Frame-Recurrent Video Inpainting by Robust Optical Flow Inference

Yifan Ding[1] Chuan Wang[2] Haibin Huang[2] Jiaming Liu[2] Jue Wang[2] Liqiang Wang[1]

[1]University of Central Florida

yf.ding@knights.ucf.edu, lwang@cs.ucf.edu

[2]Megvii Technology

{wangchuan, huanghaibin, liujiaming, wangjue}@megvii.com

Abstract

In this paper, we present a new inpainting framework for recovering missing regions of video frames. Compared with image inpainting, performing this task on video presents new challenges such as how to preserving temporal consistency and spatial details, as well as how to handle arbitrary input video size and length fast and efficiently. Towards this end, we propose a novel deep learning architecture which incorporates ConvLSTM and optical flow for modeling the spatial-temporal consistency in videos. It also saves much computational resource such that our method can handle videos with larger frame size and arbitrary length streamingly in real-time. Furthermore, to generate an accurate optical flow from corrupted frames, we propose a robust flow generation module, where two sources of flows are fed and a flow blending network is trained to fuse them. We conduct extensive experiments to evaluate our method in various scenarios and different datasets, both qualitatively and quantitatively. The experimental results demonstrate the superior of our method compared with the state-of-the-art inpainting approaches.

1. Introduction

Video creating and editing are becoming increasingly popular nowadays due to rapid development of Internet. However, it remains a big challenge for the task so called video completion, i.e. inpainting, which aims at filling missing regions caused by unwanted object removal or corruption of video data in transmission, and recovering plausible frames. Due to the additional temporal dimension, directly applying existing image inpainting techniques frame by frame is commonly problematic because it lacks the preservation of inter-frame consistency of the input video and suffers from flickering artifacts .

With the development of deep neural network in video tasks [33, 19, 20], early trial of extending 2D image inpainting methods into video inpainting via 3DCNNs is proposed in [31]. In that work, a 3DCNN module is introduced to recover the inter-frame temporal coherence and used as a guidance for a 2DCNN module which generates high-quality frames and outputs recovered videos. However, due to the memory limits and high computational cost of 3DCNN, that approach can deal with videos of fixed-size only and a streaming adaption of 3DCNN is far from straight-forward. 3DCNN also suffers from its limited capability of recovering motions in video because the kernel size in 3DCNN limits the motion range it models. These drawbacks disable it being further applied to a variety of video data.

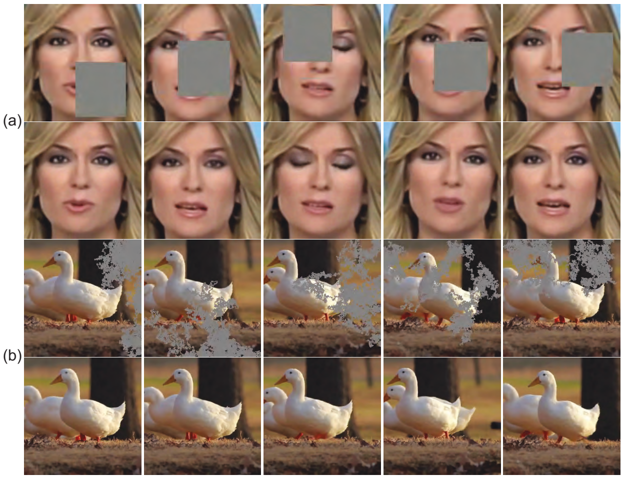

Figure 1. Video inpainting results by our method. Row 1 and 3 are consecutive input frames with random rectangle and random walker masks. Row 2 and 4 are our results which contain rich visual details and are temporally consistent.

In this work, we tackle the problem of video inpainting by solving the following issues: preserving temporal consistency and spatial details as well as handling arbitrary input video size and length efficiently. Inspired by [15, 18, 5], we propose a ConvLSTM based video inpainting method which uses image based algorithms to model spatial information intra-frame and recurrent neural networks to model temporal information inter frame with optical flow as an intermediary. Incorporating these two modules, our model circumvents the memory problems brought by 3DCNN and can handle videos with larger motion thanks to the optical flow which is not constrained by the kernel size.

A problem left to be solved in our model is that optical flow is hard to gain region in holes inside the frames. In our model, we incorporate a robust optical flow generation module to obtain a trustable optical flow to guide the output of an image inpainting algorithm the ConvLSTM. In this module, two sources of initially recovered optical flows, one by the inpainted frames and the other by the inpainted optical flow, are fused together by a blending network. The output flow commonly has higher accuracy which enables larger motion prediction. The ConvLSTM based network further models the temporal information within the inpainted frames and gradually remove the inter-frame flickers through a loss considering the temporal and spatial balance.

To verify the effectiveness of the network, we tested our model on two datasets, the FaceForensics [24] of faces with moving emotion expressions and the DAVIS+VIDEVO [15] being made of moving objects on natural backgrounds. The testing involves three types of masks, fixed rectangles, random rectangles or a more complicated random walker masks. Experimental results reveal the superior of our method than state-of-the-art. Ablation studies also demonstrate the ability of the ConvLSTM module to model temporal information as well as the robustness of the optical flow generation module.

