Research Article

Color retargeting: interactive time-varying color image composition from time-lapse sequences

Shao-Ping Lu[1] (✉), Guillaume Dauphin[2], Gauthier Lafruit[2], and Adrian Munteanu[1]

Abstract In this paper, we present an interactive static image composition approach, namely *color retargeting*, to flexibly represent time-varying color editing effect based on time-lapse video sequences. Instead of performing precise image matting or blending techniques, our approach treats the color composition as a pixel-level resampling problem. In order to both satisfy the user's editing requirements and avoid visual artifacts, we construct a globally optimized interpolation field. This field defines from which input video frames the output pixels should be resampled. Our proposed resampling solution ensures that (i) the global color transition in the output image is as smooth as possible, (ii) the desired colors/objects specified by the user from different video frames are well preserved, and (iii) additional local color transition directions in the image space assigned by the user are also satisfied. Various examples have been shown to demonstrate that our efficient solution enables the user to easily create time-varying color image composition results.

Keywords Time-lapse sequence; image composition; color retargeting; time-varying transition; optimized interpolation field.

1 Introduction

Time-lapse video sequences capture rich visual information in the scene, including not only temporal

---

1 Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB) and iMinds, Pleinlaan 2, Brussels, Belgium. E-mail: splu@etro.vub.ac.be (✉), acmuntea@etro.vub.ac.be.

2 Department of LISA, unit Image, Université Libre de Bruxelles (ULB), Belgium. E-mail: gdauphin@ulb.ac.be, glafruit@ulb.ac.be.

Fig. 1 Time-varying paintings by artists. Images are from @Four Seasons Tree Nexus.

motion of objects but also time-varying color evolution. With the wide popularization of numerous digital cameras, currently time-lapse videos can be easily generated by domestic consumers. Creative applications on further exploiting and processing such time-lapse videos attract a large number of artists and scientific researchers.

Editing and composing the time-varying color information into a static image, as will be shown in this paper, is among such applications. There are various artistic works that go in this direction. For instance, the components of an object are taken at different time instances and combined in the final object (see Fig. 1 the Four Seasons Tree Nexus). Another example is the day to night transition in a scene [30]. However, such artistic works require a very tedious interaction using existing image and video editing tools, calling for the complicated extraction of different color layers as well as their seamless composition.

In this paper, we introduce an efficient time-varying color image composition solution based on time-lapse videos. Inspired by image retargeting works [25, 26], we name our time-varying color composition as *color retargeting*. In contrast to image retargeting, where the image is spatially resampled in a content-
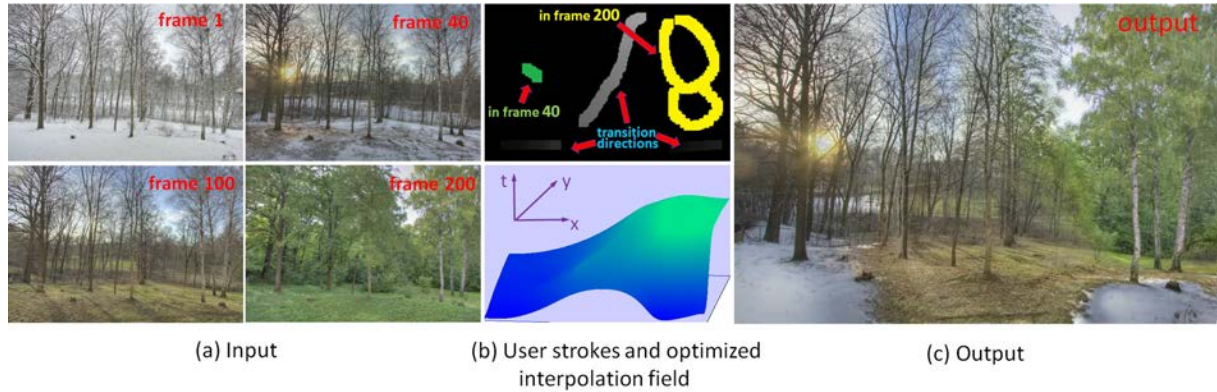
Fig. 2   Color retargeting with the proposed system. The user can easily compose and edit a time-varying color image with several strokes on a time-lapse sequence. Our system efficiently achieves globally smooth color transition when satisfying the user's requirements of preserving colors/objects and local color transition directions.

dependent manner, our work is concerned with the color resampling of the spatio-temporal pixels from the input time-lapse video cubes. We suppose that the color variance is continuous in the temporal direction, and the output image can be composed by copying (or weighted summing) pixels which are at the same spatial position but may be originating from different video frames. In order to match the user's requirements of both preserving specified colors/objects and satisfying some desired local color transition directions in the image space, we construct a globally optimized interpolation field for resampling of all pixels from the input time-lapse video. With the proposed interpolation field, our system achieves globally smooth color transition for all pixels, and thus effectively avoids visual artifacts.

