Supplementary material: Inertia-Guided Flow Completion and Style Fusion for Video Inpainting

Kaidong Zhang[1] Jingjing Fu[2] Dong Liu[1]
[1] University of Science and Technology of China [2] Microsoft Research Asia
richu@mail.ustc.edu.cn, jifu@microsoft.com, dongeliu@ustc.edu.cn

1. Implementation Details

1.1. Network structure

We propose **I**nertia **G**uided **F**low **C**ompletion (IGFC) network to complete the optical flows, which are calculated using RAFT [7]. For the flow guided gradient warp stage, we adopt DeepFillV1 [10] to complete the occlusion regions, and we use our designed **A**daptive **S**tyle **F**usion **N**etwork (ASFN) to refine the style of the warped gradients for spatial coherence. We illustrate the detailed network structure of our proposed IGFC and ASFN in Tab 1 and Tab 2.

In IGFC, we adopt the matching network and the temporal modulation modules to aggregate the inertia guided aligned flow features. These two networks are adapted from [8]. Specifically, the matching net aims to generate the spatial adaptive attention maps between the target flow feature and the reference flow features. The temporal modulation component modulates the generated spatial adaptive attention maps along the temporal dimension. Given the features from the target flow F_t and the reference flows F_r, the fusion process can be denoted as:

$$\hat{F}_t = (1 - M_t) \odot F_t + M_t \odot \sum_{i=1}^{T} F_{r_i} A_{r_i} \tag{1}$$

where A_{r_i} represents the i-th attention map generated by the matching net and the temporal modulation module, while F_{r_i} represents the i-th feature map from F_r. M_t denotes the corresponding mask of F_t, and $\odot$ means the Hardmard product. We only update the corrupted regions of F_t to maintain the coherence in the valid regions.

1.2. Ternary census transform loss (TCT loss)

We adopt the TCT loss to supervise the completed optical flows based on the warping quality between the their corresponding images.

Suppose we restore the forward optical flow $\hat{F}_t$, which describes the motion field between ground truth frames I_t and I_{t+1}. We use the backward warping to warp I_{t+1} to I_t with the completed optical flow $\hat{F}_t$, and get the warped frame $I_{t+1\to t}$. Given the frames $I_{t+1\to t}$ and I_t, we first transform them into the gray images $g_{t+1\to t}$ and g_t. For each pixel in the gray images, we extract a $p \times p$ patch around the pixel and compare each surrounding pixel with the center. We illustrate such operation with g_t for simplicity, which can be formulated as,

$$\begin{aligned} T_t[i,j,:] &= \sqrt{(C(g_t[u,v] - g_t[i,j]))^2} \\ -\frac{p}{2} &\le u, v \le \frac{p}{2} \end{aligned} \tag{2}$$

T_t represents the census transformed image from g_t, while C represents the concatenation operation. If the shape of g_t is $H \times W$, the shape of the ternary census transformed image T_t will be $H \times W \times p^2$ (we pad g_t to make each point can be cropped into patches). We also apply such operation to $g_{t+1\to t}$ to get $T_{t+1\to t}$. After we calculate the ternary census transformed maps, we calculate the hamming distance between them.

$$D = \sum_{c=0}^{p^2} (T_{t+1\to t}[:,:,c] - T_t[:,:,c])^2 \tag{3}$$

Since we don't care about the warping in the valid regions, we utilize the corresponding mask M_t to filter out the invalid regions. To filter out the occlusion regions, we warp the frame I_{t+1} with the non-corrupted optical flow $F_{t\to t+1}$ and calculate the difference between the warped frame $I_{t+1\to t}$ and the frame I_t in order to locate the occlusion regions, which is calculated as,

$$m = exp(\alpha \cdot \sum (I_{t+1\to t} - I_t)^2) \tag{4}$$

where α is a negative constant, which indicates the magnificent to penalize the occlusion regions, and empirically set to -50. Combining the equation 2, 3 and 4, we calculate the TCT loss for all the completed flows under different resolutions to supervise the flow completion under multiple resolutions. We take the average of these losses to form L_{ter}. We illustrate the calculation in the original resolution

Module	Block	Filter size	In channels	Out channels	Stride/Up	Dilation
Encoder	2DConv	(5,5)	3	64	1	1
	2DConv	(3,3)	64	128	2↓	1
	2DConv	(3,3)	128	128	1	1
	2DConv	(3,3)	128	256	2↓	1
	ResBlocks×4	(3,3)	256	256	1	1
Matching Net	2DConv	(1,1)	256	64	1	1
	2DConv	(1,1)	256	64	1	1
	2DConv	(3,3)	128	192	1	1
	2DConv	(3,3)	192	256	2↓	1
	2DConv	(3,3)	256	256	1	1
	2DConv	(3,3)	256	128	1	1
	2DConv	(3,3)	128	64	2↑	1
	2DConv	(3,3)	64	1	1	1
Temporal modulation	2DConv	(1,1)	n	n	1	1
1/4 resolution reconstruction	2DConv	(3,3)	256	256	1	8
	2DConv	(3,3)	256	256	1	4
	2DConv	(3,3)	256	256	1	2
	2DConv	(3,3)	256	256	1	1
	2DConv	(3,3)	256	2	1	1
1/2 resolution reconstruction	2DConv	(3,3)	512	128	2↑	1
	2DConv	(3,3)	128	128	1	1
	2DConv	(3,3)	128	128	1	1
	2DConv	(3,3)	128	2	1	1
Full resolution reconstruction	2DConv	(3,3)	256	64	2↑	1
	2DConv	(3,3)	64	32	1	1
	2DConv	(3,3)	32	2	1	1