In summary, our main contributions include:

- A ConvLSTM based video inpainting network which combines the temporal and spatial constraints to reconstruct videos with details but without flickering artifacts.
- A robust motion estimation network via trainable optical flow fusion module which enables the prediction of large motions so as to play a key role in the success of our method.
- State-of-the-art performance in the video inpainting task, both in quality and efficiency under three types of masks. Our method is able to handle videos of arbitrary length and frame size, which is a limitation of previous work.

2. Related Work

Patch-based image/video inpainting. Patch-based synthesis is the most used traditional strategy for image inpainting. It was firstly proposed in [4] and later improved by works [14, 35, 1, 2]. The basic idea is to recover the missing contents in a region-growing way, i.e. the algorithms start from boundary of holes and extend the region by searching appropriate patches and assembling them together. The improved methods work on different directions in searching and optimization, or for application like face [41, 39]. It is also adapted to video inpainting problem by replacing 2D patch synthesis with 3D spatial-temporal patch synthesis across frames. This was firstly proposed in [34, 35] to ensure the temporal consistency of the generated video and later improved in [11, 29] to handle more complicated video input. However, all of these works are designed for the video with repeated content across frames. They are unable to tackle the problem we proposed in this paper where missing parts cannot be replaced by similar content in the input. Resorting to a large video dataset, we try to train a ConvLSTM in this work for missing contents prediction based on the high-level spatial-temporal context understanding.

CNN-based image/video inpainting. Recently, Convolutional Neural Network was applied to image inpainting for small holes [36] and then extended to larger missiong regions [21]. Later on, Yang et al. [40] further developed a multi-scale neural patch synthesis algorithm that not only preserves contextual structure but also produces high-frequency details. The algorithm proposed in [9] further enhances the performance by involving two adversarial losses to measure both the global and local consistency of the result. Different from the previous works which only focus on box-shaped holes, This method also develops a strategy to handle the holes with arbitrary shapes. Liu et al. further solve the inpainting problem by introducing a partial convolution operator [17], to better handle challenging holes generated by a random walker.

Wang et al. extended the CNN solution from image to video, proposing a hybrid network combining 2DCNN and 3DCNN. They used 3DCNN to recover the temporal coherence and 2DCNN to reconstruct the spatial details. However, due to the heavy computational cost and limited power to learn temporal information by 3DCNN, this method works only for videos with limited frame size and length. It cannot handle videos with medium motion either. On the contrary, our method removes the restriction of video length and frame size, and is able to better model the temporal consistency by a ConvLSTM module and a robust flow generation network.

Learning temporal consistency by LSTM. It has been well demonstrated in recent years that recurrent neural network (RNN), or its variants Long Short-Term Memory (LSTM) [8, 6, 22] and ConvLSTM [37] are powerful tools to model long-term temporal consistency for video related tasks, e.g. video captioning [5] and action recognition [18]. The power of ConvLSTM results from its self-parameterized controlling gates which can decide a memory

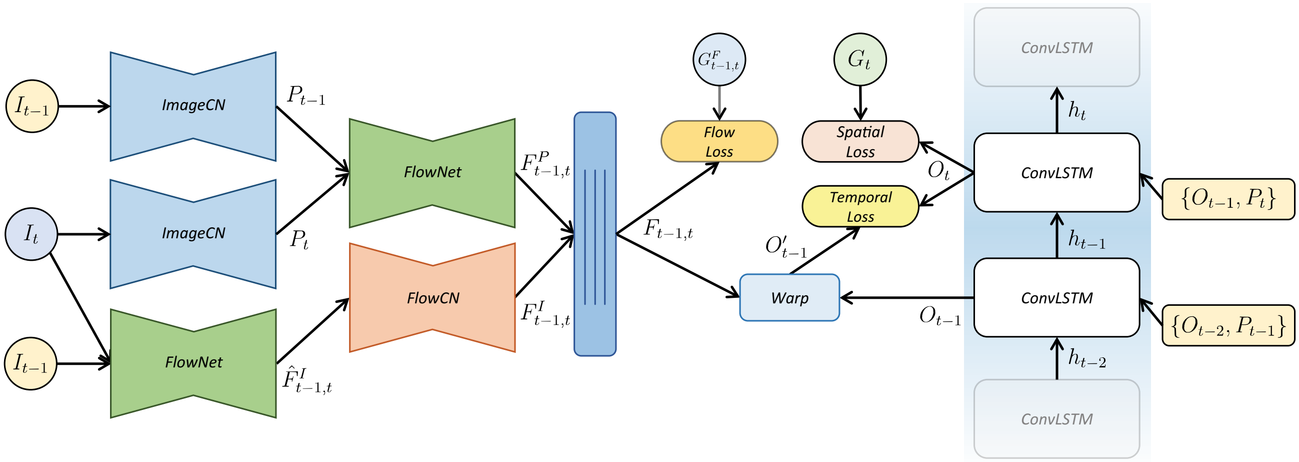

Figure 2. Network structure. $\{I_t\}$ denotes the input frames with holes inside. $\{P_t\}$ denotes the inpainted frames using image inpainting method. $\{F^P\}$ and $\{F^I\}$ are the two branch optical flow generated from $\{P_t\}$ and $\{I_t\}$ separately. $\{F_t\}$ is the final output of the Blending network which is illustrated in Figure 3

cell "remember" or "forget" the present and past information, which is commonly formulated as the following equations:

$$\begin{aligned}
i_t &= \sigma(W_{xi} \circledast X_t + W_{hi} \circledast H_{t-1} + W_{ci} \circ C_{t-1} + b_i) \\
f_t &= \sigma(W_{xf} \circledast X_t + W_{hf} \circledast H_{t-1} + W_{cf} \circ C_{t-1} + b_f) \\
c_t &= f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} \circledast X_t + W_{hc} \circledast H_{t-1} + b_c) \\
o_t &= \sigma(W_{xo} \circledast X_t + W_{ho} \circledast H_{t-1} + W_{co} \circ C_t + b_o) \\
H_t &= o_t \circ \tanh(C_t)
\end{aligned}$$

where $\circ$ is the Hadamard product and $\circledast$ is the convolution operator.

