Illumination-Guided Video Composition via Gradient Consistency Optimization

Jingye Wang, Bin Sheng, Ping Li, Yuxi Jin, and David Dagan Feng, *Fellow, IEEE*

***Abstract*— Video composition aims at cloning a patch from the source video into the target scene to create a seamless and harmonious blending frame sequence. Previous work in video composition usually suffers from artifacts around the blending region and spatial–temporal consistency when illumination intensity varies in the input source and target video. We propose an illumination-guided video composition method via a unified spatial and temporal optimization framework. Our method can produce globally consistent composition results and maintain the temporal coherency. We first compute a spatial–temporal blending boundary iteratively. For each frame, the gradient field of the target and source frames are mixed adaptively based on gradients and inter-frame color difference. The temporal consistency is further obtained by optimizing luminance gradients throughout all the composition frames. Moreover, we extend the mean-value cloning by smoothing discrepancies between the source and target frames, then eliminate the color distribution overflow exponentially to reduce falsely blending pixels. Various experiments have shown the effectiveness and high-quality performance of our illumination-guided composition.**

***Index Terms*— Illumination aware, image cloning, gradient fields, video composition.**

I. Introduction

VIDEO composition is the process of cloning a patch from the source video into the target video sequence, where the patch is often the moving foreground objects in the source video. Such composition is a popular and useful video editing technique in the film, game and entertainment industries [1]–[4]. To date, there is no acknowledged measuring standard for evaluating the quality of composition video. In general, three challenges are involved in such tasks, including extracting the patch automatically, cloning it to the target region naturally, and getting rid of supervision. A common method is a cyclic process of cutting out and pasting using interactive video editing tools or utilizing image blending methods frame-by-frame. The 3D Poisson video composition method [5] addressed the problem by solving a 3D Poisson equation system. Apart from Poisson blending, mean-value coordinates (MVC) interpolation [6] has also been introduced, which can achieve similar blending results and is highly parallelizable compared to Poisson blending. However, these frame-by-frame processing methods cost intensive labor, and the user-given trimap is not always precise enough.

Recently, many methods on video composition have been proposed. Xie *et al.* [7] refined the composition results by removing artifacts along the blending boundary, but the method lacks temporal coherency with the challenge of illumination varying and object motion. Chen *et al.* [8] proposed to mix the gradient field and use mean-value interpolation to support hybrid blending to deal with motion difference around blending boundary. Shen *et al.* [9] introduced a superpixel segmentation through density-based spatial clustering of applications with noise (DBSCAN) in real time. A fast two-step approach is applied in their work to decrease the calculation cost. Image matting [10], [11] can be utilized for video composition. These matting-based methods are apt to produce good composites with great discrepancies between the source path and target scene. In general, many conventional video composition methods are developed on the basis of image composition like Poisson blending and mean-value coordinates interpolation. They used optical flow and local features to keep spatial and temporal consistency. They concentrate on dealing with motion problems [8] or removing blending artifacts [7], but are not effective enough to significant illumination condition variation, including illumination intensity change and cast shadow. Especially, if light intensity changes reversely in source and target videos, the appearance of the blended object is usually not in accordance with the target scene brightness change. The challenges may also lead to object appearance flickering and temporal incoherence. Furthermore, it brings more difficulty to obtain high-quality compositions if there is complex motion in surroundings like smokes, clouds, etc.

Manuscript received January 5, 2018; revised July 29, 2018 and March 2, 2019; accepted April 22, 2019. Date of publication May 20, 2019; date of current version August 14, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61872241 and Grant 61572316, in part by the National Key Research and Development Program of China under Grant 2017YFE0104000 and Grant 2016YFC1300302, in part by the Macau Science and Technology Development Fund under Grant 0027/2018/A1, and in part by the Science and Technology Commission of Shanghai Municipality under Grant 18410750700, Grant 17411952600, and Grant 16DZ0501100. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zhengguo Li. *(Corresponding author: Bin Sheng.)*

J. Wang is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China.

B. Sheng is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: shengbin@sjtu.edu.cn).

P. Li is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: lipingfire@ieee.org).

Y. Jin is with the Faculty of Information Technology, Macau University of Science and Technology, Macau 999078, China.

D. D. Feng is with the Biomedical and Multimedia Information Technology Research Group, School of Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: dagan.feng@sydney.edu.au).

Digital Object Identifier 10.1109/TIP.2019.2916769

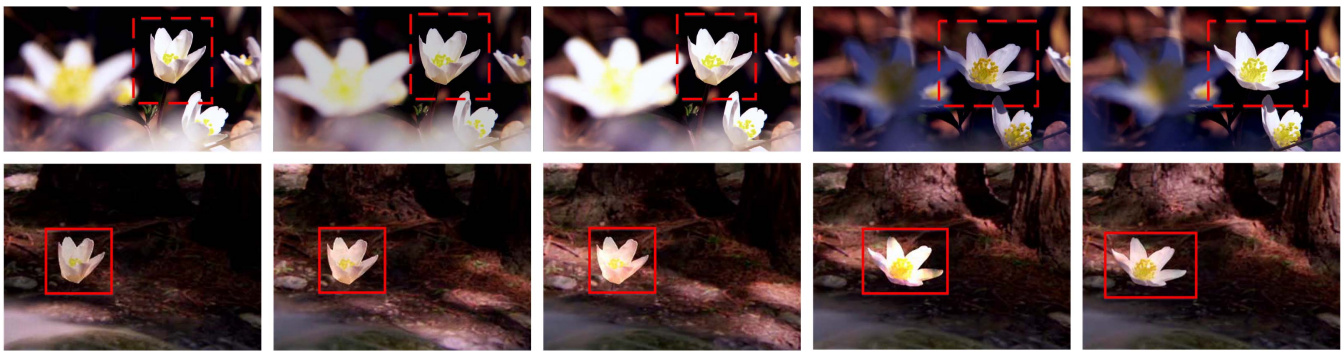


Fig. 1. A video composition example of our method on *flower1* sequence. It demonstrates the effectiveness to illumination difference of our method. Note that the flower is adjusted to suitable size in the composition video. The source frames are listed in the first row, where the red dashed rectangle marks the blooming flower to blend. In the source scene, the global light intensity changes over time. The corresponding composition results are shown in the second row. The region of interest is marked with a red rectangle. The flower in the blending region is harmonious with the target illumination condition.

Our key idea is to use gradient consistency optimization and local discrepancy smoothing to address the above issues. When processing each frame, we define a cost function for spatial-temporal coherent blending boundaries where taking into account the global and partial illumination variation and motion difference. We propose a spatial-temporal consistent mixing strategy to achieve seamless blending results when illumination changes. The gradient mixing parameter is generated in the first frame by the gradients ratio, and adjusted by inter-frame color difference. We deal with sudden illumination variation by designing the color terms in the cost function to reduce color flickering in the blending regions. We further enhance the temporal consistency by optimizing a patch-based energy function. Moreover, we optimize the mean-value interpolation with an additional term to smooth the contrasts between the source patch and target scene. We further eliminate the discoloration artifacts by constraining the color overflow in interpolation. Fig. 1 shows the effectiveness to deal with illumination difference using our approach, where we intend to cut out the blooming flower in the source frame and paste it into the region of interest. The global light intensity varies in the source frames. Meanwhile, the illumination condition changes due to the tree shadow as shown in Fig. 1. In the composition video frames, the blended flower is in accordance with the cast shadow of the tree in the target. Our approach has the following three main contributions:

- **Optimized interpolation for image cloning** We make the blending object consistent with the target brightness by smoothing the local discrepancies between the source and target videos, and constraining the overflow in color distribution to reduce discoloration in the blending region.
- **Spatial-temporal consistent blending boundary optimizing** We propose a spatial-temporal consistent boundary computing by optimizing an energy function to tackle the challenge of illumination variation and motion.
- **Illumination-guided gradients mixing** We propose a seamless composition strategy by mixing gradients to maintain temporal-spatial consistency, and further optimize temporal coherency over all frames.

II. Related Work

Video composition is a hot topic in video processing and computer vision. We could not enumerate all literature, but attempt to focus on the popular and effective work. Researches on matting techniques, gradients domain methods, and video composition approaches are reviewed here.

Matting techniques produce an alpha-matte, which provides the weights for linear pixel interpolation of two images. Image matting methods often require a user-defined trimap to compute the alpha values in the uncertain region and the foreground. Video matting approaches are extended from image matting. Thus, additional constraints are required to use trimaps or other constraints before image matting approaches are applied on all video frames [12]–[17]. Traditional trimap propagation methods include: optical flow [18], graph-cut [19] and geodesic segmentation [20]. However, it is difficult for these methods to produce a precise trimap if foreground and background layers have divergent motions. Matting Laplacian methods can provide spatial-temporally coherent clusters of source patch pixels, but user-provided trimap needs to be dense and precise [21], [22]. Closed-form matting are proposed to deal with scribbled pixels [11], [23], [24]. Many state-of-the-art methods focus on maintaining temporal-spatial consistency, but more manual supervision is required to address complex motion and varying illumination.