Table 1. The structure of the IGFC. The "Matching Net" and the "Temporal modulation" parts correspond to the "Feature fusion" part in the main paper. These two components are adapted from [8]. We use these two components to aggregate the features from the inertia warped flows. "n" represents the number of the optical flows to be processed. The input dimension is 3 because we concatenate the optical flows and their corresponding masks. ↓ means the downsampling operation, while ↑ means the upsampling operation. Skip connection is inserted before the first block in the "1/2 resolution reconstruction" and the "Full resolution reconstruction" module.

Module	Block	Filter size	In channels	Out channels	Stride/Up	Dilation
Encoder	2DConv	(5,5)	3	64	1	1
	2DConv	(3,3)	64	128	2↓	1
	2DConv	(3,3)	128	128	1	1
	2DConv	(3,3)	128	256	2↓	1
Middle	ASF module×4	-	256	256	1	1
Decoder	ResBlock×2	(3,3)	256	256	1	1
	2DConv	(3,3)	512	128	2↑	1
	2DConv	(3,3)	128	128	1	1
	2DConv	(3,3)	256	64	2↑	1
	2DConv	(3,3)	64	32	1	1
	2DConv	(3,3)	32	2	1	1

Table 2. The structure of the ASFN. We use the network to correct the style of the warped gradients with the guidance of the valid regions. Skip connection is inserted in the first and the third convolution block in decoder. The detailed structure of the ASF module can be viewed in the main paper.

for simplicity, and the calculation in other resolutions can be extended straightforwardly.

$$L_{ter}^{original} = \frac{\|D \odot m \odot M_t\|_1}{\|M_t\|_1} \tag{5}$$

1.3. Gradient warping procedure

Given the corrupted video frames and the completed optical flows, we extract the gradients from the video frames in x and y direction, and warp the valid regions of the gradients to fill the corrupted regions with the corresponding bidirectional completed flow in a chain-like manner, which means we reuse the content warped from other frames and may propagate it to the others. Therefore, for each frame except the head and the tail in the video, we can obtain the forward and the backward warped gradient maps in x and y directions. We fuse the forward and backward propagated gradient maps in each direction with the forward-backward consistency checked by the completed flow pair. Given the completed optical bidirectional flows $\hat{F}_{t-1\to t}$ and $\hat{F}_{t\to t-1}$, the forward-backward consistency check is illustrated as,

$$\begin{aligned} k &= \hat{F}_{t\to t-1}(p) \\ \hat{D}_{t\to t-1}(p) &= \left\| k + \hat{F}_{t-1\to t}(p+k) \right\|_2^2 \end{aligned} \tag{6}$$

where p the sampled point, k is the mesh warped from frame t to frame $(t-1)$, and $\hat{D}_{t\to t-1}$ is the cycle consistency checked with the chain of $t \to (t-1) \to t$. For the cycle consistency map between frame t and frame $(t+1)$, replace $(t-1)$ in equation 6 with $(t+1)$.

We calculate the weight map $\omega_{t\to t-1}$ and $\omega_{t\to t+1}$ for gradient fusion with the consistency error map $\hat{D}_{t\to t-1}$ and $\hat{D}_{t\to t+1}$, which can be formulated as,

$$\begin{aligned} \omega_{t\to t-1} &= \frac{exp(-\hat{D}_{t\to t-1}/d)}{exp(-\hat{D}_{t\to t-1}/d) + exp(-\hat{D}_{t\to t+1}/d) + \epsilon} \\ \omega_{t\to t+1} &= \frac{exp(-\hat{D}_{t\to t+1}/d)}{exp(-\hat{D}_{t\to t-1}/d) + exp(-\hat{D}_{t\to t+1}/d) + \epsilon} \end{aligned} \tag{7}$$

where d is the temperature coefficient and ϵ is an extremely small value to avoid division by zero error. We set d to 0.1 and ϵ to 1e-7. Given the warped gradient map $\nabla_x \tilde{I}_{t-1}$, $\nabla_x \tilde{I}_{t+1}$, $\nabla_y \tilde{I}_{t-1}$ and $\nabla_y \tilde{I}_{t+1}$, we fuse the gradient maps with the weight maps $\omega_{t\to t-1}$ and $\omega_{t\to t+1}$

$$\begin{aligned} \nabla_x \tilde{I}_t &= \omega_{t\to t-1} \nabla_x \tilde{I}_{t-1} + \omega_{t\to t+1} \nabla_x \tilde{I}_{t+1} \\ \nabla_y \tilde{I}_t &= \omega_{t\to t-1} \nabla_y \tilde{I}_{t-1} + \omega_{t\to t+1} \nabla_y \tilde{I}_{t+1} \end{aligned} \tag{8}$$

After the gradient fusion, if there still exists unfilled regions (occluded regions), we adopt Poisson blending to map the gradient to RGB domain and adopt the DeepFillV1 [10] to fill the frame with the largest unfilled regions, and propagate such filled regions to the other frames with the completed flows.