Our approach provides the user a novel but also friendly composition solution for color images. With our system the user can easily produce time-varying editing results (see Fig. 2 for an example). We note that achieving such results with other existing editing tools would need much more tedious interaction. We believe that our efficient editing framework will further motivate more interesting applications on time-lapse videos.

2   Related work

Our color retargeting approach draws from numerous techniques developed for image composition, image recoloring and time-lapse video based editing. In the following, we focus on the contributions most related to the above-mentioned research domains.

Image composition. The pioneering work by Perez *et al.* [22] performs image composition by solving the Poisson equation. Its numerical result aims to find seamless filling with some selected content under given boundary conditions. Agrawala *et al.* [2] propose a well-known digital PhotoMontage framework, where both the graph-cut optimization [6] and gradient-domain fusion [14] are constructed to avoid visible seams. Pritch *et al.* [23] also introduce a graph-cut based shift-map optimization to minimize the gradient discontinuities of all pixels in the output image. In order to efficiently achieve seamless composition between the source and the target patches, Farbman *et al.* [13] propose a harmonic-like interpolation scheme using mean-value coordinates. Darabi *et al.* [10] introduce the image melding concept to further model potential geometric and photometric transformations when blending different images with inconsistent color and texture properties. We note that both the continuous and the discrete rescaling of image resolution, such as the famous seam carving [4, 25], image retargeting [17, 26] and other related applications [34], are well investigated. In contrast, our color retargeting work focuses on continuous color rescaling for image composition.

Image recoloring. To avoid precise segmentation and propagate the desired color to different video frames, Levin *et al.* [16] formulate a color intensity optimization model for space-time neighboring pixels. Such an approach can effectively achieve the recoloring effect by interactive editing. Another edge-preserving energy minimization method [18] is introduced for locally adjusting tonal values in digital photographs. Recoloring is also applied for editing the appearance of materials in all-pairs propagation [3] and its acceleration model [32]. A variety of solutions have been proposed by further considering the feature space of pixels, such as diffusion distance for the

global distribution [12], locally linear embedding (LLE) for local manifold preservation [8] and sparse control samples for the propagation influence [33]. The literature also offers other recoloring-based applications, such as color correction for multiview videos [7, 31].

Time-lapse video editing. In the last decades, researchers have also been interested in means to further exploit and edit visual information from time-lapse videos. The PhotoMontage approach [2] aims to easily compose the interesting objects into a single image, while it is unapplicable for a long-term video with hundreds of frames. Sunkavalli *et al.* [28] focus on the modeling and editing of shadow, illumination and reflectance components under clear-sky conditions. Another interesting application is motion-oriented composition. Bennett *et al.* [5] investigate the video rearrangement by frame-level resampling. Dynamic narratives [9] enables the user to interactively represent the motion in a static background. In [24], the desired content can also be resembled by 4D min-cut based optimization. After extracting the motion objects, Lu *et al.* [20] efficiently achieve condensing and other manipulations. From a single image, Shih *et al.* [27] generate the same scene at a different time of day using a database of time-lapse videos. Oppositely, Estrada *et al.* [11] pursuit the long-exposure effect in a static image by frame-level color resampling. Latest interesting applications of time-lapse video editing can also be seen on decomposing paintings into different layers [29] and time-lapse generation from internet photos [21]. Our proposed approach differs from such motion objects based editing in the sense that we compose the scene by considering the color consistency within a time-lapse image set or a video clip.

3 Problem formulation

Formally, suppose there are N video frames in the input time-lapse sequence, where we denote the nth frame as I_n, where $0 \leqslant n < N$. The composed color image is then represented as G. Accordingly, pixels in the input and output images are $I_n(\vec{x})$ and $G(\vec{x})$, where $\vec{x}$ is the pixel coordinate in the image space. In principle, pixels in G are resampled from those being at the same position (or co-located) in the input sequence. That is, we need to construct a mapping function to satisfy that

$$f : (I_0 \cdots I_N) \mapsto G. \tag{1}$$

The proposed color retargeting approach is detailed in the following section.

4 Color retargeting solution

4.1 Overview

The input to our approach is a time-lapse video sequence, and the output is a single edited color image that optimally represents the time variance information guided by the user interaction. The proposed framework is illustrated in Fig. 3. Firstly, the original video frames are preprocessed with frame-level color consistency correction and background panorama based scene alignment. Secondly, in the interaction phase, the user can 1) draw some desired areas (or objects) of any input frame that should be preserved in the output image, and/or 2) specify some local time variant directions that the output color image should follow. Next, our global optimization algorithm automatically computes the smooth interpolation field to satisfy the user's expectation. Finally, according to such globally smooth interpolation field, the output image is generated by pixel-level resampling from the input time-lapse video cube.