Gradient domain methods are often used for their effectiveness in seamless video composition. In video composition and video editing technologies, gradients of two input frames are mixed to create a consistent blending region around the fuzzy object boundaries [25], [26]. However, the composition quality depends heavily on the blending region boundary, thus the boundary optimization is a necessary step in these methods. Drag-and-drop pasting [27] computed a boundary from user-drawn loop to minimize color mismatch then mix the gradients. Chen *et al.* [8] proposed a mixing gradient field and used mean-value interpolation to support hybrid blending. Most gradient domain methods take into account the intra-frame space information, and are often combined with motion information methods like optical flow to create results with temporal consistent effects [28]–[30]. If the illumination varies in the source object and target scene, or the texture

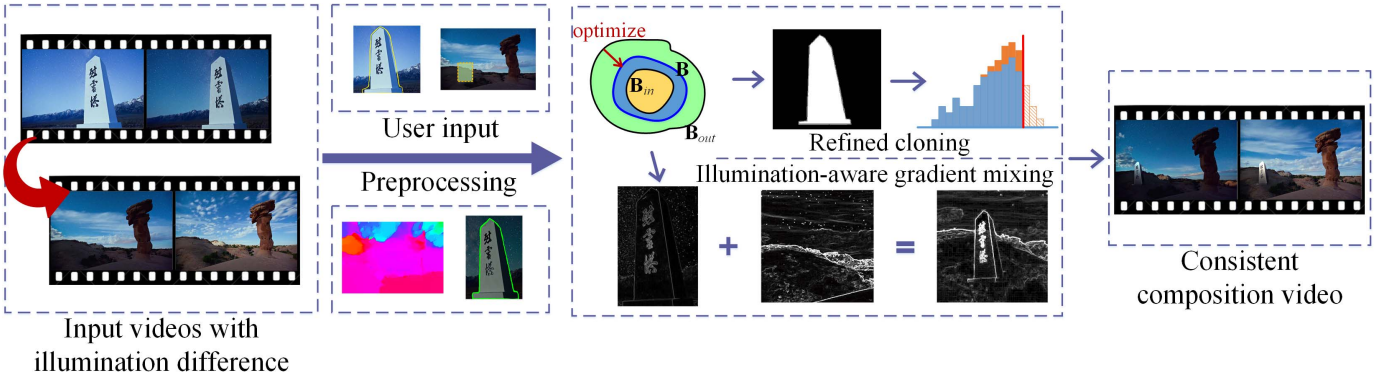


Fig. 2. Overview of the proposed illumination-aware video composition in the gradient domain on *stone* sequence. We intend to paste the tower into the target scene, where the illumination intensity changes oppositely. User strokes are required in the keyframes, and preprocessing consists of bidirectional optical flow estimating together with definite foreground extracting. We first optimize the blending boundary, then clone the patch by smoothing local discrepancies and constraining color overflow. We also composite the video in the gradient domain where inter-frame color difference is taken into account. The output video is spatial-temporal consistent with the challenge of dramatic illumination difference.

of the source object and target is inconsistent, users have to re-draw the blending boundary in more key-frames. Moreover, gradients mixing based methods cannot effectively suppress color flickering in cloned region when illumination intensity varies dramatically in the source or target scene, because they consider the temporal consistency based on only optical flow between adjacent frames.

Video composition approaches have been proposed to paste an image patch into the region of interest. They are developed based on image cloning methods like Poisson blending [5] or mean-value coordinate composition [6]. 3D Poisson video cloning obtains relatively good results but is labor-intensive because it requires frame-by-frame supervision. Levin *et al.* [31] introduced Poisson blending into image stitching in the gradient domain, and the artifacts caused by misalignment are reduced. Chen *et al.* [32] solved the Poisson equation by using the mixed boundaries to remove the composition artifacts. Besides Poisson equation based cloning, many composition methods can achieve similar composition results. Mean-value coordinates interpolation [6] achieves similar blending results but is lower in computational complexity. Approaches on video editing and processing are also highly related to video composition, which also concentrate on video consistency. Intra-frame and inter-frame consistency is usually incorporated with other cues like motion patterns to perform video editing [33], [34]. The across-video consistency can be further maintained by optimizing a temporal consistency function over the entire video [35], [36].

The Poisson-based cloning methods often generate unrealistic blending results, and matting-based methods are recently researched to deal with such problem. Chen *et al.* [37] proposed a video composition method, which automatically composites several related single frames into a long-take video sequence using feature extracting and matching. Li and Hu [38] extended mean-value cloning to decrease Dirichlet energy and estimated a harmonic function to make the composition result realistic. Wang *et al.* [10] remove the falsely computed pixel intensity of composition in the blending region by enhancing usage of mattes to handle artifacts. However, the excessive use of may lead to the loss of blending details around the boundary. Most of the existing video composition approaches are developed based on Poisson blending or mean-value cloning, and concentrate on removal of boundary artifacts to improve composition quality. However, sudden illumination intensity change and unclear object surroundings like smoke or cloud still limit the applicability of these approaches.

III. Approach Overview

The proposed video composition approach is effective to deal with illumination intensity varying. Fig. 2 is the overview of our approach. The input consists of two video sequences including the source and target videos, where the illumination intensity changes inversely. The output is the composition video sequence by pasting the source patch into the region of interest into the target scene. In the interaction part, the user strokes are provided in the keyframes. The coarse blending boundary $\mathbf{B}_{out}$ is provided by the user in the keyframe, which is shown as the yellow closed curve in Fig. 2. The fuzzy surroundings like the smoke, dust, etc. and motion blur which is around the source object should be encompassed by $\mathbf{B}_{out}$. The inner boundary $\mathbf{B}_{in}$ is also given in the keyframes which roughly surrounds the definite foreground object region. The region of interest in the target video is provided as well. It is marked as the green rectangle enclosed by the yellow dashed line in Fig. 2. In the preprocessing step, the optical flow field is estimated for each frame bidirectionally. Trimaps are generated and updated based on the two boundaries and matting results. Algorithm 1 shows our illumination-aware video composition. Video composition is developed based on image cloning methods, so we first introduce our improved image cloning method in Section IV. We propose a local smoothing term to compute the mean-value cloning interpolant to decrease the discrepancies between the source and target in Section IV-A. In Section IV-B, we further refine the computed interpolant by constraining color overflow exponentially to avoid discoloration artifacts in the blending region.

Algorithm 1 Illumination-Aware Video Composition

Input: Source video $\mathcal{F}^s$, target video $\mathcal{F}^t$, user-provided inner and outer boundaries $\mathbf{B}_{in}$ and $\mathbf{B}_{out}$ in the first frame
Output: Composition video $\mathcal{F}$

```
User indicates B_in and B_out in the first frame;
Cut out the foreground objects, compute the optical flow v_p^t and v_p^s in the target and source video bidirectionally;
Compute the initial gradient mixing parameter a_i^init via Eq. (14);
repeat
    Optimize the blending boundary B using Eq. (8);
    Update gradient mixing parameter a_i using Eq. (16);
    Compute the optimized mean-value interpolant;
    Refine the composition frames using Eq. (17);
    Copy the composition frame into output sequence F;
until User strokes are required in the keyframes
```

To conduct composition task on the video sequences, we introduce how we obtain the blending boundary on each frame, and how to maintain the spatial-temporal consistency of composition video frames in Section V. We define a cost function to compute a coherent blending boundary by minimizing the neighboring pixel color mismatch between the source and target in Section V-A. The gradient mixing and optimized mean-value cloning are executed within the obtained boundary. In Section V-B, we introduce our gradient mixing method which helps obtaining a globally consistent composition result and tackles the challenge of motion and illumination variation. We further refine the composition results by considering the temporal consistency over all frames.

IV. Refined Interpolation for Image Cloning

Poisson equation can be utilized for image blending, but Poisson-based methods cost much time to solve a linear equation system, thereby is not time efficient. In view of this issue, our approach is developed based on mean-value interpolation [6]. Mean-value cloning has advantageous in terms of speed and parallelization. However, similar to Poisson blending, it is still limited with the presence of color flickering and spatial inconsistency when introduced into video composition. We optimize mean-value cloning by first smoothing discrepancies between the source and target, then constraining the color overflow, which aims to make blending region harmonious with target surroundings and eliminate the discoloration artifacts.

Let $\mathbf{\Omega}$ denote the region of source patch, which is pasted into the region of interest in target, and let $\mathbf{B}$ be the boundary of $\mathbf{\Omega}$. The outer boundary $\mathbf{B}_{out}$ is a closed loop curve in key-frames defined by the user. A possible way to extract foreground object is to identify the background in dynamic videos [39]. We use semi-supervised image segmentation method the obtain the foreground region. The inner boundary $\mathbf{B}_{in}$ is generated by applying GrabCut [40] to generate a closed loop based on another user-drawn loop curve which surrounds the object boundary in the key-frames. The trimap of the first source frame is provided by the user, and then generated using propagation in the subsequent frames. The inner boundary $\mathbf{B}_{in}$ is projected to the next frame with the method of Bai *et al.* [41], and the outer boundary $\mathbf{B}_{out}$ is updated using propagation using [8]. Chen *et al.* [32] classified the blending region into two types $\mathbf{R}_1$ and $\mathbf{R}_2$, and we follow the similar approach. The pixels between $\mathbf{B}_{in}$ and $\mathbf{B}_{out}$ are classified as $\mathbf{R}_1$ if the difference of Gabor feature vectors and the UV color components of them are small, and other pixels are regarded as $\mathbf{R}_2$.