1.4. Detailed adversarial loss for ASFN training

As for the training of ASFN, we adopt reconstruction loss Ls_{rec} and the GAN loss. We adopt hinge loss to supervise the ASFN training. Given the refined gradient $\nabla \hat{I}_t$ and the ground truth gradient ∇I_t, the discriminator loss can be illustrated below:

$$\begin{aligned} Ls_D = &\mathbb{E}_{x\sim P_{\nabla I_t}(x)}[\text{ReLU}(1 + D(x))] \\ &+ \mathbb{E}_{z\sim P_{\nabla \hat{I}_t}(z)}[\text{ReLU}(1 - D(z))] \end{aligned} \tag{9}$$

where D is the discriminator, the adversarial loss is:

$$Ls_{adv} = -\mathbb{E}_{z\sim P_{\nabla \hat{I}_t(z)}}[D(z)] \tag{10}$$

1.5. Acceleration of Poisson blending

FGVC [2] constructs the Poisson equation based on the whole frame to render the RGB frames from the completed gradients. The Poisson equation can be written as $Ax = b$. A is the coefficient matrix, x is the vector contains the pixels of each regions to be synthesized, and b is the known gradients in the boundary. In the implementation of FGVC, they synthesize all the pixels with Poisson equation. Given the frame with $H \times W$ resolution and the mask with $h \times w$ size of masked regions, the shape of A in FGVC is $hw \times HW$, the shape of x is $HW \times 1$, and the shape of b is $hw \times 1$. Because the corrupted regions mainly occupy a small area of each frame, it is unnecessary to construct Poisson equation based on the whole frame.

Different from FGVC, we construct the Poisson equation based on the corrupted regions and their 2-pixel boundaries. In this setting, Poisson blending can get access to the gradients and variation in the boundaries, which is enough for Poisson blending to synthesize the corrupted pixels.

Therefore, the shape of A in our method is $hw \times (hw + 4(h+w))$, the shape of x is $(hw + 4(h+w)) \times 1$, and the shape of b is $hw \times 1$. Since $hw + 4(h+w)$ is always smaller than HW, we can accelerate the Poisson blending.

For the masks with multiple connected components, we search the connected components first, and use our accelerated Poisson blending to synthesize these connected components in an iterative manner. The detailed algorithm is shown in Alg. 1

We test our accelerated Poisson blending on multiple video sequences, and observe 20% to 30% acceleration.

Algorithm 1 Accelerated Poisson blending

Require:
Warped and refined gradients $\nabla\hat{I}_x$ and $\nabla\hat{I}_y$
Corrupted frame $\overline{I}_t$ and its corresponding mask M_t
Ensure: The inpainted result $\hat{I}_t$
function MAIN($\nabla\hat{I}_x, \nabla\hat{I}_y, M_t, \overline{I}_t$)
    # Dilate mask by 1 pixels
    $\hat{M}_t = Dilate(M_t, 1)$
    $cc = searchConnectedComponent(\hat{M}_t))$
    for $c \in cc$ **do**
        $c = Dilate(c, 1)$
        $\hat{I}_t(c) = SolvePoisson(\overline{I}_t(c), \nabla\hat{I}_x(c), \nabla\hat{I}_y(c))$
    end for
    return $\hat{I}_t$
end function

2. More experimental results

2.1. Flow numbers in IGFC

As for IGFC, we adopt 3 consecutive optical flows $\tilde{F}_{t-1} \sim \tilde{F}_{t+1}$ to restore the target flow $\tilde{F}_t$ based on the efficiency and performance tradeoff. We observe a constant performance gain when the flow numbers grows from 1 to 5. The performance drops when the flow number is 7. We argue that the inertia prior between the target flow and the reference flow degrades when they are distant. Therefore, the performance in IGFC drops when the flow number is too large. Fig. 1 illustrates the variation of PSNR w.r.t. the number of optical flows on DAVIS dataset [1].

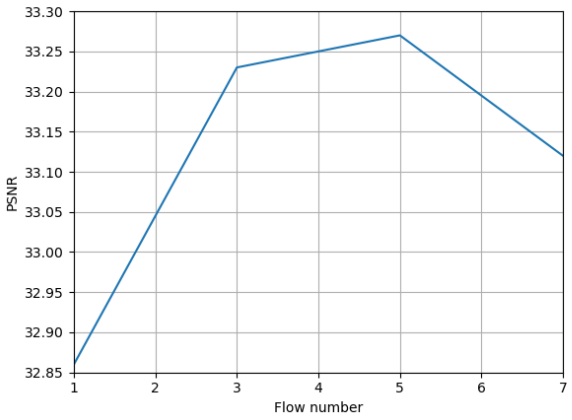


Figure 1. The PSNR value w.r.t. the number of flows, the x-axis represents the flow number for processing, and the y-axis represents the corresponding PSNR value. This experiment is conducted on the DAVIS square mask set for parameter selection.