4.2 Preprocessing

In our solution, we suppose the input time-lapse videos are captured by static cameras. To this end, the input video could be simply aligned by homography parameters to match a reconstructed background panorama [20]. We note that the time-lapse videos should follow consistency of appearance in the temporal direction. That is, the colors of each frame should continuously evolve in the temporal direction. However, individual photos or frames in time-lapse sequences easily suffer from different exposure and lighting conditions. In order to ensure such color consistency in frame-level evolution, each frame is directly color corrected using histogram matching, where the targeted histogram is obtained by cubic interpolation under a sliding window consideration. Note that in our work we only perform such color correction for long-term video sequences (i.e. more than 100 frames), in which the frame-level color evolution is relatively slow. Alternatively, one could employ more complicated temporal color transfer models (*e.g.* [15] or our approach in [19]) to further improve the color consistency.

4.3 Optimized interpolation field

As mentioned before, our system needs to compute a good mapping between the input video and the output image. For convenience, we define here an interpolation field $\mathcal{L}$, whose corresponding value at each pixel is between 0 and N. By means of this interpolation field,
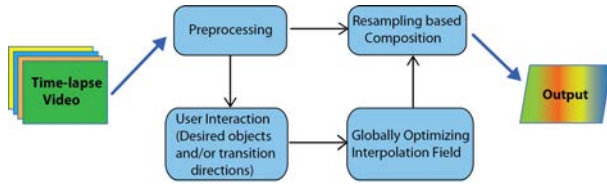
each each pixel in G receives a label indicating the input frame from which it should be duplicated. Here we consider the following three constraints to construct a globally optimized interpolation field.

Firstly, in order to keep smooth transition for all pixels over the output image, we force the resampling only over neighboring pixels from the same or neighboring input frames. More precisely, we formulate it as a globally smooth gradient constraint, by which the gradient is zero everywhere,

$$\Delta \mathcal{L} = 0, \qquad (2)$$

where Δ is the Laplace operator.

Secondly, for the areas covered by the user's strokes on specified frames, their colors in the output image should minimize the following energy

$$\sum_{\vec{x},\mathcal{M}(\vec{x})=n} \omega_a(G(\vec{x}) - I_n(\vec{x})), \ \forall n, 1 \leqslant n \leqslant N_a \qquad (3)$$

where $\mathcal{M}(\vec{x})$ is a subset of $\mathcal{L}$ and represents the objects/areas to be preserved, ω_a is the weighting factor. N_a is the sum of the user's strokes for specifying areas or objects.

Thirdly, to satisfy local color transition directions, which are assigned by the user, we consider that in the output image the following equation should also be minimized:

$$\sum_{\vec{x},\mathcal{N}(\vec{x})=n} \omega_d(G(\vec{x}) - I_n(\vec{x})), \ \forall n, 1 \leqslant n \leqslant N_d \qquad (4)$$

where $\mathcal{N}(\vec{x})$ represents the desired frame number from which the output color is sampled at pixel $\vec{x}$. Note that in $\mathcal{N}$ all pixels are on fitted splines of local color transition directions specified by the user. $\mathcal{N}$ is also a subset of $\mathcal{L}$. The weighting factor ω_d corresponds the optimization energy of this constraint. N_d is the sum of the local transition directions.

Considering the above-mentioned three constraints, our color retargeting problem is to find the smooth interpolation field $\mathcal{L}$ according to two given subsets $\mathcal{M}$ and $\mathcal{N}$. Moreover, this problem can easily be reformulated as a typical linear system, its global optimization solution is then the expected time-varying interpolation field for the color resampling of the time-lapse video.
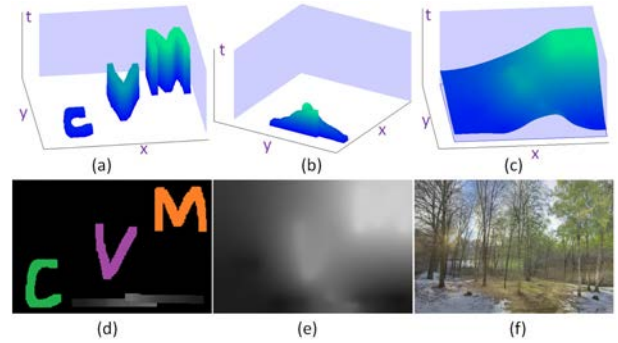

Fig. 4   Globally optimized interpolation field. Sub-figure (a) shows the areas/objects in different input frames that the user aims to preserve. (b) shows specified local color transition directions. (e) and (c) are the optimized interpolation field and its visualization version. (d) and (f) are all interactive strokes and the output image respectively.