In [37], the capacity of ConvLSTM is well demonstrated by the task of weather forecast, where spatial-temporal information in satellite images is modeled by the combination of convolution layers and LSTM. Additionally, optical flow conveys motion information which can propagate the results in previous frame into the current one. It usually serves as an assistant to the ConvLSTM based tasks, e.g. video segmentation [20, 28], action recognition [3, 16] and detection [42]. Following the similar idea, in this paper we also apply ConvLSTM and optical flow to video inpainting task, to demonstrate their effectiveness.

3. Algorithm

Overview. Our method is built upon deep neural networks, including CNN and ConvLSTM. It streamingly takes two successive frames $\{I_{t-1}, I_t\}$ of an incomplete video V_{in} as input, and produces a complete frame O_t as output, where O_t contains spatial details and holds temporal consistency with last time output, O_{t-1}.

To achieve the two goals simultaneously, we design a framework composed of three functional parts. The first part is an image-based inpainting module $\mathcal{H}_s$, produces an inpainted frame P_t given I_t, i.e.

$$P_t = \mathcal{H}_s(I_t)$$

which is achieved by an existing image inpainting algorithms such as Partial Convolution [17] which is used in this paper, where the produced frames $\{P_t\}$ normally contain flickering artifacts. Then we recurrently feed a pair of results $\{O_{t-1}, P_t\}$ to a ConvLSTM module $\mathcal{H}_t$, to produce the output O_t of the current time stamp t, which is introduced in details in Section 3.2. To ensure O_t to be visually plausible to its ground truth G_t and coherent with O_{t-1}, we involve newly designed losses and adopt several training strategies (Section 3.3), which rely on a robust optical flow generation module $\mathcal{H}_f$ as introduced in Section 3.1. To give an overview, we illustrate the entire pipeline of our framework in Figure 2.

3.1. Robust Flow Generation

Optical flow plays a key role in video related tasks as it provides direct access to the temporal information. However, with holes in the frames, estimating the optical flow becomes far from straight-forward for video inpainting task. To tackle this issue, we consider two separated paths to generate a robust optical flow which we can rely on for the output temporal consistency.

We manage to consider two sources of the optical flow. One branch for the optical flow, i.e. $F^P_{t-1,t}$, is generated from the separately inpainted frames $\{P_t\}$ by an optical flow estimation module $\mathcal{F}_l$, i.e.

$$F^P_{t-1,t} = \mathcal{F}_l(P_{t-1}, P_t)$$

The other branch, i.e. $F^I_{t-1,t}$, results from the completion of defective optical flow $\hat{F}^I_{t-1,t}$ generated from the input

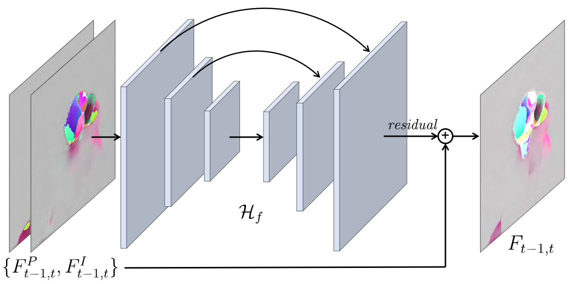

Figure 3. Structure of Flow Blending Network. We use a three layers encoder-decoder architecture and the feature maps from the encoder are concatenated to the corresponding decoder layers to improve the performance. The input flow pairs are stacked channel-wisely.

frames $\{I_t\}$, by an optical flow completion network $\mathcal{H}_c$, i.e.

$$F^I_{t-1,t} = \mathcal{H}_c(\hat{F}^I_{t-1,t}), \qquad \hat{F}^I_{t-1,t} = \mathcal{F}_l(I_{t-1}, I_t)$$

Coming from different paths, $F^I_{t-1,t}$ and $F^P_{t-1,t}$ normally have various distributions. Specifically, $F^P_{t-1,t}$ looks smoother but sometimes brings in artifacts that may mislead ConvLSTM to correct the inpainting results. On the contrary, $F^I_{t-1,t}$ accords to the statistical characteristics of the optical flow in the training dataset, but it occasionally fails to be well filled and a border is usually shown in the inpainted optical flow. To take advantage of the two flows, we design a Flow Blending Network $\mathcal{H}_f$ to generate a more robust flow.