2.2. ASF module numbers in ASFN

We adopt 4 ASF modules in ASFN to strike the balance between efficiency and performance. The performance saturates when the number of ASF module is larger than 4. Fig. 2 illustrates the variation of PSNR on DAVIS w.r.t. the number of ASF modules in ASFN.

Time	FGVC [2]	Ours
Flow computation	17.43s	17.43s
Flow completion	116.54s	9.58s
Flow guided gradient propagation	8.23s	7.48s
ASFN gradient refine	0.00s	4.53s
Poisson blending	55.65s	46.15s
Sum on the full sequence	197.85s	85.17s
Average run time of each frame	2.83s	1.22s

Table 3. The runtime analysis on "Paragliding" video sequence from DAVIS, which contains 70 frames. We adopt the accelerated Poisson blending to render the inpainted frames. FGVC does not apply ASFN for gradient refinement, therefore the corresponding processing time is zero. And the "Flow guided gradient propoagation" includes the spatial inpainting time cost.

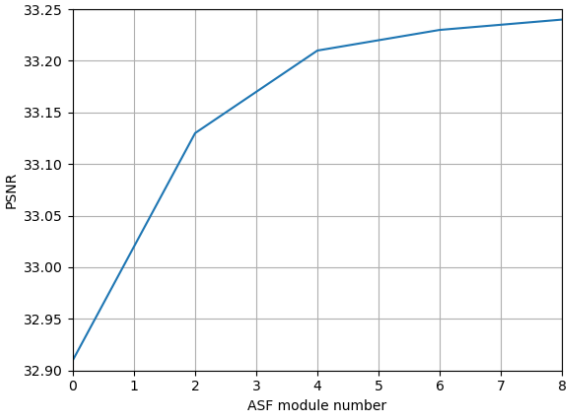
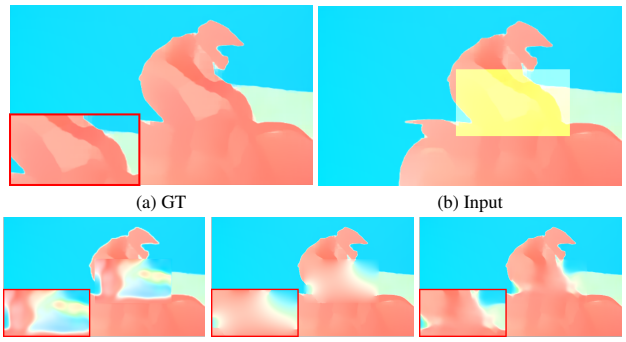


Figure 2. The PSNR value with regard to the number of ASF modules. We illutrate the variation of PSNR w.r.t. number of ASF modules on the DAVIS square mask set for parameter selection.

2.3. Run time analysis

We analyse different parts of our method on "paragliding" video sequence from DAVIS. We also analyse the speed of FGVC [2] on the same sequence. Our workstation is equipped with four NVIDIA 2080ti GPUs and one Intel(R) Xeon(R) Gold 5118 CPU. All the video frames are resized to 432 $\times$ 256 for fair comparison. We ignore the IO time cost which varies across different workstations. The run time analysis is shown in Tab. 3. Compared with FGVC, our method mainly accelerates the flow completion and the Poisson blending procedure. The newly introduced ASFN does not consume too much computation resource because of its light-weight network structure.



(a) GT (b) Input

(c) DFGVI [9] (d) FGVC [2] (e) Ours

Figure 3. Flow completion results when the flow contains fine-grained structure. Although our flow completion method can achieve more accurate structure than previous methods, our flow completion results still suffer the restoration in the regions with fine-grained details.

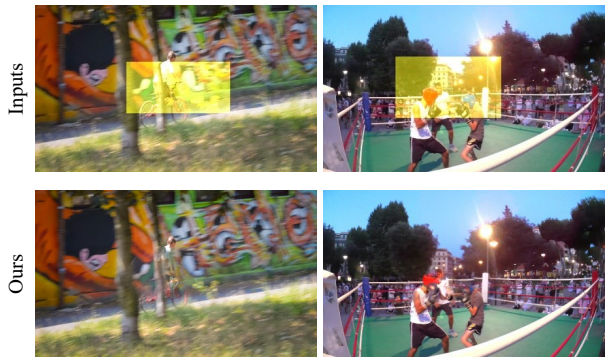


Figure 4. Our limitations in fast motion and large corrupted regions.

3. Detailed Limitation Analysis

Fine-grained flow completion Flow completion is still an open problem. When the mask occludes the fine-grained semantic structure of the optical flow, even if our flow completion can reconstruct partial structure, the fine-grained details is missing, which is shown in Fig. 3.

Fast motion and large corrupted regions Our method suffers from inpainting upon videos with fast motion and large corrupted regions. When the range of motion gets larger, the performance of our method degrades, as illustrated in Fig. 4.