A. Local Smoothing Interpolant

In mean-value cloning [6], the membrane value at each interior point in the blending region is computed as the weighted sum of values around the boundary. After the blending region is triangulated, two steps are involved in image cloning: mean-value coordinate computing and interpolation. Let $\mathbf{F}^s$, $\mathbf{F}^t$ and $\mathbf{F}$ denote the image intensities of the source, target and composition frame, respectively. In this method, the interpolant at each point is obtained based on mean-value coordinates, then $\mathbf{F}$ is computed by adding up $\mathbf{F}^s$ and the interpolants in each channel of color space. Considering an interior point $q \in \mathbf{\Omega}$, we denote $r_1(q)$ as the mean-value interpolant with respect to point q which diffuses the difference of the source patch and the target image. The interpolant is computed as:

$$r_1(q) = \sum_{i=0}^{n-1} \frac{w_i}{\sum_{j=0}^{n-1} w_j} \left(\mathbf{F}^t(p_i) - \mathbf{F}^s(p_i)\right) \tag{1}$$

where, $p_i \in \mathbb{R}^2$ is denoted as a boundary point and $p_i \in \mathbf{B}$. The blending boundary $\mathbf{B}$ is denoted as $\mathbf{B} = \{p_0, p_1, p_2 ... p_{n-1}\}$. The mean-value interpolant weight w_i for boundary point p_i is computed as:

$$w_i = \frac{\tan(\beta_{i-1}/2) + \tan(\beta_i/2)}{||p_i - q||} \tag{2}$$

where, β_i is the angle $\angle p_i, q, p_{i+1}$, $||p_i - q||$ is the space distance between boundary point p_i and inner point q.

When color discrepancy between the patch and target scenes is large, the result of conventional mean-value cloning is sometimes not natural or realistic, which manifests as dramatic appearance brightness difference between the blending region and background. In this case, illumination difference in the source and target video can lead to spatial-temporal inconsistency in the composition video. Wang *et al.* [10] optimized mattes computing by suppressing the difference around the object of interest. We follow the idea of its usage in mattes and design an additional term for interpolant computing to smooth the local discrepancies around the boundary and extend it to image cloning. We utilize the added term to meliorate the blending results to make the composition result more realistic. We add up the conventional interpolant in Eq. (1) and the additional term to compute the new interpolant as:

$$r_2(q) = r_1(q) + \kappa \cdot \sum_{x \in \mathbf{\Omega}} H_x \cdot S(q, x) \cdot \left(\mathbf{F}^t(x) - \mathbf{F}^s(x)\right) \tag{3}$$

where, H_x is a normalized weight for smoothing, we compute H_x for point x in blending region based on alpha matting. H_x

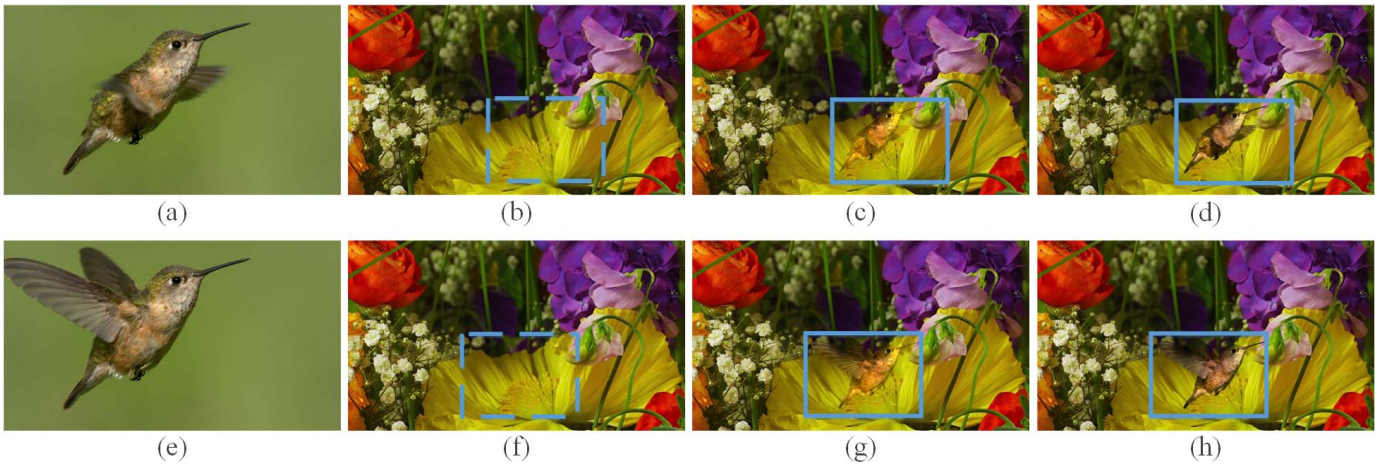


Fig. 3. An example on *hummingbird* showing the effectiveness of the additional smoothing term in Eq. (3). We intend to clone the hummingbird into the anthemy, and we show our strategy can achieve realistic results by adjusting the discrepancies between the source and target surroundings. The source and target frames are shown in (a), (e) and (b), (f) respectively. (c) and (g) show the blending results using [6]. In (d) and (h) we list the results using Eq. (3).

is computed as: $H_x = (1-\alpha_x)/\sum_{y\in\Omega}(1-\alpha_y)$, in which α_x and α_y are the matte value at point x and y, respectively. H_x tends to be large for points between $\mathbf{B}$ and $\mathbf{B}_{in}$, while relatively small for points inside $\mathbf{B}_{in}$. Hence, the effects of the source are restrained in region between $\mathbf{B}$ and $\mathbf{B}_{in}$ if κ is set to a positive value. $S(q,x)$ is designed based on the space distance between the points and is computed as: $S(q,x) = \exp(-||q-x||^2)$, which makes the points near q have greater influence to the interpolant so that the object appearance is well preserved. κ is a parameter controlling the effect of the term, and we set it to 0.08 in most cases in experiments.

Denote N as the number of pixels in Ω. The computational complexity of Eq. (3) can be reduced to $O(N)$ by pre-computing the summation of mattes in Ω. As the value of $S(q,x)$ will be negligible when the distance of point q and x is large, we can further cut down the computational complexity by only taking into account the neighborhood pixels around point q. We consider the points inside a 7×7 window of q rather than all points in Ω in the experiments, and the blending results is visually similar. Fig. 3 shows the effectiveness of our interpolant which is computed by Eq. (3). Compared to Farbman *et al.* [6], our blending results preserve more source patch information and look more realistic. In Fig. 4, the difference of brightness between the target is restrained, and the flower is adjusted to be harmonious with the target illumination condition compared to the results without smoothing.

We use the closed-form matting [11] to remove the artifacts in the uncertain region. We intend to compute the mattes of the source frame for mean-value interpolant, so that the boundary region smoothness and the structure transfer characteristics in matting are well maintained. The cost function for closed-form alpha mattes is:

$$E_m(\alpha) = (\alpha-\gamma)^T\Lambda(\alpha-\gamma) + \alpha^T\mathcal{L}\alpha \tag{4}$$

where, γ is the user-supplied constraint and we use the trimap as constraint in experiments. Λ is the diagonal matrix which consists of constraint values. In Eq. (4), $\mathcal{L}$ is the matting Laplacian matrix. The (i,j)-th element of $\mathcal{L}$ is chosen to be:

$$\mathcal{L}_{ij} = |\mathbf{w}_k|(\xi_{ij} - U_{ij}(I)) \tag{5}$$

where, ξ_{ij} is the Kronecker delta, and $U_{ij}(I)$ is computed as the kernel weight of the guided filter [42]. Compared to with Poisson-based blending methods, mean-value cloning can achieve similar blending results and faster computing speed, but it may lead to smudging pixels around blending boundary, especially when there is a fuzzy boundary around the definite foreground in source frame.

B. Discoloration Artifacts Removing

We can obtain a relatively natural composition result by using the smoothing term. However, the discoloration artifact often exists in some composition images, which appears as dramatic visual difference between the source region and the target background. It primarily derives from overflow of the color distribution in the process of mean-value cloning. To address the problem, conventional mean-value cloning limits the overflow intensity value to 255 in RGB channels. Discoloration artifacts often derive from immoderate color transferring. Based on this fact, we further constrain the color distribution in the interpolant in Eq. (3) to achieve moderate color transferring. Denote μ and δ as the mean intensity and standard deviation value of pixels in the source patch. We think the color distribution in the interval $[\mu-\delta, \mu+\delta]$ of the source are non-overflow, and intend to limit the variable range of mean of the composition to $[\delta, 255-\delta]$.