Flow blending network $\mathcal{H}_f$. We propose Flow Blending Network $\mathcal{H}_f$ to blend the two optical flows so as to remove the errors. Its structure is illustrated in Figure 3, where the input is a channel-wisely stacked version of $\{F^P_{t-1,t}, F^I_{t-1,t}\}$ and the output is a refined flow $F_{t-1,t}$. $\mathcal{H}_f$ adapts an U-Net [23] architecture where the extracted features from the encoder are concatenated to the corresponding decoder layers to enhance the performance. To produce $F_{t-1,t}$, a residual value is predicted and added to $F^P_{t-1,t}$ and $F^I_{t-1,t}$, i.e.

$$F_{t-1,t} = \frac{1}{2}\left(F^P_{t-1,t} + F^I_{t-1,t} + \mathcal{H}_f(F^P_{t-1,t}, F^I_{t-1,t})\right) \quad (1)$$

Note that, there is still other options for the design of such a flow blending network. For example, the attention mechanism [38] uses fully connected layers to weight the feature maps, providing more flexibility. However, due to large number of parameters involved, this method requires more computational resources which potentially limits the efficiency for video inpainting. On the contrary, with only 6 convolutional layers (3 for the encoder and 3 for the decoder), our flow blending network significantly limits the number of parameters, making it invariant with the size of the feature maps. Together with the ConvLSTM module which is also built upon FCN, our network can accept arbitrary size of frames theoretically.

With the ground truth optical flow $G^F_{t-1,t}$ as a supervision, our flow blending network $\mathcal{H}_f$ can learn an optimized blending mechanism for the two source optical flows. This process can be written as:

$$\min_{\mathcal{H}_f} L_f = \sum_{t=1}^{T} ||G^F_{t-1,t} - F_{t-1,t}||_1$$

where L_f is the sum of l_1 loss between $G^F_{t-1,t}$ and $F_{t-1,t}$ over all T time steps. We demonstrate the effectiveness of $\mathcal{H}_f$ in Table 3 and Paragraph 4.3.

Optical flow completion network $\mathcal{H}_c$. We follow a similar idea to inpaint the optical flow by designing an optical flow completion network. The modification lies in that, unlike imagery data, optical flow has different value range and channel numbers, so that special training schemes have to be designed for it. To be specific, we first normalize the optical flow data by an instance-wise value range instead of a constant range as usually adopted for RGB image data. Additionally, we transform the two-channel flow data to three channels to utilize the pre-trained weights, where the third channel is filled with the mean value of the first two channels.

3.2. Recurrent Neural Network

With the single inpainted frames $\{P_t\}$ and the blended optical flow $\{F_{t-1,t}\}$ computed, we further apply Recurrent neural network [25] to enforce the temporal consistency and remove inter-frame flickers in $\{P_t\}$.

Specifically, we adopt ConvLSTM to capture the spatial-temporal information in the videos. We first pass O_{t-1} and P_t through several convolutional layers separately, and then concatenate their extracted features and feed them to the ConvLSTM module $\mathcal{H}_t$. Finally, the output state of the ConvLSTM is decoded to the residual value between O_{t-1} and O_t, i.e.

$$O_t = O_{t-1} + \mathcal{H}_t(O_{t-1}, P_t) \quad (2)$$

With ConvLSTM involved, the temporal correlation between two successive frames are well modeled. Since it does not restrict the length of the input sequences, our method can work in a streaming manner unlike [31]. Furthermore, with fully convolutional modules inside, the spatial details can be also reconstructed to produce accurate and smooth frames.

3.3. Training Losses and Strategy

Spatial losses. We compute the l_1 distance between the output frames $\{O_t\}$ and the ground truth frames $\{G_t\}$ to guide our model spatially reconstructing the frames. Besides the per-pixel loss, we also incorporate perceptual losses [12] between $\{O_t\}$ and $\{G_t\}$. Specially, the two losses are

$$L_d = \sum_{t=1}^{T} M_t \cdot ||O_t - G_t||_1 \tag{3}$$

$$L_p = \frac{1}{N} \sum_{l=1}^{N} \sum_{t=1}^{T} ||\mathcal{H}_p(O_t) - \mathcal{H}_p(G_t)||_1 \tag{4}$$

where $\mathcal{H}_p$ is a pre-trained feature extractor such as ResNet [7] or VGG-16 [26] as used in this paper, and N is the number of feature maps of $\mathcal{H}_p$.

Short-term temporal losses. The temporal losses are proposed to maintain the temporal consistency between consecutive output frames. We first compute the l_1 loss between O_t and O'_{t-1}, a warped version of O_{t-1}, to enforce the transition from O_{t-1} to O_t as smooth as possible. Specifically, it is calculated as

$$L_s = \sum_{t=2}^{T} M_t \cdot ||O_t - O'_{t-1}||_1 \tag{5}$$

where M_t is a mask representing the missing pixels in the input frame, i.e. regions with $M_t = 1$ need to be recovered. The warping operation is achieved by re-mapping the pixel values of O_{t-1} by flow $F_{t-1,t}$, i.e.

$$O'_{t-1} = Warp(O_{t-1}, F_{t-1,t}) \tag{6}$$

which naturally supports gradient back-propagation.

L_s encodes a short-term temporal loss which involves two consecutive frames in the forward direction. In practice, we also apply the aforementioned process to the reverse version of the input video, producing another short-term temporal loss, i.e.

$$L_r = \sum_{t=2}^{T} M_t \cdot ||O_{t-1} - O'_t||_1 \tag{7}$$

With bi-directional temporal losses involved, the results could be potentially optimized in a more stable and rarely error-prone manner.