Processing speed The running speed of our method is comparable to other flow based video inpainting methods. However, all the flow-based video inpainting methods are slower than the attention-based methods. The bottleneck of the speed of our method comes from two aspects, the first is the flow extraction, and the second is Poisson blending. Our implementation of Poisson blending is on CPU now, we will transfer the implementation of Poisson blending to GPU for much faster running speed in the future work.

4. More visual results

4.1. Our results on 4K sequences.

Compared with attention based video inpainting methods, our proposed method is quite memory-efficient. Without bells and whistles, we can process even 4K video sequences with high quality. Our results on 4K sequences are shown in Fig. 5.

4.2. More flow comparison results.

We provide more visualized flow results to demonstrate the superior performance of our proposed IGFC in comparison with two previous flow guided video inpainting methods [2, 9]. The comparison is shown in Fig. 6.

4.3. More qualitative results.

We show more results to compare the subjective performance between our method and the others. We provide video demo in https://youtu.be/dHuFDPDWkYc for video-wise qualitative comparisons. And we also provide the frame-wise qualitative comparisons in Fig. 7, 8, 9, and 10, which correspond to the square mask, object mask, object removal on DAVIS and square mask on YoutubeVOS, respectively.

References

[1] Sergi Caelles, Alberto Montes, Kevis-Kokitsi Maninis, Yuhua Chen, Luc Van Gool, Federico Perazzi, and Jordi Pont-Tuset. The 2018 DAVIS challenge on video object segmentation. *arXiv preprint arXiv:1803.00557*, 2018. 4

[2] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *ECCV*, pages 713–729, 2020. 3, 4, 5, 7, 8, 9, 10, 11

[3] Sungho Lee, Seoung Wug Oh, DaeYeun Won, and Seon Joo Kim. Copy-and-paste networks for deep video inpainting. In *ICCV*, pages 4413–4421, 2019. 8, 9, 10, 11

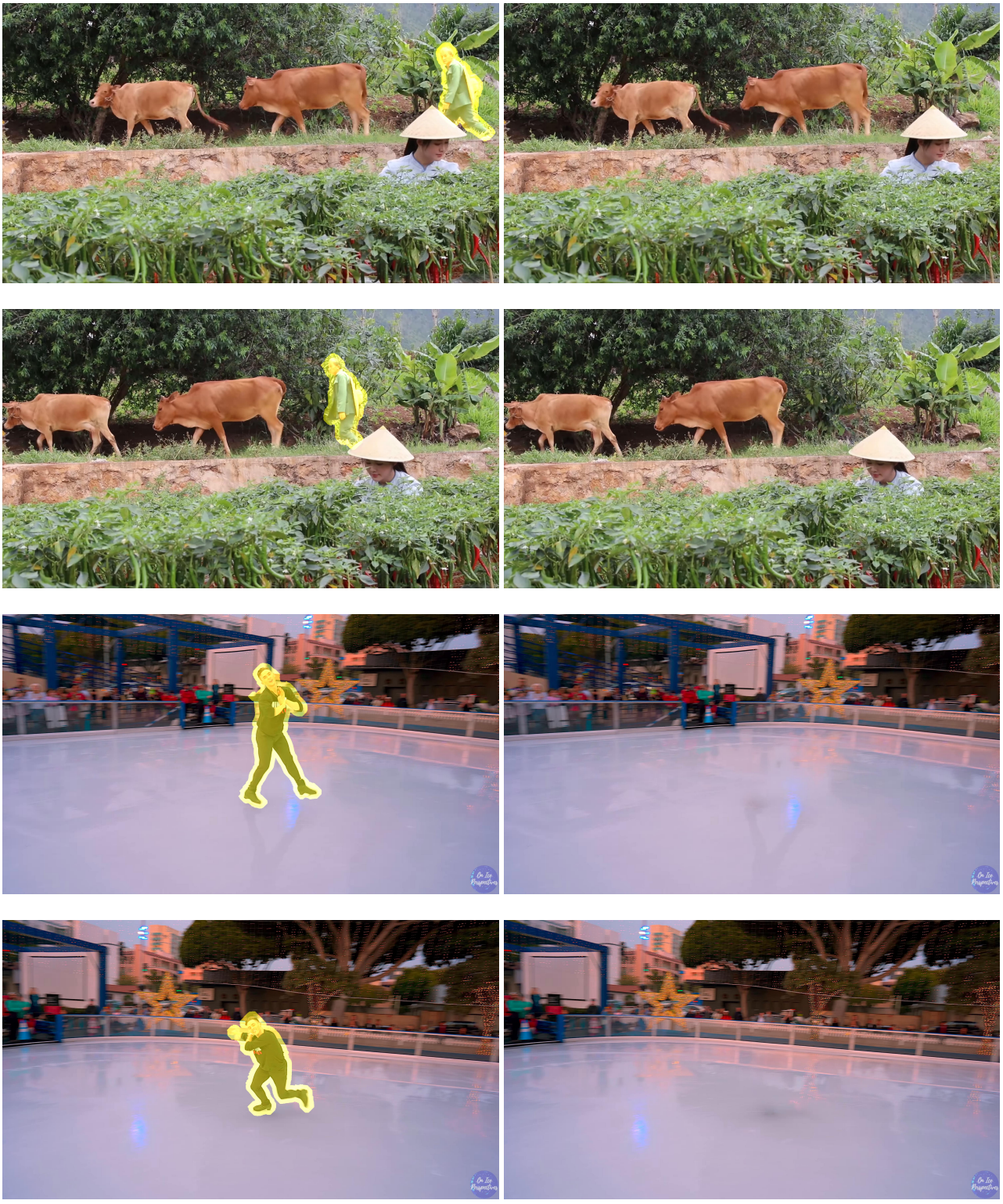
[4] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *ICCV*, 2021. 8, 9, 10, 11

