCA) and Dynamic Long-term Context Aggregation (DLCA). In Table 2, we report all the quantitative results under different ablation settings on the DAVIS test set with shuffled object masks.

Effectiveness of BSCA. As we described in Sec. 3.2, the purpose of BSCA is to alleviate the negative effects of missing regions in the target frame. Table 2 compares our full model with its two variants: 1) "w/o BSCA" (the first row in Table 2), which removes the BSCA module and directly uses the flows to warp reference feature maps onto the target feature map as [11]. Then it concatenates the warped reference features with the target feature map as the input for attention-based aggregation; 2) "scale $\frac{1}{8}$" (the second row in Table 2), which performs the BSCA module only at the $\frac{1}{8}$ encoding scale. The performance drop of these two variants justifies the effectiveness of the BSCA module and the multi-scale design at the encoding stage. As we further show in Figure 8, the model without BSCA suffers from inaccurate feature alignment due to the occlusion effect of missing regions in the flows, thus producing distorted contents. Using BSCA only at the $\frac{1}{8}$ encoding scale apparently improves the results, but is still affected by the distortions from the previous scales. In contrast, using BSCA at multiple encoding scales lead to better results with temporally consistent details.

Table 2. Comparisons of different settings on BSCA and DLCA.

BSCA (scale $\frac{1}{8}$)	BSCA	DLCA (fixed)	DLCA	PSNR	SSIM	VFID
			✓	27.17	0.847	0.063
✓			✓	27.72	0.859	0.052
	✓			27.58	0.858	0.056
	✓	✓		27.85	0.862	0.048
	✓		✓	**28.13**	**0.867**	**0.042**

Effectiveness of DLCA. We test two other variants of our full model regarding the DLCA module: 1) "w/o DLCA" (the third row in Table 2), which directly removes the DLCA module; and 2) "fixed" (the fourth row in Table 2), which keeps the DLCA module but uses fixedly sampled reference features (rather than dynamic updated ones) that takes one frame for every five frames out of the entire input video sequence. Both variants exhibit performance degradation. Although

(a) Input (at time t) | (b) Ours (w/o DLCA) | (c) Ours (Fixed) | (d) Ours | (e) $\hat{Y}_{t-13}$ (not in V) | (f) $\hat{Y}_{t-25}$ (in V)

Fig. 9. Ablation study on DLCA. Better viewed at zoom level 400%.

fixedly sampled reference features can help, it is still less effective than using our dynamically updated long-term features. As shown in Figure 9, the model without DLCA module (w/o DLCA) fails to recover the background after object removal due to the lack of long-term frame information. Although the model with fixedly sampled reference features successfully restores the background, blurriness and artifacts can be still be found. Figure 9 (e) and Figure 9 (f) further illustrate the dynamic characteristic of our dynamic updating rule, which can effectively avoid irrelevant reference frames (eg., $\hat{Y}_{t-13}$ is not in our long-term feature set V) for the current target frame.

Dynamically Updated Long-term Features. We investigate the impact of different lengths of dynamically updated long-term features V on the inpainting results. Small lengths of q is insufficient to capture long-term frame information, resulting in inferior performance. On the contrary, large lengths will include more irrelevant reference frames, which also leads to degraded performance. The best result is achieved at length $q = 10$. For the parameter r (long-term range), we empirically find that $r = 9$ works well across different settings.

5 Conclusion

We studied the problem of video inpainting and addressed three limitations of existing methods: 1) ineffective usage of short-term or long-term reference frames; 2) inaccurate short-term context aggregation caused by missing regions in the target frame; and 3) fixed sampling of long-term contextual information. We therefore proposed a Short-term, and Long-term Context Aggregation Network with two complementary modules for the effective exploitation of both short-term and long-term information. We have empirically demonstrated the effectiveness of our proposed approach on benchmark datasets and provided a comprehensive understanding of each module of our model.

Acknowledgement

This research was supported by Australian Research Council Projects FL-170100117, IH-180100002, IC-190100031, LE-200100049.

References

1. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Transactions on Graphics (2009)
2. Bertalmio, M., Vese, L., Sapiro, G., Osher, S.: Simultaneous structure and texture image inpainting. IEEE Transactions on Image Processing (2003)
3. Chang, Y.L., Liu, Z.Y., Hsu, W.: Free-form video inpainting with 3d gated convolution and temporal patchgan. ICCV (2019)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)
5. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR (2016)
6. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014)
7. Huang, J.B., Kang, S.B., Ahuja, N., Kopf, J.: Temporally coherent completion of dynamic video. ACM Transactions on Graphics (TOG) **35**(6), 196 (2016)
8. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. ACM Transactions on Graphics (TOG) (2017)
9. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: CVPR (2017)
10. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016)
11. Kim, D., Woo, S., Lee, J.Y., Kweon, I.S.: Deep video inpainting. In: CVPR (2019)
12. Lai, W.S., Huang, J.B., Wang, O., Shechtman, E., Yumer, E., Yang, M.H.: Learning blind video temporal consistency. In: ECCV (2018)
13. Lee, S., Oh, S.W., Won, D., Kim, S.J.: Copy-and-paste networks for deep video inpainting. ICCV (2019)
14. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: ECCV (2018)
15. Newson, A., Almansa, A., Fradet, M., Gousseau, Y., Pérez, P.: Video inpainting of complex scenes. SIAM Journal on Imaging Sciences **7**(4), 1993–2019 (2014)
16. Oh, S.W., Lee, S., Lee, J.Y., Kim, S.J.: Onion-peel networks for deep video completion. ICCV (2019)
17. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR (2016)
18. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: CVPR (2016)
19. Sagong, M.c., Shin, Y.g., Kim, S.w., Park, S., Ko, S.j.: Pepsi : Fast image inpainting with parallel decoding network. In: CVPR (2019)
20. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
21. Wang, C., Huang, H., Han, X., Wang, J.: Video inpainting by jointly learning temporal structure and spatial details. In: AAAI (2019)
22. Wang, T.C., Liu, M.Y., Zhu, J.Y., Liu, G., Tao, A., Kautz, J., Catanzaro, B.: Video-to-video synthesis. arXiv preprint arXiv:1808.06601 (2018)

23. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: CVPR (2018)
24. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: CVPR (2004)
25. Xiong, W., Yu, J., Lin, Z., Yang, J., Lu, X., Barnes, C., Luo, J.: Foreground-aware image inpainting. In: CVPR (2019)
26. Xu, N., Yang, L., Fan, Y., Yang, J., Yue, D., Liang, Y., Price, B., Cohen, S., Huang, T.: Youtube-vos: Sequence-to-sequence video object segmentation. In: ECCV (2018)
27. Xu, R., Li, X., Zhou, B., Loy, C.C.: Deep flow-guided video inpainting. CVPR (2019)
28. Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., Li, H.: High-resolution image inpainting using multi-scale neural patch synthesis. In: CVPR (2017)
29. Yeh, R.A., Chen, C., Lim, T.Y., Schwing, A.G., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with deep generative models. In: CVPR (2017)
30. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: CVPR (2018)
31. Zeng, Y., Fu, J., Chao, H., Guo, B.: Learning pyramid-context encoder network for high-quality image inpainting. In: CVPR (2019)
32. Zhang, H., Mai, L., Xu, N., Wang, Z., Collomosse, J., Jin, H.: An internal learning approach to video inpainting. In: CVPR (2019)