4.4   Postprocessing

Until now, we have described our color resampling algorithm as a one-to-one direct pixel copying between the input video and the output image. As a result, we found that the output image may easily suffer from visual artifacts. This problem is mainly caused by 1) the imperfect temporal color consistency, and 2) the frame-level discretization errors of the temporal color transition. Therefore, inspired by the exposure fusion processing in [11], we generate each output pixel by linear combination of multiple neighboring temporal pixels, after obtaining the optimized interpolation field $\mathcal{L}$. That is, for the pixel at $\vec{x}$ position, if its resampling index in the optimized interpolation field is $\mathcal{L}(\vec{x}) = n$, the corresponding output color is computed as

$$G(\vec{x}) = \sum_{i=-R}^{R} I_{n+i}(\vec{x})\mathcal{G}(n, n+i), \qquad (5)$$

where R is the resampling radius in the temporal direction. $\mathcal{G}(a, b)$ denotes the discretization value at position b of the normalized Gaussian function centered over position a.

4.5   User interface

We also developed an interface to enable the user's quick interaction. Firstly, the user can directly draw strokes on any frame to select some desired areas or objects, where the content should be well preserved in the output image. Secondly, the system supports the user to address local color transition directions. To achieve this function, we employ cubic spline interpolation to fit such transition directions drawn by the user, and record pixels on this spline as the desired resampling values. After that, we perform
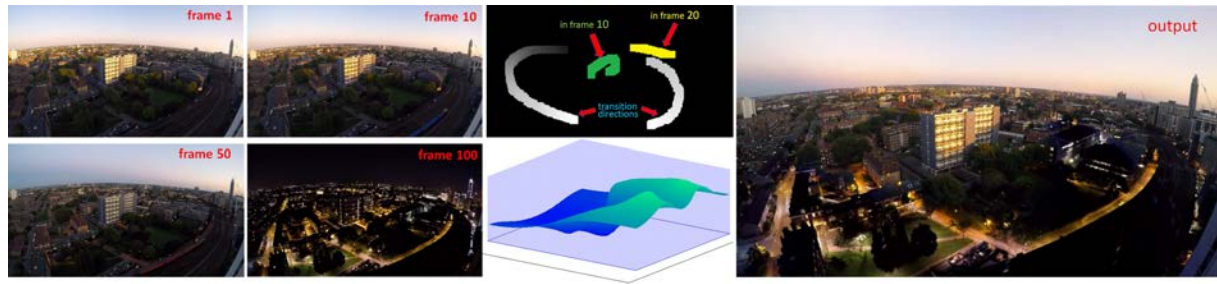
Fig. 5 Composition result for the *London* sequence.

our optimization solution to finish the color image composition. It should be noted that the user can easily refine the optimized interpolation field by further interaction, and thus the edited results can also be easily refined under our optimization framework.

5 Results

5.1 Implementation details

The proposed approach was implemented in Microsoft Visual Studio C++ 2010 on a high performance laptop with 2.3 GHz Quad-Core Intel-i7 CPU and 8 GB memory. The proposed system, including the user interaction and interpolation field optimization are performed with interactive speed, and thus the composition result can be instantly generated. For the parameters setting, we empirically choose ω_a and ω_b as 100, which means that the target colors in the specified areas defined by the user are given more important priorities than the globally smooth gradients constraint in the interpolation field. The temporal sampling radius is set as $R = 10$ in our implementation. With such parameters, the solution gives good results. We use the Eigen *et al.* [1] to efficiently solve the proposed linear system. In order to accelerate this processing, we first obtain the low-resolution version of the optimized interpolation field with a factor of 4 in each spatial direction, and recover it with bicubic interpolation. It ensures the real-time interactivity in our released implementation.

5.2 Examples

We have performed our approach on a variety of time-lapse video sequences. For instance, In Fig. 2 we present the color transition result changing from spring to winter. The left two columns are composed by the *1*st, *40*th, *100*th and *200*th input frames, respectively. To depict the time transition in one single image, we simply draw several strokes (see the sub-figures in the third column). In particular, we aim to preserve the color of the sun in the *40*th frame and the green tree in the *200*th frame, but also specify several other local horizontal and vertical color transition directions. Accordingly, the proposed global optimization solution computes the smooth interpolation field (see middle column, bottom part), and the composed image is obtained as shown in the right sub-figure of Fig. 2. One can observe that both the expected sun and the green tree are well preserved, while the ground and other areas are also smoothly changed from spring to winter following the user's expectation.