Long-term temporal losses. We also calculate a long-term temporal loss between O'_t and O_1 and O_T, respectively, to maintain the consistency of the whole video as well rather than just two consecutive frames. Similarly, it is written as

$$L_l = \sum_{t=1}^{T} M_t \cdot (||O_1 - O'_t||_1 + ||O_T - O'_t||_1) \tag{8}$$

	FaceForensics		DAVIS+VIDEVO	
	CombCN [31]	Ours	CombCN [31]	Ours
Fixed Rectangles	13.68	**12.03**	37.63	**23.38**
Random Rectangles	5.84	**4.01**	42.41	**12.27**
Random Walker [27]	5.07	**2.25**	42.57	**6.01**
Average	8.20	**6.10**	40.87	**13.89**

Table 1. l_1 loss on dataset FaceForensics and DAVIS+VIDEVO.

Model	CombCN [31]	Ours
Inference Time (ms/frame)	36.06	**23.31** (35.35% less)
No. of Parameters	15,489,539	**2,947,011** (80.97% less)
Input Length	fixed	arbitrary
Input Resolution	128	arbitrary*

Table 2. Comparison in time and memory efficiency between CombCN and our method. * means our model is fully convolutional and not restricted to the input size.

Training. During training, we weight and sum up the aforementioned losses for a global optimization, covering temporal and spatial ones. Specifically, the overall loss of our model is defined as:

$$\mathbf{L} = \sum_{i \in U} \lambda_i \cdot L_i, \quad U = \{s, f, d, p, r, l\} \tag{9}$$

where λ_i denotes the weight of the corresponding loss term L_i, which is manually pre-defined in the experiments. In practice, to facilitate the optimization, we first pre-train the image inpainting and flow inpainting modules to a stable status, and then train the ConvLSTM and the Flow Blending Network without finetuning the two inpainting modules. With all of the training losses and strategies as proposed, our network is able to inpaint frames with plausible details but without flickering artifacts.

4. Experimental Results

4.1. Dataset and Experimental Settings

We test our network on two datasets and three types of masks. FaceForensics [24] is a human face dataset containing 1,004 video clips with near frontal pose and neutral expression changed across frames. To fully excavate the potential of our framework, we also test on the DAVIS+VIDEVO dataset [15] which has 190 videos and contains a variety of moving objects and motion types. The three types of masks include:

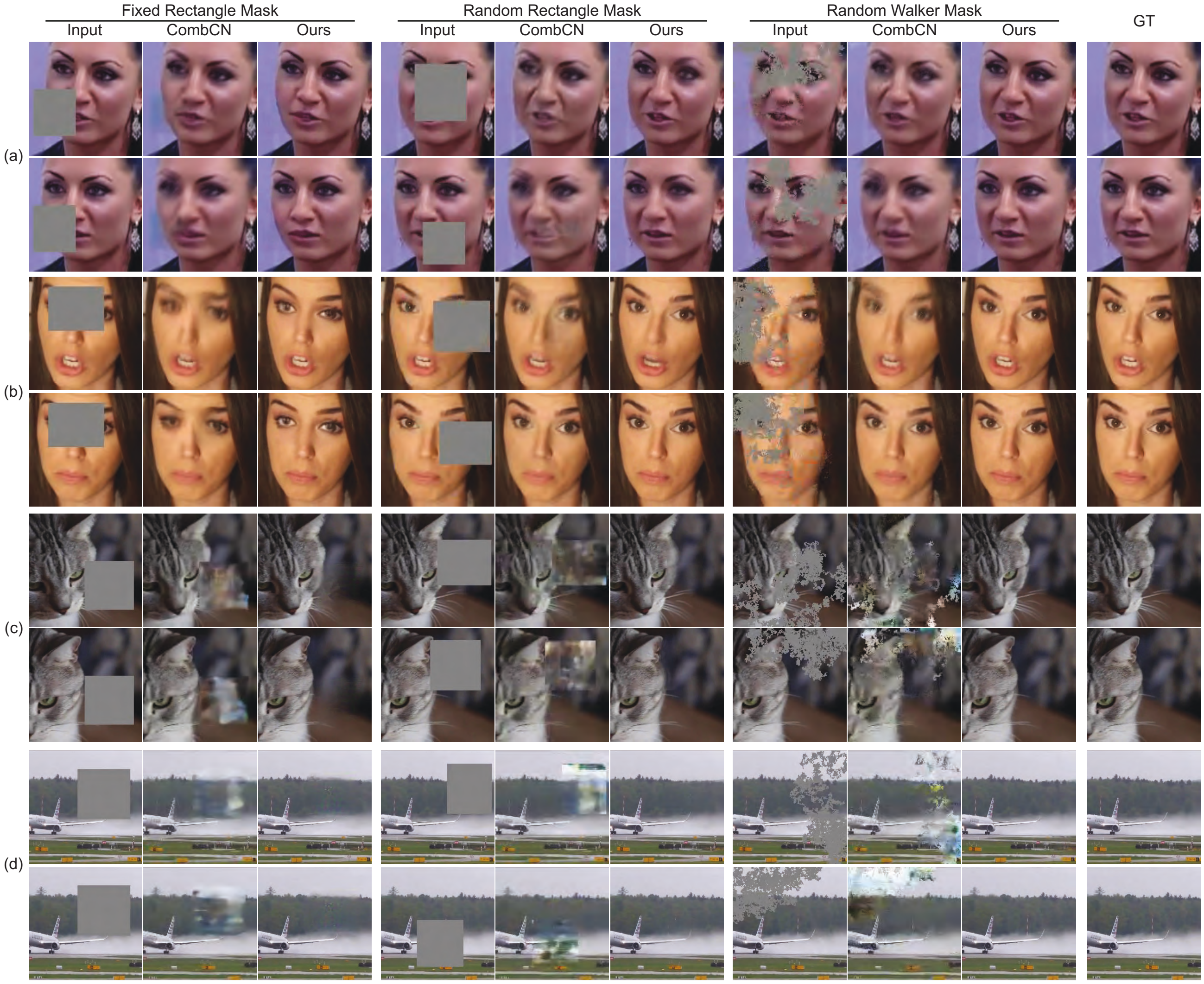

Figure 4. Results: The first 2 samples are from FaceForensics [24] and the rest are from DAVIS+VIDEVO [15]. The rows shows different frames in a video, and the column shows the results comparison under different masks. Checking the columns marked as "CombCN" and "Ours", it can be found that our results outperform CombCN in all mask and dataset combinations, especially on the complex natuaral dataset DAVIS+VIDEVO [15] and random masks.

1) Fixed rectangles: the rectangle masks are the same across all frames in a video;
2) Random rectangles: each frame in a video have rectangle mask of changing size and locations;
3) Random walker: the masks have random streaks and holes of arbitrary shapes as in [27]

For the two rectangle masks, we follow the setting in [31] which generates masks of size between $[0.375l, 0.5l]$, where l is the frame size.

In all experiments, we use FlowNet2 [10] for online optical flow generation. As mentioned, we also pre-train the frame inpainting network $\mathcal{H}_s$ and the flow inpainting network $\mathcal{H}_c$ to boost the training process. We use Adam [13] as the optimizer and set the learning rate to be 1.0×10^{-4}. For the DAVIS+VIDEVO dataset [15], random crop and rotation are used as standard data augmentation operations, while for the FaceForensics [24] dataset, only center crop is applied. As for the weights of each loss, we set empirically $\{\lambda_s, \lambda_d, \lambda_r\} : \{\lambda_f, \lambda_p, \lambda_l\} = 10 : 1$.

4.2. Comparison with Existing Methods

Result quality. We compare our method with [31], the only deep video inpainting solution to our best knowledge, both quantitatively and qualitatively, as shown in Table 1 and Figures 4 and 1 respectively. We also include more video results in our accompanying supplemental materials. Table 1 compares our model with [31] in terms of

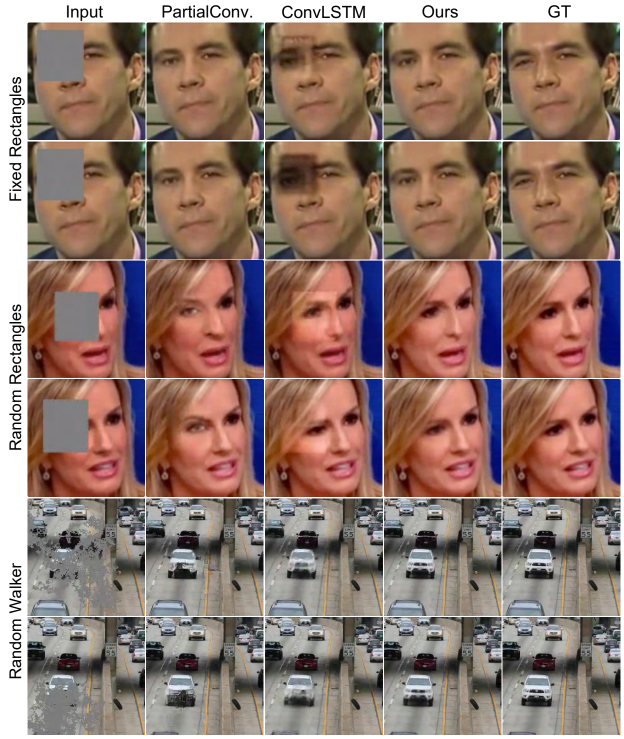

Figure 5. Ablation study results. We list 3 samples, each of which contains 2 frames from top to bottom rows, under the conditions of fixed rectangles, random rectangles and random walker masks, respectively. In the 2nd and 3rd columns, we illustrate the results by PartialConv-only and ConvLSTM-only modules. Compared with ours, these two models produce low quality results in terms of time consistency and spatial details.

the average l_1 difference on all validation video frames. It can be found that our statistic data are better among three mask types and two datasets. It also reveals that the quality gap between [31] and us is larger on the natural dataset DAVIS+VIDEVO with larger motion, where the average l_1 loss is 40.87 and 13.89, in comparison with the values 8.2 and 6.1 for the FaceForensics dataset [24]. This phenomenon potentially results from the fact that our model is designed to work better on large motion videos due to ConvLSTM and optical flow as involved, while 3DCNN in [31], has very limited capability to capture motion. Furthermore, the quality gap on random walker masks between [31] and ours is larger than those on the two rectangle masks, which demonstrates that our model can better deal with masks with various shapes. The last but not the least, it is worth noting that with one single Titan Xp GPU, our model can deal with videos of arbitrary length, and of frame size as high as 256p in real time[1], in comparison with [31] which can handle videos with fixed length of 32 frames and size of 128p only.