The composition image intensity $\mathbf{F}$ is computed by adding up $\mathbf{F}^s$ and the computed interpolant. We compute the intensity of the composition frame at point q as:

$$\mathbf{F}(q) = \mathbf{F}^s(q) + r_2(q)\cdot\zeta \tag{6}$$

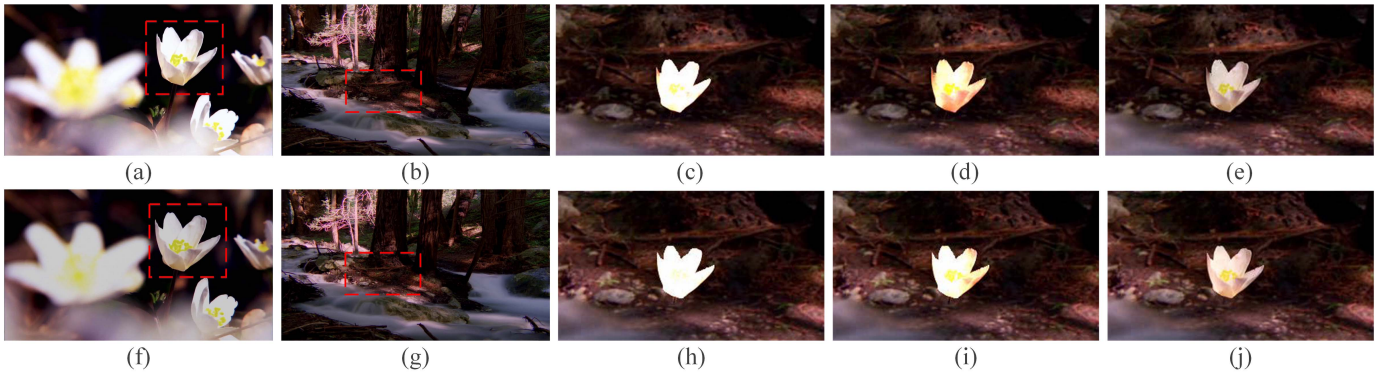


Fig. 4. An example of our color distribution constraining on *flower1* sequence. (a) and (f) are two selected source frames where we mark the source patch with red dashed rectangles. (b) and (g) are the target frames in which the region of interest is marked. (c), (h) and (d), (i) are the corresponding blended results using Eq. (1) and Eq. (3) as the interpolant for cloning, respectively. (e) and (j) are the cloning results which computes the mean-value interpolant with Eq. (4). The results of our method are harmonious with the target illumination condition, and discoloration artifacts are well reduced. Meanwhile the inter-frame color difference is restrained in the blending region as well.

where ζ is a coefficient to refine the interpolant in Eq. (3), and it is computed as:

$$\zeta = \begin{cases} \exp\left(\tau \cdot \left(\frac{\delta - \mu}{\bar{r}} - 1\right)\right), & \text{if } \mu + \bar{r} \in [0, \delta) \\ 1, & \text{if } \mu + \bar{r} \in [\delta, 255 - \delta] \\ \exp\left(\tau \cdot \left(\frac{255 - \mu - \delta}{\bar{r}} - 1\right)\right), & \text{if } \mu + \bar{r} \in (255 - \delta, 255] \end{cases} \tag{7}$$

where, $\bar{r}$ is the mean value of r_2 in the blending region $\mathbf{\Omega}$ and computed as: $\bar{r} = \frac{1}{N}\sum_{x \in \mathbf{\Omega}} r_2(q)$, τ is a controlling parameter and are set to 0.02 in experiments. If color intensity not overflows, which means $\mu + \bar{r} \in [\delta, 255 - \delta]$, $r(q)$ is computed as $r_2(q)$. Once $\mu + \bar{r}$ is out of the range $[\delta, 255 - \delta]$, the interpolant is constrained to control the immoderate overflow of color distribution. We use the new interpolant r for cloning, and Fig. 4 is an example showing the color distribution constraining, where we can see the constrained mean-value interpolant removes the discoloration artifacts effectively. Meanwhile, it is worth noticing that there is illumination intensity variation in the target or source scene, the color in the blending region may flicker over the time, and the limited color distribution can enhance the temporal consistency of composition. From Fig. 1, we can also find that color flickering in the blending region is weakened when there is illumination change. Note that our approach computes the mean-value interpolant in Eq. (1) and Eq. (3) in LUV color space, while we first transfer the color space into RGB channels to compute Eq. (6), and change the new interpolant r back to LUV as the interpolant which is used in Eq. (7). The blending boundary is optimized when tackling each frame as introduced in Section V-A, but it would be computationally unbearable to repeat computing the tangents of each pixel of the patch. We use the method in [8] to compute the mean-value interpolants approximately and the computational complexity is largely reduced by pre-computing the tangents values. The membrane evaluation at each vertex is performed independent to others. This allows us to provide a parallel implementation on a GPU.

V. Illumination-Aware Video Composition

The refined image cloning can composite frames seamlessly and plausibly when illumination varies. In this section, we introduce our adaptive composition. With the deformation of source patch and variation of illumination intensity, blending boundary should be modified automatically in each frame. Gradient domain mixing can achieve a seamless blending, but the brightness change may lead to temporal incoherency in blending region. We first introduce how to find a blending boundary, which maintains spatial-temporal consistency with the challenge of motion and illumination difference. Then we combine gradient fields of the source patch and target frames, which tackle sudden illumination changes.

A. Spatial-Temporal Consistent Boundary Computing

Blending boundary optimization is necessary for video composition because the inner and outer boundary is only given in keyframes. Pérez *et al.* [26] deal with problem of pasting a patch into the region of interest by solving a minimization problem with a guidance field to refine the user-defined blending boundary. Jia *et al.* [27] addressed the optimal boundary by solving the minimization equation in the Laplacian form instead of Poisson equations, and found that a smoother boundary with the minimal color mismatch (color intensity difference at each boundary points) has the least variational energy in solving the Laplacian equations. For the frames with fuzzy boundaries like dust or smoke etc. and motion blur, we determine a blending boundary between $\mathbf{B}_{in}$ and $\mathbf{B}_{out}$ which has the minimal color mismatch considering illumination change and similar optical flow between the source and target boundary pixels. The optimization step operates on super-pixels by applying over-segmentation for efficiency. This strategy effectively suppresses visible seam around the foreground patch and appearance variation. As introduced in Section IV, the blending region is divided into $\mathbf{R}_1$ and $\mathbf{R}_2$ according to Gabor feature vectors. The boundary in $\mathbf{R}_2$ can be generated using matting technique. Thus, we only need to optimize the boundary in $\mathbf{R}_1$.

Jia *et al.* [27] optimized the blending boundary using dynamic programming in order to minimize the color mismatch. GradientShop [43] minimized the color mismatch between the motion-compensated neighborhood pixels to maintain temporal coherence. Chen *et al.* [8] use optical flow to address the complex motion of boundary pixels. We define an energy function for boundary curve $\mathbf{B}$ as:

$$E_h(\mathbf{B}, h) = \sum_{p \in \mathbf{B}} \mathcal{S}(p, h) + \lambda_1 \sum_{p \in \mathbf{B}} \mathcal{N}(p, h) + \lambda_2 \sum_{p \in \mathbf{B}} \mathcal{W}(p, h) + \lambda_3 \sum_{p \in \mathbf{B}} \mathcal{V}(p) \quad (8)$$

The energy function consists of three color terms $\mathcal{S}(p, h)$, $\mathcal{N}(p, h)$, $\mathcal{W}(p, h)$ and a motion term $\mathcal{V}(p)$. λ_1, λ_2 and λ_3 are weighting parameters. These terms are designed to seek a spatial-temporal coherent boundary so that pixel values will be stably changed if there is a fuzzy boundary around the object.

The first term $\mathcal{S}(p, h)$ is designed to maintain spatial consistency, which is computed as:

$$\mathcal{S}(p, h) = \exp\left(-\epsilon \|\mathbf{h}_p - \mathbf{h}\|^2\right) \quad (9)$$

where $\mathbf{h}_p$ is the color mismatch vector of target and source at the location of a boundary point p, $\mathbf{h}$ is average color mismatch vector of all boundary points on current frame. Initially, the average color mismatch vector is set as the mean color difference at all points in $\mathbf{B}_{out}$. ϵ is a constant and is computed [40] as:

$$\epsilon = \left(2 \cdot \sum_{(p_1, p_2) \in \mathbf{U}_s} \|\mathcal{C}(p_1) - \mathcal{C}(p_2)\|\right)^{-1} \quad (10)$$

where $\mathbf{U}_s$ consists of all 8-neighboring point pairs within the same frame.

The second term of Eq. (8) is designed to maintain temporal consistency, and is computed as:

$$\mathcal{N}(p, h) = \exp\left(-\epsilon \|\mathbf{h}_p - \bar{\mathbf{h}}_p^n\|^2\right) \quad (11)$$

where $\bar{\mathbf{h}}_p^n$ is the average mismatch color vector between point p and the nearest boundary points on the two adjacent frames. When there is noticeable illumination variation, including partial appearance change caused by cast shadow and light intensity change in global scene, the second term in Eq. (8) is not practical. This issue may lead to temporal inconsistency in composition videos.

The third term $\mathcal{W}(p, h)$ in Eq. (8) is the energy term considering the inter-frame pixel intensity difference caused by illumination variation to enhance temporal coherency when illumination changes, and it is computed as:

$$\mathcal{W}(p, h) = e_1 \cdot \exp\left(-\epsilon \|\mathbf{h} - \mathbf{h}^n\|^2\right) + e_2 \cdot \exp\left(-\epsilon \|\bar{\mathbf{h}}_{p,w} - \bar{\mathbf{h}}_{p,w}^n\|^2\right) \quad (12)$$

where e_1 and e_2 are two 'toggle' parameters. When there is sudden illumination intensity change in the global target or source scene, e_1 is set to 1, otherwise 0. Similarly, when the source object appearance or the target blending region changes partially due to cast shadow, e_2 is set to 1, otherwise 0. $\mathbf{h}^n$ is denoted as the average color mismatch vector at all boundary points on the next frame. The first term of Eq. (12) penalizes the illumination variation in the global scene. $\bar{\mathbf{h}}_{p,w}$ and $\bar{\mathbf{h}}_{p,w}^n$ are the average color mismatch of all points inside the 3×3 window of point p on the current frame and the corresponding pixel obtained by optical flow in the next frame, respectively. The second term of Eq. (12) takes into account the fractional intensity variation of object appearance which appears when there is cast shadow or uneven light distribution in the background.