In Fig. 5 we show the *London* example. As it can be seen from the left two columns, the lighting is continuously varying from afternoon to night. By several interactive strokes, the shining textures of the tall building, captured during the day time, are well preserved in the output image. Moreover, according to the local color transition directions draw by the user, the expected time variance (i.e. color transition) from far and near also takes place smoothly. It demonstrates that our solution can effectively avoid visual artifacts for the color transition between the shining areas and the dark parts in the output image.

We also show more color retargeting examples in Fig. 6−8. In Fig. 6 and Fig. 7, we show the day to night blending effect for the *Philadelphia* and *Singapore* examples. With our system, the user can easily change the color of different scene content without obvious visual artifacts. To further investigate the effectiveness of our resampling algorithm on extremely short video clips, we also perform our approach with only four input images for the *Jungle* example in Fig. 8. In this case it is obvious that the bridge, the trees and the water are clearly different among input frames. Interestingly, various specified scene areas/objects are elegantly blended following the desired local color transition directions. This thus demonstrates that the global smooth gradient constraint in our optimization model works well to eliminate the color gaps for even extremely short videos where the recorded colors are obviously different.
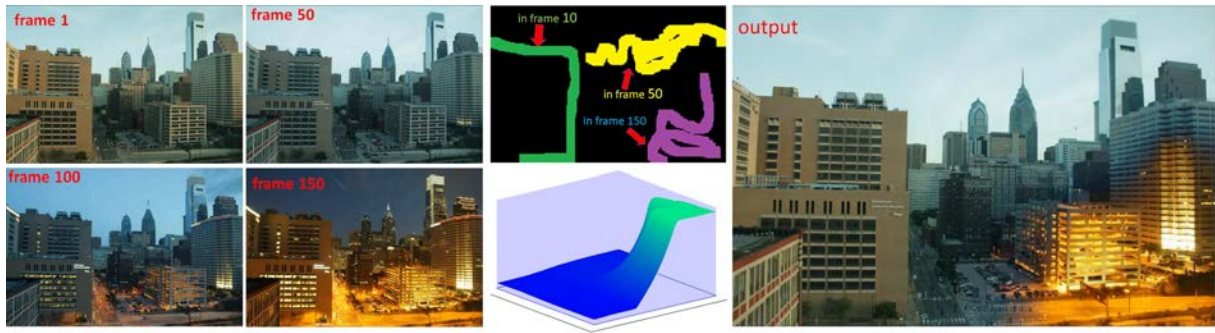
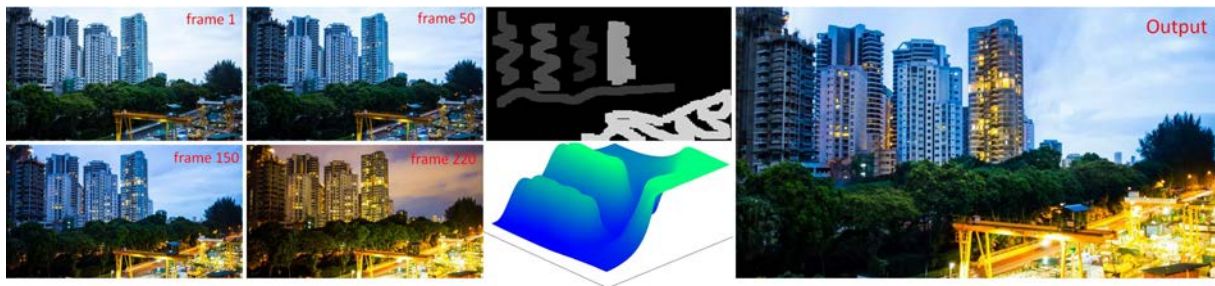**Fig. 6** Composition result for the *Philadelphia* sequence.

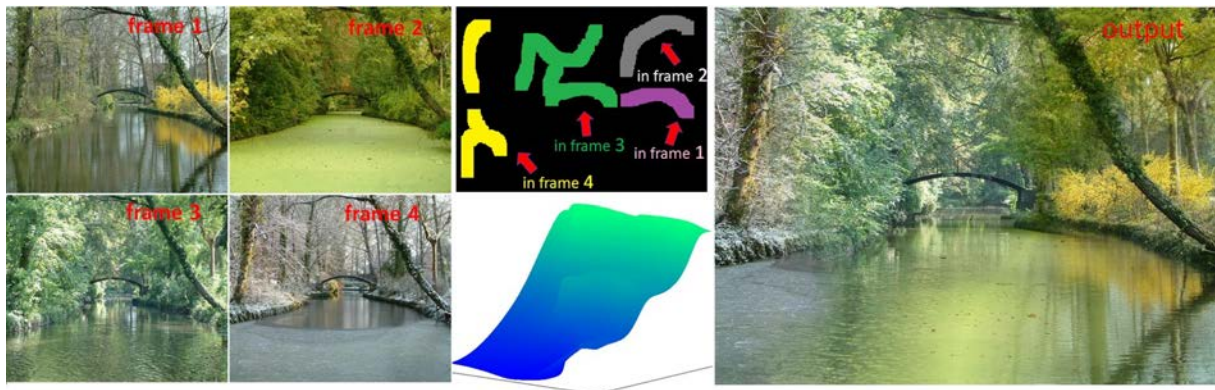**Fig. 7** Composition result for the *Singapore* sequence.