[5] Seoung Wug Oh, Sungho Lee, Joon-Young Lee, and Seon Joo Kim. Onion-peel networks for deep video completion. In *ICCV*, pages 4403–4412, 2019. 8, 9, 10, 11

[6] Hao Ouyang, Tengfei Wang, and Qifeng Chen. Internal video inpainting by implicit long-range propagation. In *ICCV*, 2021. 6

[7] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419. Springer, 2020. 1

[8] Jianyuan Wang, Yiran Zhong, Yuchao Dai, Kaihao Zhang, Pan Ji, and Hongdong Li. Displacement-invariant matching cost learning for accurate optical flow estimation. *NIPS*, 33, 2020. 1, 2

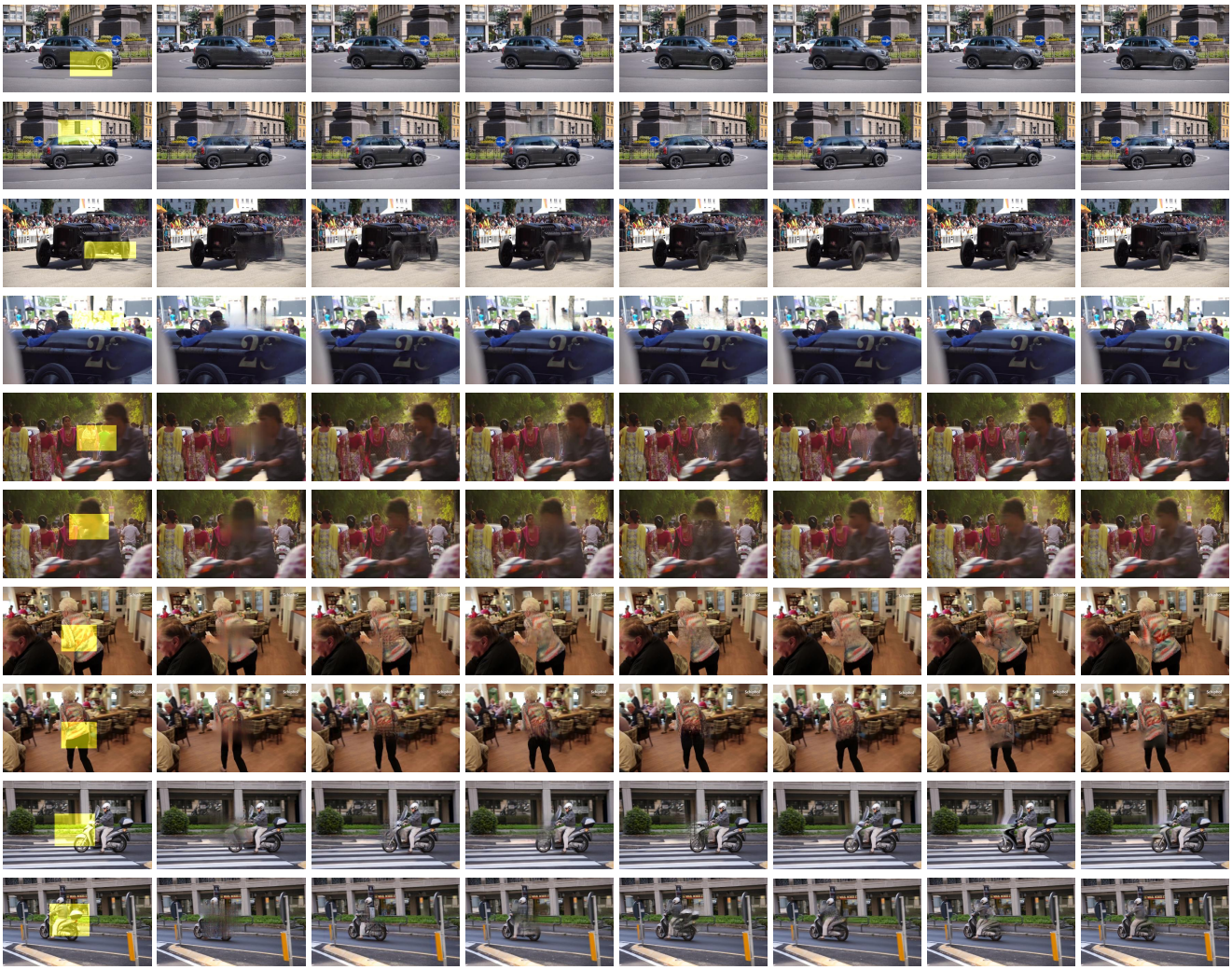


(a) Inputs (b) Results

Figure 5. The object removal results of our proposed method on the videos at 3840 × 2160 resolution. The original 4K video sequences are collected and annotated by [6].