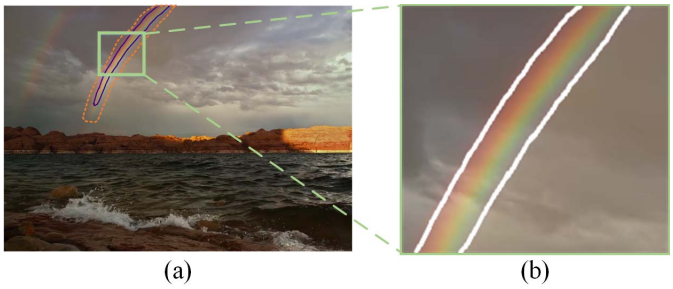


Fig. 5. Optimizing blending boundary on *rainbow* sequence. We intend to drag the rainbow as the source patch in (a), where the orange dashed loop curve is the initial outer boundary $\mathbf{B}_{out}$, and the dark blue curve encircles the definite foreground region. (b) is a zoom-in region of rainbow, where we show the optimized blending boundary with the white curve.

In the last term of Eq. (8), we optimize the temporal coherency of motion-compensated neighbor points with optical flow. $\mathcal{V}(p)$ is computed as:

$$\mathcal{V}(p) = \|\mathbf{v}_p^s - \mathbf{v}_p^t\|^2 \quad (13)$$

where $\mathbf{v}_p^t$ and $\mathbf{v}_p^s$ are optical flow vectors at point p in the current target and source frames, respectively. In Fig. 5(b), we determine a closed curve to be the blending boundary as shown with the white curve between $\mathbf{B}_{out}$ and $\mathbf{B}_{in}$. In the first frame, the initial blending boundary is set to $\mathbf{B}_{out}$. In Eq. (8), λ_1, λ_2 and λ_3 are set to 1, 5 and 0.6 in implementation. We solve the function iteratively and the maximum number of the iteration is set to 25 except that the boundary converges. With the optimized boundary in $\mathbf{R}_1$ and the boundary computed in $\mathbf{R}_2$, a loop curve is obtained as the final optimized boundary.

B. Illumination-Aware Gradient Fields Mixing

We mix the gradient of source patch and target frames between region of $\mathbf{B}$ and $\mathbf{B}_{in}$ to obtain a seamless composition result. We create a gradually mixed gradient field to remove motion inconsistent artifacts and immoderate color transferring in definite foreground region. The gradients at pixels near $\mathbf{B}_{in}$ in the composition image should be more affected by source frame, while the ones which locate besides $\mathbf{B}$ should depend more on the gradient values of target scene. Wang *et al.* [44], [45] proposed a framework for video editing in gradient domain by considering two constraints: spatial consistency and temporal coherence. At the i-th pixel of blending region, the new gradients are obtained by linear superposition of those of source and target frames. The naive form of gradient combination mixes gradients at each pixel

crosswise and lengthways with same parameter. Denote $\mathbf{G}^s$ and $\mathbf{G}^t$ as gradient vectors of source and target respectively, We denote $\mathbf{a}_i = (a_i^x, a_i^y)$ as the mixing weighting factor at pixel i, so that the mixing parameter is computed discrepantly. The gradient mixing parameter value approximates to 1 when the pixel locates near the inner boundary $\mathbf{B}_{in}$, and is close to 0 when the pixel position is around the determined blending boundary $\mathbf{B}$. The mixing gradients are the weighted sum of gradients in the source and target frames.

Denote $\mathbf{G}$ as the mixing gradients, and we compute the mixing gradients as: $\mathbf{G} = \mathbf{a}_i \circ \mathbf{G}^s + [(1,1) - \mathbf{a}_i] \circ \mathbf{G}^t$, where $\circ$ is the symbol of Hadamard product of matrix. In the first frame, we compute the initial mixing weight based on gradient magnitudes which is adjusted by optical flow and inter-frame color difference. We denote the initial mixing weight as $\mathbf{a}_i^{init} = (a_i^{init}, a_i^{init})$. The initial value for pixel i is computed as:

$$a_i^{init} = \frac{\|\mathbf{G}_i^s\|^2}{\|\mathbf{G}_i^s\|^2 + \|\mathbf{G}_i^t\|^2} + \eta(C^t - C^s) \tag{14}$$

where, $||\mathbf{G}_i^s||$ and $||\mathbf{G}_i^t||$ are the gradients of the source and target at location of pixel i, respectively. η is a parameter and we set it to 0.03 in the experiments. The initial weight is determined based on the ratio of L2-Norm gradients, and the first term gives more priority to the preservation of larger gradient magnitudes. The second term emphasizes color transfer which is different from backgrounds. We denote C^s as the difference of normalized histograms of the current and next source frame in Kullback-Leibler distance, and define C^t in the target video the same way. We denote the histogram in each bin in frame I_f as $c_f(b)$, where $b = 1...K$ and K is the number of bins. The histograms of the current and previous frame are normalized so $\sum c_f(b) = \sum c_{f-1}(b) = 1$. The Kullback-Leibler distance between the subsequent frames is:

$$C(I_f, I_{f-1}) = \sum_{b=1}^{K} c_f(b) \log \frac{c_f(b)}{c_{f-1}(b)} \tag{15}$$

If $C(I_f, I_{f-1})$ is small either for the source or the target frame, we assume that there is no significant illumination change between frame I_f and I_{f-1}. On the other hand, if $C(I_f, I_{f-1})$ is relatively great, it implies that the color cues are not reliable, thus the effects of the corresponding frame should be constrained in gradient mixing. We design the second term in Eq. (14) to reduce color flickering in the blending region.

In view of object deformation and occlusion, the mixing parameters in the subsequent frames are updated by solving the cost function. The mixing parameter is determined based on the initial value and adjusted to ensure feature preservation and smoothness. The cost function for mixing parameter updating is defined as:

$$\begin{aligned} E_G(\mathbf{a}_i) = {} & \left(1 + \omega_1 \cdot (C^t + C^s)\right) \cdot \|\mathbf{a}_i - \mathbf{a}_i^{init}\|^2 \\ & + \omega_2 \cdot \sum_{j \in \mathbf{U}_m(i)} \|\mathbf{a}_i - \mathbf{a}_j\|^2 + \omega_3 \cdot \sum_{k \in \mathbf{U}_n(i)} \|\mathbf{a}_i - \mathbf{a}_k\|^2 \end{aligned} \tag{16}$$

where, the first term requires that the mixing weights temporally stable and penalizes illumination variation in source and target, and we use Eq. (15) to measure inter-frame likelihood of color in the target video sequence. When illumination intensity varies, the gradient mixing parameter should be close to initial value to reduce color flickering in blending region. $\mathbf{U}_m(i)$ is the pixel set of temporal neighbors of pixel i computed by bi-directional optical flow. $\mathbf{U}_n(i)$ is the set of spatial 4-connected neighbors of pixel i. ω_1, ω_2 and ω_3 are controlling parameters, and are set to 0.05, 0.3 and 0.5 in experiments, respectively. The optimal mixing parameter $\mathbf{a}_i$ can be obtained in the linear time by seeking partial derivative of $\mathbf{a}_i$ in Eq. (14). Fig. 6 is an example of our composition method based on gradient mixing, where a jet aircraft is pasted into the target with complex illumination condition. Our method provides a seamless composition result, which is in accordance with the target light condition.

We have considered the temporal consistence between adjacent frames in the gradient domain. In many cases the input frames are inconsistent due to varying illumination or object deformation. We further interpolate the source patch and target image temporally by optimizing a cost function for all frames $I^{1...T}$. The cost function E_c to optimize is computed as:

$$\begin{aligned} E_c(I^t, I^c) = \sum_{f=1}^{T} \{ & E_b(I_f^t, I_f^c) + \phi_1 E_b(I_f^c, I_{f-1}^c) \\ & + \phi_2 E_b(I_f^c, I_{f+1}^c)\} \end{aligned} \tag{17}$$

where, T is the number of frames, I^t and I^c are the target frames and composition result frames respectively, ϕ_1 and ϕ_2 are two parameters which indicate the temporal coherency weight, and we set them both to 5 in our experiments. The first term is designed to help make the composition result similar to the target frame, so that the composition result is globally consistent. The latter two terms help ensure similarity between adjacent frames of composition frames. We use bidirectional similarity [46] as measurement between images I_1 and I_2 as $E_b(I_1, I_2) = E_L(I_1, I_2) + E_L(I_2, I_1)$. The second term helps the content of source patch which will appear in the target prevent converging to excessively smooth. E_L is posed as a patch-based energy function which makes the local region appear most similar to some local neighborhood regions:

$$E_L(I_1, I_2) = \Sigma_{p_1 \in I_1} \min_{p_2 \in I_2} (\mathcal{D}_{ssd}(P_1, P_2) + \lambda_D \mathcal{D}_{ssd}(\nabla I_1, \nabla I_2)) \tag{18}$$

where P_1 is a $w \times w$ local window with pixel p_1 at its center, and we define P_2 in the similar way. $\mathcal{D}_s$ is the sum of squared distance (SSD). We measure the dissimilarities between two windows with three color channels of LUV color space in the first term in Eq. (18), and use gradients of luminance in the second term. λ_D controls the contribution of luminance gradient dimensions. We chose the gradient weight $\lambda_D = 0.2$ for most of our cases. We refine the video composition results with Eq. (17) by iteratively optimizing it for 5 times.

VI. Experimental Results

We use several challenging video sequences with illumination differences as inputs for our video composition. The high-quality composition results are demonstrated by showing the results and composition results comparison with previous

Fig. 6. Gradient field mixing of our method on *jet aircraft* sequence. The illumination condition changes over time in the target. The source patch is zoomed out in the composition. From (a) to (d) are source frames, composition results using [8], [27] ours. A visible seam can be seen in (b) and (c). We create a spatial consistent composition results by applying our illumination-aware gradient mixing. The color flickering is also suppressed in the composition results.