Fig. 8 Composition result for the *Jungle* sequence.

Finally, more results are shown in Fig. 9 for the *Jungle* and *Philadelphia* sequences. Again, with our color retargeting approach, the user just needs to draw several strokes and the proposed system can easily produce different composition results for the same scene.

5.3 Limitations

Our color retargeting approach has several drawbacks. Firstly, as mentioned earlier, our solution is based on the assumption that the color evolution in the input time-lapse videos is consistent with the temporal change. In other words, our color resampling approach is to represent the temporal variance by colors chosen from the corresponding frames. Thus, our solution cannot handle the color inconsistency cases with respect to the time direction. Also, if in the original video there are too many motion objects or the background is changed too frequently, it would be difficult for our approach to generate a satisfying composition result. Moreover, our system allows the user flexibility in specifying the color of objects or local transition directions in any input video frame. However, when the user's strokes are unreasonably placed or even conflicting, the proposed algorithm may fail to perform continuous resampling. This can introduce visual artifacts in the composed image. In Fig. 10 for example, one can see the color
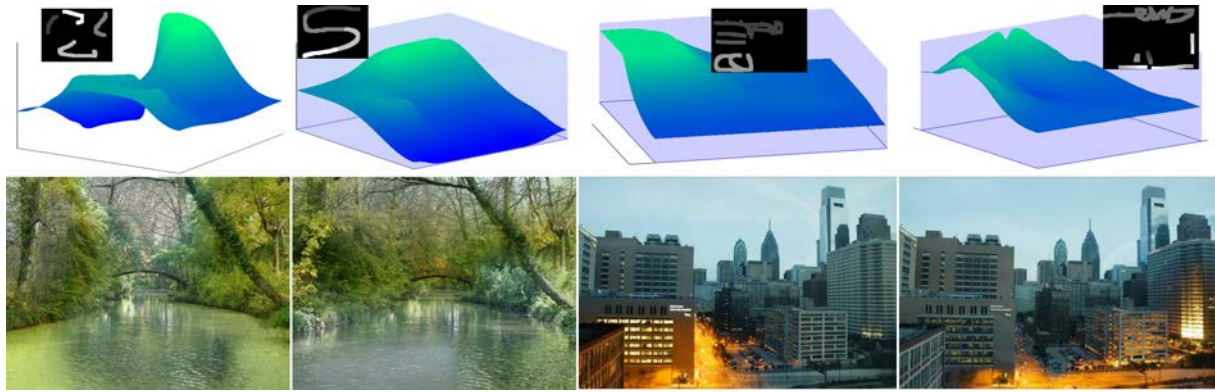
Fig. 9 Different composition results.

distortion in the sky area (close to the lamp). In this case, based on the user's strokes, the resampling is performed in frames whose indices are not consecutive. Currently our Gaussian-weighting resampling is based on the target frame of the interpolation field and its neighboring temporal frames from the input video. More complicated adjustments, such as incorporating the neighboring resampling frames on the surface of the optimized interpolation field, may improve the composition effect.

6 Conclusion and future work

In this paper we propose an interactive color retargeting approach to efficiently compose the time-varying color transition from time-lapse videos. We formulate the color composition as a pixel-level resampling problem instead of performing image matting or blending techniques. By constructing a globally optimized interpolation field, the resampling solution not only matches the user's editing requirements of preserving specified colors and satisfying local color transitions, but also effectively avoids visual artifacts in the composed image. Examples demonstrate that our efficient solution enables the user to easily edit various time-varying color image composition results. In the future we would like to extend our solution to motion-objects oriented time-varying video composition and other related applications.
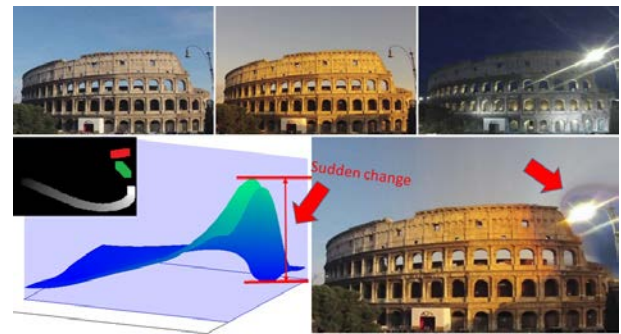
Acknowledgements

Fig. 10 A fail case. If the strokes drawn by the user are unreasonably placed or even conflicting, our approach may be difficult to construct a smooth interpolation field, and it would result in visual artifacts.