As for qualitative results, it is also obvious that our results outperform [31] as Figure 4 shows. The difference between our results and [31] on FaceForensics mainly lies in the sharpness of the results. With the 3DCNN as a temporal inference approach, it seems pixels from multiple frames are blended without being aware of the location movement. As for the results of DAVIS+VIDEVO, [31] barely fills in reasonable patches, making the border easily noticeable in contrary with ours.

Computational efficiency. We also include several other metrics such as the time and memory efficiency to compare [31]. We calculate the average inference time per video and the number of trainable parameters in both models. As seen from Table 2, without 3DCNN module involved, our model achieves better time and memory efficiency compared with [31], i.e. our inference time and number of parameters is only around $2/3$ and $1/5$ of CombCN. Also thanks to the fully convolutional architecture of the model, our approach can accept videos of larger frame size and arbitrary length.

4.3. Ablation Studies

We further conduct the following ablation study to discover to influence of each part in our network towards the final results.

Ours vs. PartialConv. We first compare our results $\{O_t\}$ with those produced by PartialConv [17] frame-wisely, i.e. $\{P_t\}$. The statistic data is listed in the Columns "PartialConv" and "Ours" in Table 3 and some samples are shown in Figure 5 accordingly. As seen, our approach achieves higher quality compared with PartialConv-only method, especially for the cases of random rectangle and random walker masks. This is due to the fact that by using ConvLSTM, our method can model the temporal coherence so that information from adjacent frames could be learnt. Even though a region is damaged in the current frame, similar patches could be found in its adjacent frames due to random masks moving away. Meanwhile, our method better improves the temporal coherence and reconstructed details in DAVIS+VIDEVO than FaceForensics dataset, which lacks enough motion and diversity. This phenomenon demonstrates the effectiveness of ConvLSTM module, which enables our method to preserve the temporal consistency across frames and to be specialized in inpainting videos with more motion.

Ours vs. ConvLSTM. We further validate the necessity of inpainting all frames by frame-wise inpainting module $\mathcal{H}_s$. Specifically, we instead inpaint the first and last frame only in a video by PartialConv, producing $P_t =$

[1] For fair comparison, we limit the test videos to be of 32 frames and 128p for our method in the experiments.

	FaceForensics [24]					DAVIS+VIDEVO [15]				
	PartialConv	ConvLSTM	$F^P_{t-1,t}$	$F^I_{t-1,t}$	Ours	PartialConv	ConvLSTM	$F^P_{t-1,t}$	$F^I_{t-1,t}$	Ours
Fixed Rectangles	11.59	38.08	**11.54**	12.22	12.03	24.74	53.10	24.94	27.30	**23.38**
Random Rectangles	11.28	22.68	4.13	4.73	**4.01**	24.99	17.80	12.80	13.07	**12.27**
Random Walker	3.31	3.75	2.35	2.75	**2.25**	10.67	10.00	7.74	7.39	**6.01**

Table 3. l_1 losses of results in ablation study for datasets FaceForensics [24] and DAVIS+VIDEVO [15]. The ablation study include only using PartialConv, ConvLSTM, flow $F^P_{t-1,t}$ or $F^I_{t-1,t}$.

$\mathcal{H}_s(I_t), t \in \{1, T\}$ and keep the rest frames untouched. We then feed the sequence $\{P_1, I_2, I_3, ..., I_{T-1}, P_T\}$ to the ConvLSTM module for training and validation, and show the results in Table 3 and Figure 5. As seen, without the PartialConv module, ConvLSTM produces low-quality results compared with ours, especially for fixed rectangle masks. The reason behind is that, fixed rectangle masks make the missing regions constant across frames, in which case ConvLSTM has very limited information from adjacent frames to inpaint the current frame.

From the two comparisons as above, we see that our method generally outperforms each single module of the entire model as proposed. With PartialConv only, the produced results are of satisfied quality for a single frame, yet look incompatible when being played, lacking of stable temporal coherence. On the other hand, with ConvLSTM involved only, temporal information from the adjacent frames could be recovered, but details are hardly reconstructed, causing the filled region inconsistent with the content of the present frame. This is due to that the strength of ConvLSTM is on temporal information recovery but it lacks enough capability of modeling spatial details. Our approach naturally combine the advantages of the two modules, so that benefits from the strengths of both of them.

Disabling the flow blending network $\mathcal{H}_f$. To verify the effectiveness of the flow blending network $\mathcal{H}_f$ as proposed, we alternatively disable $\mathcal{H}_f$ and set $F_{t-1,t}$ to $F^P_{t-1,t}$ or $F^I_{t-1,t}$ directly. For these two cases, we additionally list their l_1 losses in Table 3 to compare with ours. We see that without the flow blending network, the performance drops on almost all datasets and mask types. With the flow $F^I_{t-1,t}$ generated by the flow completion network $\mathcal{H}_c$, the misleading information in $F^P_{t-1,t}$ is corrected, which enables our approach to produce more robust inpainted frames. We also accompany the inpainted videos by using $F^I_{t-1,t}$ and $F^P_{t-1,t}$ only in our supplementary materials for better presentation, where more noticeable flickering artifacts exist.