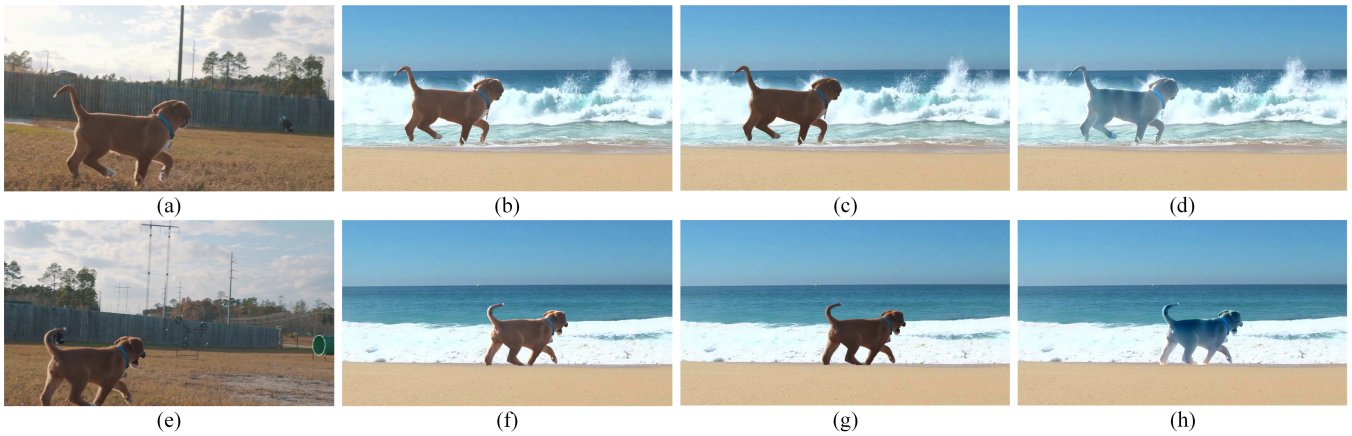


Fig. 7. The composition results comparison on *dog* sequence by setting different parameter on κ. We compose a running dog on a beach, where the brightness differs in the source and target scene. (a) and (e) are two input video frames from the video sequence about the source scene. We set κ to 0.08 in (b) and (f), 0.02 in (c) and (g), and 0.2 in (d) and (h). It can be observed that the composition effects are far from satisfied in (d) and (h). Composition results in (b) and (f) are more realistic compared to (c) and (g), as the brightness of the source object is consistent with target scene.

methods are also shown. The experiments are performed in a computer equipped with a 2.5GHz of Inter Core i5-3210M CPU, 16 GB memory and NVIDIA GeForce GT 740M.

A. Parameter Analysis

In the image cloning part, the controlling parameter of the smoothing term κ can notably influence the visual effects of composition results. We set it to 0.08 in most cases of our experiments. We set it relatively great when the brightness difference between the source and target scene is conspicuous, so that luminance of the blending region and the target scene is consistent. On the other hand, κ is set relatively small if the user intends to preserve the color and texture information of the source patch. Fig. 7 shows the results by setting κ to 0.02, 0.08 and 0.2, respectively. If κ is set small as in Fig. 7(c) and Fig. 7(g), the luminance discrepancy between the blending region and the target scene is obvious, thus the composition result is not globally consistent. On the contrary, κ is set too large in Fig. 7(d) and Fig. 7(h), the composition result is far from satisfying because it takes into account too much background color information.

In the blending boundary computing section, e_1 and e_2 are two toggle parameters which are set by the user. When there

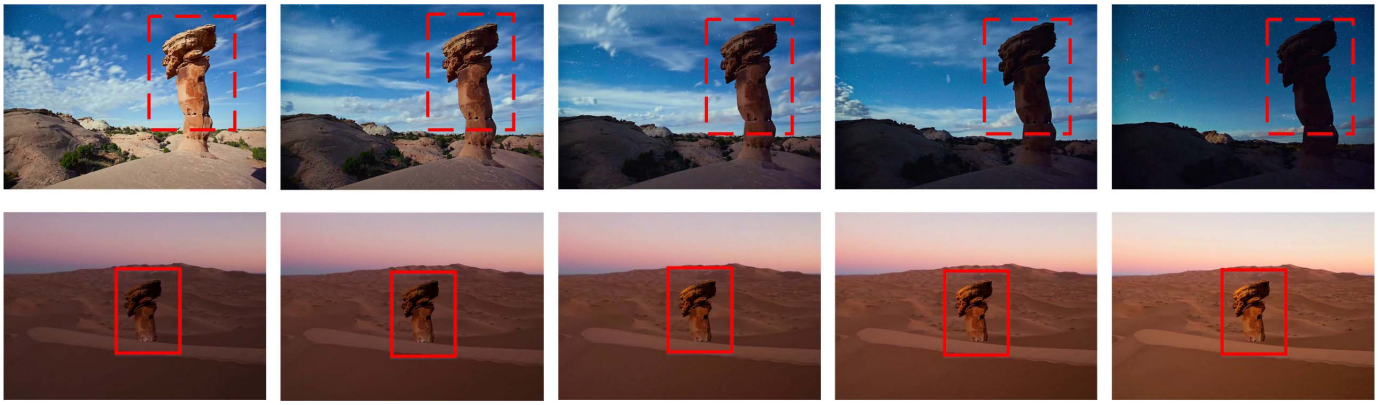


Fig. 8. An example of our video composition method on *desert* sequence. We list the source frames and the corresponding composition results in the first and the second row, which are selected from sunset and sunrise sceneries respectively. The source scene illumination becomes more intensive while the target illumination changes inversely. The appearance of the stone changes in accordance with the target scene brightness.

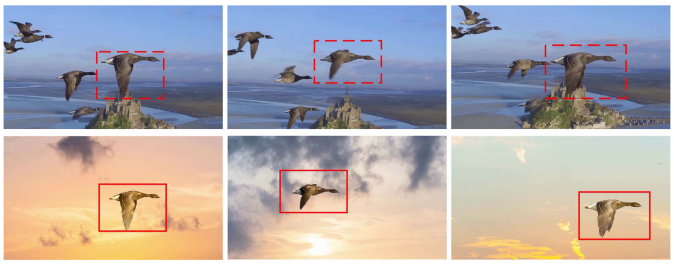


Fig. 9. Video composition results on *goose* sequence. There is complex motion for the goose and a fuzzy boundary exists around the feather of the goose. Meanwhile, the illumination condition changes over time in the target. Our composition results are shown in the second row, where the blended object is harmonious with the target illumination condition.

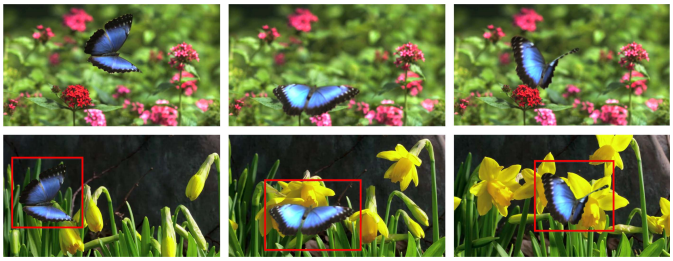


Fig. 10. A composition result on *butterfly* sequence. The foreground is a flying butterfly. The boundary of the object is fuzzy, and its motion is complex due to dramatic deformation. The background is a time-lapse video of flower blooming, and the flowers swaying over time due to the wind. Our method can achieve realistic and consistent composition results with the challenge of complex motion.

is no obvious illumination change in the source nor target video, they are both set to 2. As Fig. 8, e_1 is set to 1 because the global illumination changes in the target desert. On the other hand, in the outdoor scenarios where the illumination changes due to shadow, we set e_2 to 1. In Fig. 4 where we try to compose the flower under the shader of a tree, the toggle parameter e_2 should be set to 1.

B. Composition Quality

Fig. 8 shows the composition results by cutting out a stone in the sunset and paste it into the region of interest in target sunrise desert. The light intensity in the source scene decreases dramatically, while the target scene is becoming brighter which is opposite to the source frames. The object is marked by red dashed rectangles in the first row in Fig. 8, and the red solid rectangles shows the region of interest and the blending results inside. Illumination difference is the main challenge for this video, that is, the illumination varies inversely in the source patch and target video scenes. Thanks to our refined interpolant computing strategy, the blending region appearance harmonizes with the target brightness condition, which demonstrates the effectiveness to illumination variation of our method. In Fig. 1, the light intensity changes over time in source sequence, and there is luminance disaffinity inside the region of interest due to the cast shadow. Obviously, the brightness of appearance of the blending flower in the second row is in accordance with the shadows on the ground, which demonstrates that our approach is illumination-aware and can handle illumination intensity varying. There is challenge of deformation and motion blur in the source frames in Fig. 9. Meanwhile, the sunlight condition changes in the target scene. We utilize our illumination-aware gradient mixing strategy, which helps achieve seamless and globally consistent composition results. Fig. 10 shows the capability of our method to deal with complex motion in input video, where the source deformation and background motion is both complex.