References

[1] Eigen: a c++ linear algebra library, 2015. Available online at http://eigen.tuxfamily.org/index.php?title=Main_Page.

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM Trans. Graph. (SIGGRAPH)*, 23:294–302, 2004.

[3] X. An and F. Pellacini. Appprop: all-pairs appearance-space edit propagation. *ACM Trans. Graph. (SIGGRAPH)*, 27(40):1–10, 2008.

[4] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Trans.*

Graph. (SIGGRAPH), 26(3):1–10, 2007.

[5] E. P. Bennett and L. McMillan. Computational time-lapse video. *ACM Trans. Graph. (SIGGRAPH)*, 26:102:1–6, 2007.

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Machine Intell.*, 23(11):1222–1239, 2001.

[7] B. Ceulemans, S.-P. Lu, P. Schelkens, and A. Munteanu. Globally optimized multiview video color correction using dense spatio-temporal matching. In *Proc. 3DTV-Conference (3DTV-CON)*, pages 1–4. IEEE, 2015.

[8] X. Chen, D. Zou, Q. Zhao, and P. Tan. Manifold preserving edit propagation. *ACM Trans. Graph. (SIGGRAPH ASIA)*, 31(6):1–7, 2012.

[9] C. D. Correa and K.-L. Ma. Dynamic video narratives. *ACM Trans. Graph. (SIGGRAPH)*, 29(4):88:1–9, 2010.

[10] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen. Image melding: combining inconsistent images using patch-based synthesis. *ACM Trans. Graph. (SIGGRAPH)*, 31(4):82:1–10, 2012.

[11] F. J. Estrada. Time-lapse image fusion. In *Proc. Europ. Conf. Comput. Vision (ECCV)*, pages 441–450. Springer, 2012.

[12] Z. Farbman, R. Fattal, and D. Lischinski. Diffusion maps for edge-aware image editing. *ACM Trans. Graph. (SIGGRAPH)*, 29(145):1–10, 2010.

[13] Z. Farbman, G. Hoffer, Y. Lipman, D. Cohen-Or, and D. Lischinski. Coordinates for instant image cloning. *ACM Trans. Graph. (SIGGRAPH)*, 28:67:1–9, 2009.

[14] R. Fattal, D. Lischinski, and M. Werman. Gradient domain high dynamic range compression. *ACM Trans. Graph. (SIGGRAPH)*, 21:249–256, 2002.

[15] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Optimizing color consistency in photo collections. *ACM Trans. Graph. (SIGGRAPH)*, 32(4):85:1 – 85:9, 2013.

[16] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM Trans. Graph. (SIGGRAPH)*, 23(3):689–694, 2004.

[17] S.-S. Lin, I.-C. Yeh, C.-H. Lin, and T.-Y. Lee. Patch-based image warping for content-aware retargeting. *IEEE Trans. Multimedia*, 15(2):359–368, 2013.

[18] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski. Interactive local adjustment of tonal values. *ACM Trans. Graph. (SIGGRAPH)*, 25(3):646–653, 2006.

[19] S.-P. Lu, B. Ceulemans, A. Munteanu, and P. Schelkens. Spatio-temporally consistent color and structure optimization for multiview video color correction. *IEEE Trans. Multimedia*, 17(5):577–590, 2015.

[20] S.-P. Lu, S.-H. Zhang, J. Wei, S.-M. Hu, and R. R. Martin. Timeline editing of objects in video. *IEEE Trans. Visual. Comput. Graphics*, 19(7):1218–1227, 2013.

[21] R. Martin-Brualla, D. Gallup, and S. M. Seitz. Time-lapse mining from internet photos. *ACM Trans. Graph. (SIGGRAPH)*, 34(4), 2015.

[22] P. Perez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Trans. Graph. (SIGGRAPH)*, 22:313–318, 2003.

[23] Y. Pritch, E. Kav-Venaki, and S. Peleg. Shift-map image editing. In *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, pages 151–158, Sept. 2009.

[24] A. Rav-Acha, Y. Pritch, D. Lischinski, and S. Peleg. Dynamosaicing: Mosaicing of dynamic scenes. *IEEE Trans. Pattern Anal. Machine Intell.*, 29(10):1789–1801, 2007.

[25] M. Rubinstein, A. Shamir, and S. Avidan. Improved seam carving for video retargeting. *ACM Trans. Graph. (SIGGRAPH)*, 27:16:1–16:9, 2008.