5. Conclusion

We have presented a new video inpainting framework based on ConvLSTM and robust optical flow generation. Our framework can produce inpainted video frames with spatial details and temporal coherence. Unlike the previous volume based solutions [31, 30, 32], our method does not restrict the video length and frame size, being able to run in real-time streamingly, and can deal with large motions. These advantages lie in the strong capability of ConvLSTM in modeling spatial-temporal information simultaneously, and the rich motion information conveyed in the optical flow. To generate an accurate optical flow from frames with holes, we propose a robust flow generation module which can be fed with two sources of flows. A flow blending network is also proposed to learn the mechanism to fuse the two flows, producing results with less error. We further introduce three different masks and two datasets to thoroughly test our method. Experimental results demonstrate our superior performance in comparison with the state-of-the-art in all of the scenarios as above. Ablation studies are also conducted to reveal the effectiveness of different parts in our framework.

References

[1] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. In *ACM Transactions on Graphics (ToG)*, volume 28, page 24. ACM, 2009.

[2] C. Barnes, F.-L. Zhang, L. Lou, X. Wu, and S.-M. Hu. Patchtable: Efficient patch queries for large datasets and applications. *ACM Transactions on Graphics (TOG)*, 34(4):97, 2015.

[3] S. Das, M. Koperski, F. Bremond, and G. Francesca. Deep-temporal lstm for daily living action recognition. *arXiv preprint arXiv:1802.00421*, 2018.

[4] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1033–1038. IEEE, 1999.

[5] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen. Video captioning with attention-based lstm and semantic consistency. *IEEE Transactions on Multimedia*, 19(9):2045–2055, 2017.

[6] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with lstm. 1999.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and Locally Consistent Image Completion. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2017)*, 36(4):107:1–107:14, 2017.

[10] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2462–2470, 2017.

[11] Y.-T. Jia, S.-M. Hu, and R. R. Martin. Video completion using tracking and fragment merging. *The Visual Computer*, 21(8-10):601–610, 2005.

[12] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra. Texture optimization for example-based synthesis. In *ACM Transactions on Graphics (ToG)*, volume 24, pages 795–802. ACM, 2005.

[15] W.-S. Lai, J.-B. Huang, O. Wang, E. Shechtman, E. Yumer, and M.-H. Yang. Learning blind video temporal consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 170–185, 2018.

[16] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. Snoek. Videolstm convolves, attends and flows for action recognition. *Computer Vision and Image Understanding*, 166:41–50, 2018.

[17] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[18] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European Conference on Computer Vision*, pages 816–833. Springer, 2016.

[19] X. Meng, X. Deng, S. Zhu, S. Liu, C. Wang, C. Chen, and B. Zeng. Mganet: A robust model for quality enhancement of compressed video. *arXiv preprint arXiv:1811.09150*, 2018.

[20] D. Nilsson and C. Sminchisescu. Semantic video segmentation by gated recurrent flow propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6819–6828, 2018.

[21] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[22] J. Ren, Y. Hu, Y.-W. Tai, C. Wang, L. Xu, W. Sun, and Q. Yan. Look, listen and learnâĂŤa multimodal lstm for speaker identification. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[23] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[24] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179*, 2018.

[25] D. E. Rumelhart, G. E. Hinton, R. J. Williams, et al. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[27] N. Sundaram, T. Brox, and K. Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *European conference on computer vision*, pages 438–451. Springer, 2010.

[28] A. Terwilliger, G. Brazil, and X. Liu. Recurrent flow-guided semantic forecasting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1703–1712. IEEE, 2019.

[29] M. V. Venkatesh, S.-c. S. Cheung, and J. Zhao. Efficient object-based video inpainting. *Pattern Recognition Letters*, 30(2):168–179, 2009.

[30] C. Wang, Y. Guo, J. Zhu, L. Wang, and W. Wang. Video object co-segmentation via subspace clustering and quadratic pseudo-boolean optimization in an mrf framework. *IEEE Transactions on Multimedia*, 16(4):903–916, 2014.

[31] C. Wang, H. Huang, X. Han, and J. Wang. Video inpainting by jointly learning temporal structure and spatial details. *arXiv preprint arXiv:1806.08482*, 2018.

[32] C. Wang, J. Zhu, Y. Guo, and W. Wang. Video vectorization via tetrahedral remeshing. *IEEE Transactions on Image Processing*, 26(4):1833–1844, 2017.

[33] Y. Wang, H. Huang, C. Wang, T. He, J. Wang, and M. Hoai. Gif2video: Color dequantization and temporal interpolation of gif images. *arXiv preprint arXiv:1901.02840*, 2019.

[34] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE, 2004.

[35] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Transactions on pattern analysis and machine intelligence*, 29(3), 2007.

[36] J. Xie, L. Xu, and E. Chen. Image denoising and inpainting with deep neural networks. In *Advances in neural information processing systems*, pages 341–349, 2012.

[37] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[38] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[39] S. Yamaguchi, S. Saito, K. Nagano, Y. Zhao, W. Chen, K. Olszewski, S. Morishima, and H. Li. High-fidelity facial reflectance and geometry inference from an unconstrained image. *ACM Transactions on Graphics (TOG)*, 37(4):162, 2018.

[40] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 3, 2017.

[41] Y. Zhao, W. Chen, J. Xing, X. Li, Z. Bessinger, F. Liu, W. Zuo, and R. Yang. Identity preserving face completion for large ocular region occlusion. *British Machine Vision Conference (BMVC)*, 2018.

[42] K. Zhou, Y. Zhu, and Y. Zhao. A spatio-temporal deep architecture for surveillance event detection based on convlstm. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE, 2017.