Composition comparison results on *flower2* sequence are shown in Fig. 11. In Fig. 11(a), the input source patch is a blooming flower on the grassland, and the appearance of petal changes partially due to illumination variation. We also show the target sunset scene where the shadow expands gradually. From Fig. 11(b) to Fig. 11(e) are the outputs of copy and paste [47], enhanced use of matting [10], motion-aware [8], and ours. The composition results of state-of-the-art methods are not effective to tackle illumination change, including global light intensity variation and drop shadow. The composition video of our method is in accordance with the target illumination condition. Matting-based method [10] maintained the information in definite region well, but neglected the target color cues and the blending results are not natural. Similarly, the composition results of copy and paste [47] are also not in accordance with the varying global

Fig. 11. Blending results comparison on *flower2* which demonstrates the effectiveness to deal with illumination difference of our method. In (a), we use red dashed rectangles to mark the source patches in the left 3 columns, where the appearance of the blossoming flower changes due to illumination. The region of interest is also marked in the right 3 columns where the illumination condition changes due to cast shadow. From (b) to (e) are the results of copy and paste [47], matting-based method [10], motion-aware method [8] and ours. We show the composition results in the left 3 columns, and illustrate the corresponding zoom-in images on the right. We also use yellow rectangles to emphasize the region where our method outperforms other methods. The comparative experiment shows the effectiveness of our optimized mean-value blending approach, which makes composition results harmonious with the target surroundings.

illumination condition. Poisson-based methods like [5] achieve relatively good results like mean-value cloning, but the cloning speed is slow due to the large linear system to solve. Moreover, the amount of manual work is also unbearable. Chen *et al.* [8] preserve the texture in the blending region well. However, it still produces unrealistic results, where the brightness is not consistent with the target scene. The appearance of the flower should be darker under the shadow of the tree in Fig. 11. We follow the blending boundary optimization in [8], while further considering the illumination variation. This helps us obtain a temporal coherent boundary when the illumination condition changes. Moreover, our results are globally consistent in a single frame thanks to the proposed image cloning method. In Fig. 12, we give the composition comparison on *horse* sequence. We can see that matting-based video composition [10] gives unrealistic composition in uncertain region, like the shadow and tail. Copy and paste [47] removes the smudging pixels, but the color in the blending region is still not harmonious with the target surroundings, and the composition region is falsely severely affected by the background information. Visible smudging pixels can be seen in the fuzzy boundary region in the results of matting method [10]. Chen *et al.* [8] also addresses the issue by blending in the gradient domain. However, the method gives false optimized outer blending boundary when light intensity varies on the grass in Fig. 12. The shadow of the horse is not successfully composed due to the false blending boundary computing. The falsely blending shadow of the horse is manifested with blue rectangles in Fig. 12. Moreover, in the first column the appearance of the composed horse in Chen *et al.* [8] is not realistic. Our method achieves consistent blending results and compose the shadow region well with the challenge of source patch motion and background illumination varying.

TABLE I
PERFORMANCE OF OUR APPROACH

Sequence	***Frame number and resolution***	***MNBPF***	***NGT***	***PTPF***	***BTPF***
hummingbird	$52 \times 1920 \times 1080$	106851	11	24.2	1.7
horse	$80 \times 1920 \times 1080$	203574	18	25.1	3.7
stone	$70 \times 1600 \times 900$	54653	4	17.7	1.3
rainbow	$40 \times 956 \times 1348$	81734	7	14.1	1.8
jet aircraft	$53 \times 1200 \times 900$	238103	8	14.1	2.1
goose	$70 \times 1920 \times 1080$	94804	13	22.1	1.7
desert	$113 \times 1920 \times 1080$	357763	10	24.6	4.4
flower1	$57 \times 1920 \times 1080$	314465	15	25.1	3.8
flower2	$66 \times 1920 \times 1080$	468611	11	26.3	4.7

We further conduct a user study to compare the composition quality between the state-of-the-art and our method. We aim to investigate the effectiveness, enjoyability and fidelity of the methods. The study was carried out offline. We invited 40 participants to take part in the study by answering questionnaires. All the participants are undergraduate students. They are aged from 19 to 22 and do not major in computer science. And we assume they know nothing about our work. We only showed the composition video to them, while the source and target videos were not given to them. The evaluated videos are listed in Table I. These video sequences with the challenge of illumination difference, motion difference and fuzzy boundaries are tested as input data. The composition

Fig. 12. Composition results on *horse* sequence. We use red dashed rectangles to mark the region of interest in the right 3 columns in (a). The shadow of the horse deforms over time in the source, and the illumination condition changes on the grassland of the target. From (b) to (e) are the results of copy and paste [47], enhanced matting method [10], motion-aware method [8] and ours. We show the composition results in the left 3 columns, and illustrate the corresponding zoom-in images of the red rectangle region on the right. We also use blue rectangles to emphasize the region where our method outperforms other methods. It can be observed that our method can achieve natural blending results compared to (b) and (d). Our method can also obtain reasonable results of the shadow compared to (c) and (d) as marked using blue rectangles.

videos are evaluated by the users as "Good", "Acceptable" and "Failed". Each composition video sequence has around 20 frames and participants are required to evaluate each frame. We received 38 valid responses to the evaluated videos. We count the number of three levels and compute the proportion of them. We show the statistics of the user study in Fig. 13. It can be observed that our method outperforms other methods in enjoyability and fidelity when dealing with videos with illumination difference and other challenges.

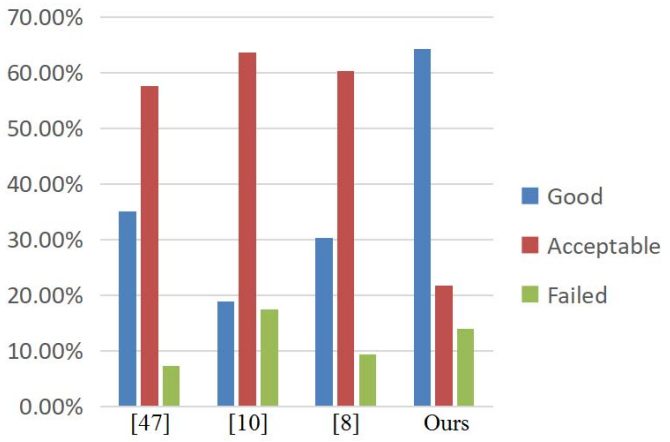


Fig. 13. The user study for evaluating the composition quality by copy and paste [47], enhanced matting method [10], motion-aware method [8], and our approach.

C. Composition Efficiency

Table I shows more details for the composition implementation of our approach, where MNBPF, NGT, PTPF and BTPF are the abbreviation of mean number of blended pixels per frame, number of given trimaps, preprocessing time per frame and blending time per frame (seconds), respectively. Preprocessing part consists of dense optical flow estimating bidirectionally, foreground segmentation in source video, in which the bidirectional optical flow computing occupies the most time, especially dealing with the frames with high resolution. For the *hummingbird* sequence, preprocessing part takes up 24.2 seconds, in which optical flow computing consumes 22.7 seconds. We mix the gradient field of source patch and target video then perform cloning. More user supervision of initial inner and outer boundaries in keyframes should be provided when the source patch has fuzzy object boundaries or the boundaries generated by propagation are severely wrong like *jet aircraft* and *horse* sequence. Compared to frame-by-frame methods, Chen *et al.* [8], Xie *et al.* [7] and ours are more efficient in timing performance, because user supervision is not required in all frames, while the previous methods such as Farbman *et al.* [6] needs dense and accurate trimap as

necessary input. The solving procedure of mean-value coordinates interpolation can be computed parallelly, so we perform it on GPU to accelerate image cloning, and we pre-compute tangents approximately to further speed up cloning patch into target frame. Poisson blending based methods like [5] need to solve a large system of linear equations, they are still computationally in disadvantage by solving the problem with large sparse linear equation solving tools like TAUCS.

D. Limitation

As discussed in Section VI-B, our method can achieve satisfying performance when processing on many challenging videos, but may fail when the texture of the source patch and the target region of interest are not consistent. Moreover, when source object and target surrounding motion is hard to estimate, more user supervision is required to correct the computed blending boundary. This needs a further study.

VII. Conclusion and Future Work

In this paper, we introduce a new illumination-guided video composition in gradient domain, which tackles the challenges of illumination intensity varying. The user-defined inner and outer blending boundaries in the source video and the region of interest in the target are required on keyframes in interaction steps. We mix the gradient fields of source patch and target video inside blending boundary, which is optimized iteratively from user-provided boundaries. On the basis of mixed gradient field, we further propose an effective implementation of mean-value coordinates interpolation, which approximately computes mattes first, then smooths the differences between source and target. And it constrains overflow in color distribution. The strategy enables the effectiveness to illumination condition changing and fuzzy object boundaries. In the future, we will work on illumination-invariant features extractions like intrinsic image decomposition to tackle illumination differences. Another possible future work is to extend the saliency cues into foreground object extraction and spatial consistency maintaining in the process of composition. Occlusion handling strategies are critical for video composition and editing with good performance, we will also investigate and incorporate latest occlusion-aware detection strategies [48], [49] to open our method for more general video datasets as our future work.

References

[1] K. Li, S. Li, S. Oh, and Y. Fu, "Videography-based unconstrained video analysis," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2261–2273, May 2017.

[2] X. Li, B. Zhao, and X. Lu, "A general framework for edited video and raw video summarization," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3652–3664, Aug. 2017.

[3] A. Kamel, B. Sheng, P. Yang, P. Li, R. Shen, and D. D. Feng, "Deep convolutional neural networks for human action recognition using depth maps and postures," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.