(a) Input (b) DFGVI [9] (c) FGVC [2] (d) Ours (e) GT

Figure 6. The flow completion results of our method, DFGVI [9] and FGVC [2]. The comparison illustrates the accuracy of our IGFC in completing optical flows.



(a) Input (b) CPN [3] (c) OPN [5] (d) STTN [11] (e) TSAM [12] (f) FFM [4] (g) FGVC [2] (h) Ours

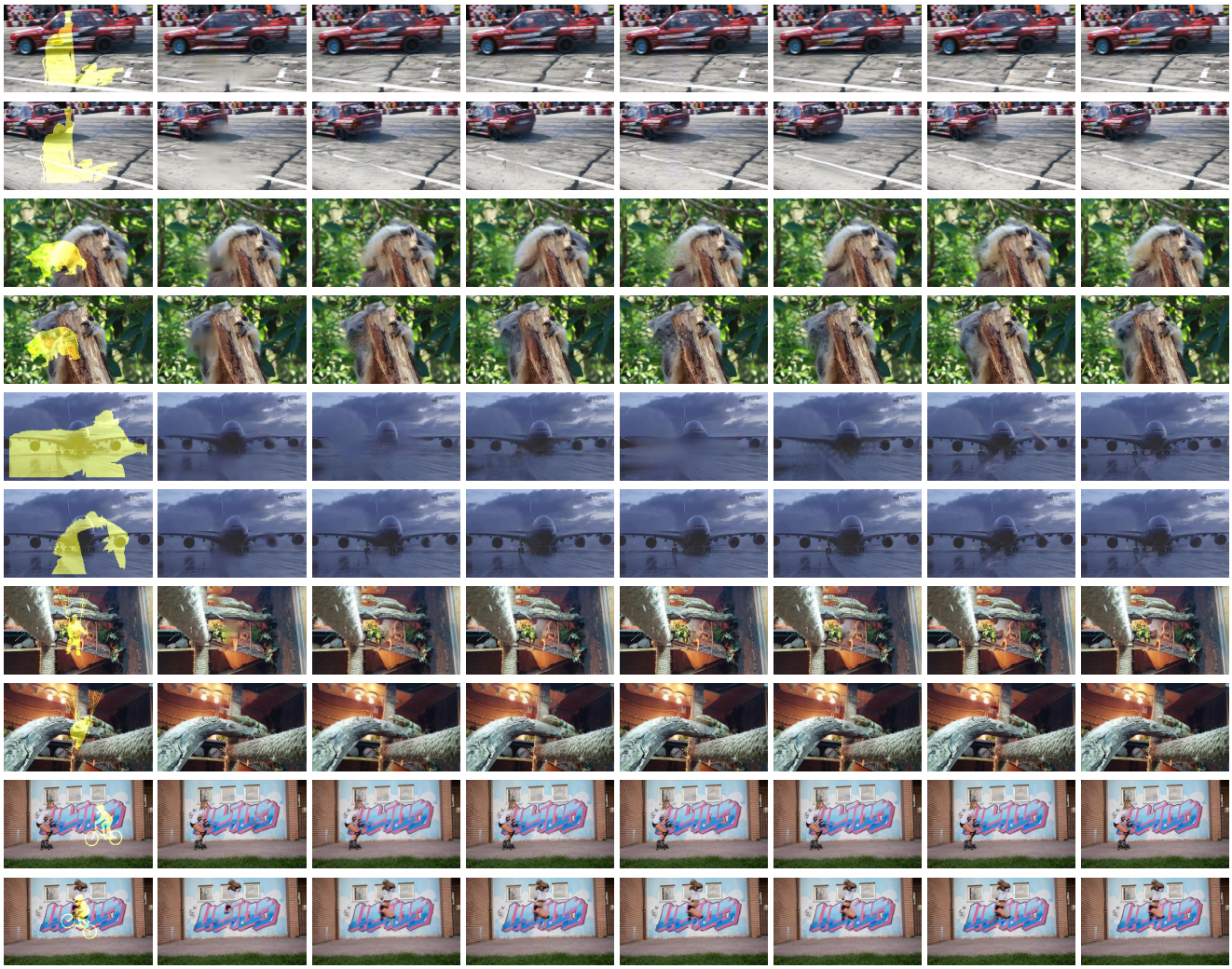
Figure 7. The qualitative results of the SOTA methods on DAVIS square mask set. Best viewed with zoom-in.