[26] V. Setlur, S. Takagi, R. Raskar, M. Gleicher, and B. Gooch. Automatic image retargeting. In *Proc. MUM*, pages 59–68. ACM, 2005.

[27] Y. Shih, S. Paris, F. Durand, and W. T. Freeman. Data-driven hallucination of different times of day from a single outdoor photo. *ACM Trans. Graph. (SIGGRAPH)*, 32(6):200:1–11, 2013.

[28] K. Sunkavalli, W. Matusik, H. Pfister, and S. Rusinkiewicz. Factored time-lapse video. *ACM Trans. Graph. (SIGGRAPH)*, 26(3):101:1–12, 2007.

[29] J. C. Tan, M. Dvoroznak, D. Sykora, and Y. Gingold. Decomposing time-lapse paintings into layers. *ACM Trans. Graph. (SIGGRAPH)*, 34(4), 2015.

[30] S. Wilkes. Day to night, 2015. Available online at http://www.stephenwilkes.com/fine-art.

[31] B.-Y. Wong, K.-T. Shih, C.-K. Liang, and H. H. Chen. Single image realism assessment and recoloring by color compatibility. *IEEE Trans. Multimedia*, 14(3):760–769, 2012.

[32] K. Xu, Y. Li, T. Ju, S.-M. Hu, and T.-Q. Liu. Efficient affinity-based edit propagation using kd tree. *ACM Trans. Graph. (SIGGRAPH ASIA)*, 28(5):118:1–6, 2009.

[33] L. Xu, Q. Yan, and J. Jia. A sparse control model for image and video editing. *ACM Trans. Graph. (SIGGRAPH ASIA)*, 32(6):1–10, 2013.

[34] Z. Zhu, R. R. Martin, and S.-M. Hu. Panorama completion for street views. *Computational Visual Media*, 1(1):49–57, 2015.
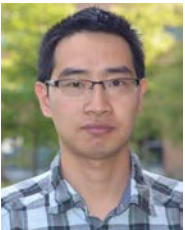
Shao-Ping Lu is currently a senior researcher at Vrije Universiteit Brussels (VUB), Brussels, Belgium, where he was a post-doc researcher from Nov. 2012 to Dec. 2014. He received the Ph.D. degree in Computer Science from Tsinghua University, Beijing, China, in July 2012.

His research interests include image/video generation, editing, representation and compression.

Guillaume Dauphin received the M.Sc. degree in Electrical Engineering from Université Libre de Bruxelles (ULB, French wing of the Brussels University) in 2015. He is now a Ph.D. student at Université Libre de Bruxelles.

Gauthier Lafruit received his Master of Engineering degree in Electronics and his Ph.D. from the Vrije Universiteit Brussel (Dutch wing of the Brussels University) in 1989 and 1995 respectively. He was research scientist with the Belgian National Foundation for Scientific Research from 1989 to 1994, being mainly active in the area of wavelet image compression implementations. In 1996, he became the recipient of the Scientific Barco award and joined Imec, Leuven, Belgium, where he was involved as Senior Scientist with the design of low-power VLSI for combined JPEG/Wavelet compression engines. From 1999 to 2013, Gauthier Lafruit held the position of Principal Scientist in Multimedia and Vision Systems, Imec. He was then promoted to Professor Visual Computing at Hasselt University before joining l'Université Libre de Bruxelles (French wing of the Brussels University) in 2014, where he is mainly active in the fields of immersive media, as well as virtual and augmented reality. He is currently co-chair of the FTV (Free viewpoint TV) working group in the international MPEG standardization committee, and actively contributes to the MPEG-ARAF and JPEG-PLENO standardization activities. Prof. Lafruit has published around seventy MPEG/JPEG standardization working documents over the past 15 years, as well as a couple of hundreds of papers in the domain of video and 3D graphics processing.

Adrian Munteanu (M'07) received the M.Sc. degree in electronics from Politehnica University of Bucharest, Romania, in 1994, the M.Sc. degree in biomedical engineering from University of Patras, Greece, in 1996, and the Doctorate degree in applied sciences (awarded with highest honours) from Vrije Universiteit Brussel (VUB), Belgium, in 2003. In the period 2004-2010 he was post-doctoral fellow with the Fund for Scientific Research - Flanders (FWO), Belgium, and since 2007, he is professor at VUB. His research interests include image, video and 3D graphics coding, distributed source coding, error-resilient coding, multimedia transmission over networks, and statistical modeling. A. Munteanu is the author of more than 200 journal and conference publications, book chapters, patent applications and contributions to standards. He is the recipient of the 2004 BARCO-FWO prize for his PhD work. A. Munteanu currently serves as Associate Editor for IEEE Transactions on Multimedia.