[4] A. Karambakhsh, A. Kamel, B. Sheng, P. Li, P. Yang, and D. D. Feng, "Deep gesture interaction for augmented anatomy learning," *Int. J. Inf. Manage.*, vol. 45, pp. 328–336, Apr. 2019.

[5] H. Wang, R. Raskar, and N. Ahuja, "Seamless video editing," in *Proc. Int. Conf. Pattern Recognit.*, 2004, pp. 858–861.

[6] Z. Farbman, G. Hoffer, Y. Lipman, D. Cohen-Or, and D. Lischinski, "Coordinates for instant image cloning," *ACM Trans. Graph.*, vol. 28, no. 3, p. 67, 2009.

[7] Z.-F. Xie, Y. Shen, L.-Z. Ma, and Z.-H. Chen, "Seamless video composition using optimized mean-value cloning," *Vis. Comput.*, vol. 26, nos. 6–8, pp. 1123–1134, 2010.

[8] T. Chen, J.-Y. Zhu, A. Shamir, and S.-M. Hu, "Motion-aware gradient domain video composition," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2532–2544, Jul. 2013.

[9] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5933–5942, Dec. 2016.

[10] W. Wang, P. Xu, X. Bie, and M. Hua, "Enhanced use of mattes for easy image composition," *IEEE Trans. Image Process.*, vol. 25, no. 10, pp. 4608–4616, Oct. 2016.

[11] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 228–242, Feb. 2008.

[12] J. Wang and M. F. Cohen, "Image and video matting: A survey," *Found. Trends. Comput. Graph. Vis.*, vol. 3, no. 2, pp. 97–175, Jan. 2007.

[13] M. Gong, Y. Qian, and L. Cheng, "Integrated foreground segmentation and boundary matting for live videos," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1356–1370, Apr. 2015.

[14] L. Wang, M. Gong, C. Zhang, R. Yang, C. Zhang, and Y.-H. Yang, "Automatic real-time video matting using time-of-flight camera and multichannel Poisson equations," *Int. J. Comput. Vis.*, vol. 97, no. 1, pp. 104–121, 2012.

[15] M. McGuire, W. Matusik, H. Pfister, J. F. Hughes, and F. Durand, "Defocus video matting," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 567–576, 2005.

[16] H. Li, K. N. Ngan, and Q. Liu, "FaceSeg: Automatic face segmentation for real-time video," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 77–88, Jan. 2009.

[17] L. Karacan, A. Erdem, and E. Erdem, "Alpha matting with KL-divergence-based sparse sampling," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4523–4536, Sep. 2017.

[18] Y.-Y. Chuang, A. Agarwala, B. Curless, D. H. Salesin, and R. Szeliski, "Video matting of complex scenes," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 243–248, 2002.

[19] Y. Li, J. Sun, and H.-Y. Shum, "Video object cut and paste," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 595–600, 2005.

[20] X. Bai and G. Sapiro, "A geodesic framework for fast interactive image and video segmentation and matting," in *Proc. IEEE ICCV*, Oct. 2007, pp. 1–8.

[21] D. Li, Q. Chen, and C.-K. Tang, "Motion-aware KNN Laplacian for video matting," in *Proc. IEEE ICCV*, Dec. 2013, pp. 3599–3606.

[22] A. Levin, A. Rav Acha, and D. Lischinski, "Spectral matting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1699–1712, Oct. 2008.

[23] C. Xiao, M. Liu, D. Xiao, Z. Dong, and K.-L. Ma, "Fast closed-form matting using a hierarchical data structure," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 1, pp. 49–62, Jan. 2014.

[24] Y. Lee and S. Yang, "Parallel block sequential closed-form matting with fan-shaped partitions," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 594–605, Feb. 2018.

[25] Z. Farbman, R. Fattal, and D. Lischinski, "Convolution pyramids," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 175:1–175:8, Dec. 2011.

[26] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 313–318, 2003.

[27] J. Jia, J. Sun, C.-K. Tang, and H.-Y. Shum, "Drag-and-drop pasting," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 631–637, Jul. 2006.

[28] W. Wang, J. Shen, and L. Shao, "Consistent video saliency using local gradient flow optimization and global refinement," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4185–4196, Nov. 2015.

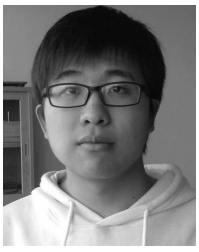
[29] W. Wang, J. Shen, Y. Yu, and K.-L. Ma, "Stereoscopic thumbnail creation via efficient stereo saliency detection," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 8, pp. 2014–2027, Aug. 2017.

[30] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, Jan. 2018.

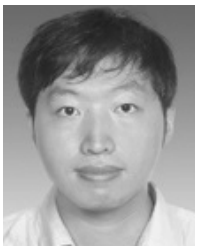
[31] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *Proc. ECCV*, 2004, pp. 377–389.

[32] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2photo: Internet image montage," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 124:1–124:10, Dec. 2009.

[33] W. Wang, J. Shen, F. Porikli, and R. Yang, "Semi-supervised video object segmentation with super-trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 985–998, Apr. 2019.
[34] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 38–49, Jan. 2018.
[35] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1451–1462, Apr. 2014.
[36] W. Wang, J. Shen, X. Li, and F. Porikli, "Robust video object cosegmentation," *IEEE Trans. Image Process.*, vol. 24, no. 10, pp. 3137–3148, Oct. 2015.
[37] Q. Chen, M. Wang, Z. Huang, Y. Hua, Z. Song, and S. Yan, "VideoPuzzle: Descriptive one-shot video composition," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 521–534, Apr. 2013.
[38] X.-Y. Li and S.-M. Hu, "Poisson coordinates," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 2, pp. 344–352, Feb. 2013.
[39] F.-L. Zhang, X. Wu, H.-T. Zhang, J. Wang, and S.-M. Hu, "Robust background identification for dynamic video editing," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 197:1–197:12, 2016.
[40] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.
[41] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video SnapCut: Robust video object cutout using localized classifiers," *ACM Trans. Graph.*, vol. 28, no. 3, p. 70, 2009.
[42] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.
[43] P. Bhat, C. L. Zitnick, M. F. Cohen, and B. Curless, "GradientShop: A gradient-domain optimization framework for image and video filtering," *ACM Trans. Graph.*, vol. 29, no. 2, pp. 10:1–10:14, 2010.
[44] H. Wang, N. Xu, R. Raskar, and N. Ahuja, "Videoshop: A new framework for spatio-temporal video editing in gradient domain," in *Proc. IEEE CVPR*, vol. 2, Apr. 2005, p. 1201.
[45] H. Wang, N. Xu, R. Raskar, and N. Ahuja, "Videoshop: A new framework for spatio-temporal video editing in gradient domain," *Graph. Models*, vol. 69, no. 1, pp. 57–70, 2007.
[46] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," in *Proc. IEEE CVPR*, Jun. 2008, pp. 1–8.
[47] Z. Wang, X. Chen, and D. Zou, "Copy and paste: Temporally consistent stereoscopic video blending," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 3053–3065, Oct. 2018.
[48] X. Dong, J. Shen, D. Yu, W. Wang, J. Liu, and H. Huang, "Occlusion-aware real-time object tracking," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 763–771, Apr. 2016.
[49] J. Shen, D. Yu, L. Deng, and X. Dong, "Fast online tracking with detection refinement," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 162–173, Jan. 2018.



Jingye Wang received the B.Eng. degree in computer science from the East China University of Science and Technology, Shanghai, China. He is currently with Shanghai Jiao Tong University, Shanghai, and also with the East China University of Science and Technology, Shanghai. His current research interests include illumination-aware video composition, image/video processing, and computer vision.



Bin Sheng received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong. He is currently an Associate Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include image-based rendering, machine learning, virtual reality, and computer graphics.

Ping Li received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong. He is currently with The Hong Kong Polytechnic University, Hong Kong. He has one image/video processing national invention patent, and has excellent research project reported worldwide by *ACM TechNews*. His current research interests include image/video stylization, GPU acceleration, and creative media.

Yuxi Jin received the B.Eng. degree in software engineering from Henan University, Kaifeng, China, and the M.Eng. degree from the East China University of Science and Technology, Shanghai, China. She is currently pursuing the Ph.D. degree in computer science with the Faculty of Information Technology, Macau University of Science and Technology, Macau, China. Her current research interests include image stylization, machine learning, and computer graphics.

David Dagan Feng (F'03) received the M.Eng. degree in electrical engineering and computer science (EECS) from Shanghai Jiao Tong University, Shanghai, China, in 1982, and the M.Sc. degree in biocybernetics and the Ph.D. degree in computer science from the University of California, Los Angeles (UCLA), Los Angeles, CA, USA, in 1985 and 1988, respectively, where he received the Crump Prize for Excellence in Medical Engineering. He is currently the Head of the School of Information Technologies, the Director of the Biomedical & Multimedia Information Technology Research Group, and the Research Director of the Institute of Biomedical Engineering and Technology, The University of Sydney, Sydney, NSW, Australia. He has published more than 700 scholarly research papers, pioneered several new research directions, and made a number of landmark contributions in his field. More importantly, however, is that many of his research results have been translated into solutions to real-life problems and have made tremendous improvements to the quality of life for those concerned. He has served as the Chair for the International Federation of Automatic Control (IFAC) Technical Committee on Biological and Medical Systems, has organized/chaired more than 100 major international conferences/symposia/workshops, and has been invited to give more than 100 keynote presentations in 23 countries and regions. He is a fellow of the Australian Academy of Technological Sciences and Engineering.