[9] Rui Xu, Xiaoxiao Li, Bolei Zhou, and Chen Change Loy. Deep flow-guided video inpainting. In *CVPR*, pages 3723–3732, 2019. 5, 7

[10] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Generative image inpainting with contextual attention. In *CVPR*, pages 5505–5514, 2018. 1, 3

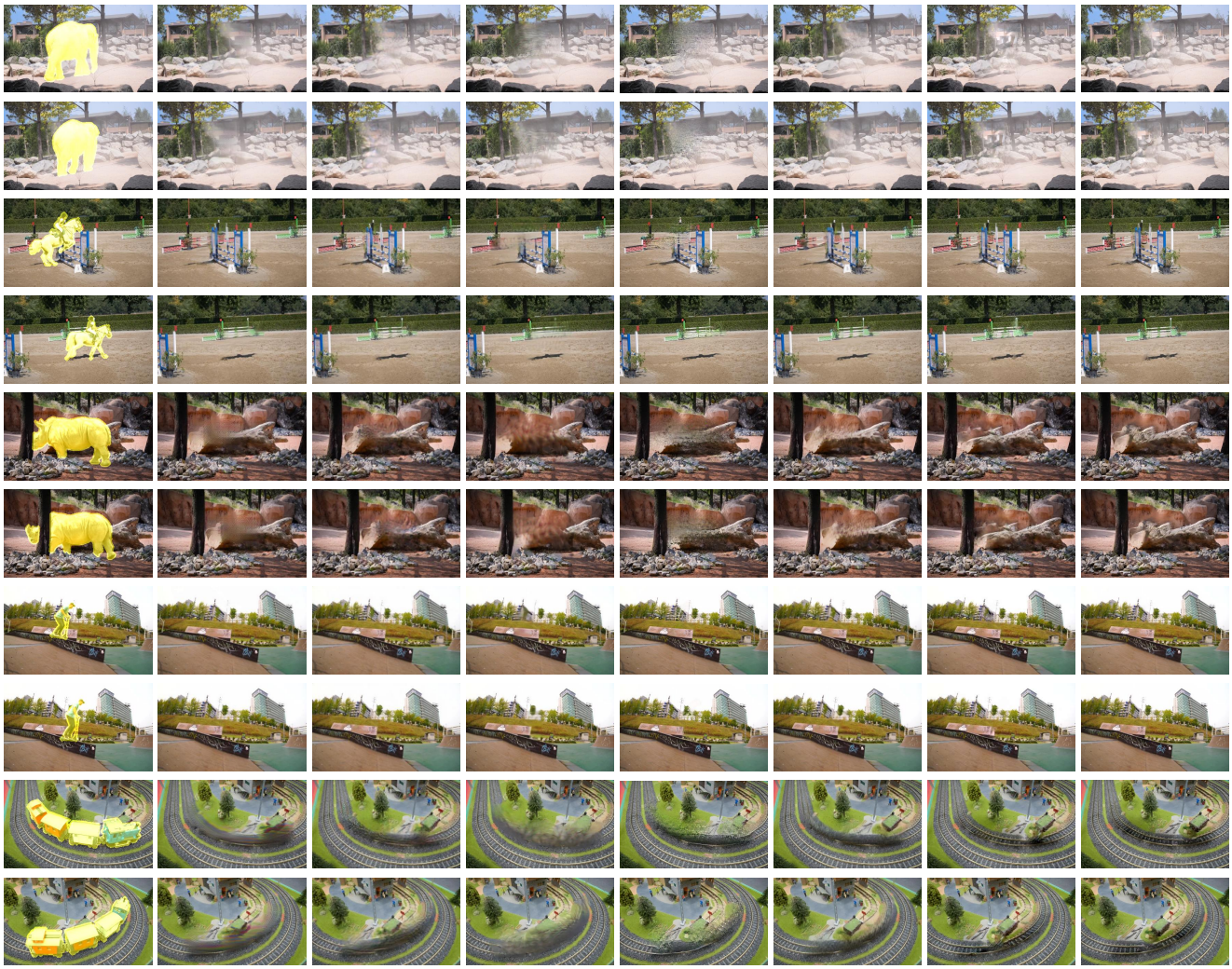
[11] Yanhong Zeng, Jianlong Fu, and Hongyang Chao. Learning joint spatial-temporal transformations for video inpainting. In *ECCV*, pages 528–543, 2020. 8, 9, 10, 11

[12] Xueyan Zou, Linjie Yang, Ding Liu, and Yong Jae Lee. Progressive temporal feature alignment network for video inpainting. In *CVPR*, 2021. 8, 9, 10, 11



(a) Input (b) CPN [3] (c) OPN [5] (d) STTN [11] (e) TSAM [12] (f) FFM [4] (g) FGVC [2] (h) Ours

Figure 8. The qualitative results of the SOTA methods on DAVIS object mask set. Best viewed with zoom-in.



(a) Input (b) CPN [3] (c) OPN [5] (d) STTN [11] (e) TSAM [12] (f) FFM [4] (g) FGVC [2] (h) Ours

Figure 9. The qualitative results of the SOTA methods on DAVIS for object removal. Best viewed with zoom-in.

(a) Input (b) CPN [3] (c) OPN [5] (d) STTN [11] (e) TSAM [12] (f) FFM [4] (g) FGVC [2] (h) Ours

Figure 10. The qualitative results of the SOTA methods on YoutubeVOS square mask set. Best viewed with zoom-in.