Eurographics Workshop on Intelligent Cinematography and Editing (2015)
W. Bares, M. Christie, and R. Ronfard (Editors)

Key-frame Based Spatiotemporal Scribble Propagation

Pelin Doğan[1], Tunç Ozan Aydın[2], Nikolce Stefanoski[2], Aljoscha Smolic[2]

[1]ETH Zürich, [2]Disney Research Zürich

Abstract

We present a practical, key-frame based scribble propagation framework. Our method builds upon recent advances in spatiotemporal filtering by adding key-components required for achieving seamless temporal propagation. To that end, we propose a temporal propagation scheme for eliminating holes in regions where no motion path reaches reliably. Additionally, to facilitate the practical use of our technique we formulate a pair of image edge metrics influenced from the body of work on edge-aware filtering, and introduce the "hybrid scribble propagation" concept where each scribble's propagation can be controlled by user defined edge stopping criteria. Our method improves the current state-of-the-art in the quality of propagation results and in terms of memory complexity. Importantly, our method operates on a limited, user defined temporal window and therefore has a constant memory complexity (instead of linear) and thus scales to arbitrary length videos. The quality of our propagation results is demonstrated for various video processing applications such as mixed HDR video tone mapping, artificial depth of field for video and local video recoloring.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques, I.3.8 [Computer Graphics]: Applications—.

1. Introduction

Region based application of image and video edits is one of the most common post-processing tasks in practice. However, the manual creation of masks by identifying regions of interest (rotoscoping) is very labor intensive, especially for videos. This makes semi-automatic methods highly desirable for practical use. Such semi-automatic workflows often start with the user roughly marking a region of interest through some user interface element (e.g. a scribble) on a number of *key-frames*, followed by the system automatically propagating the scribbles (both spatially and temporally) to the pixels of the marked region throughout the video sequence. Depending on the application needs, the masks generated through a semi-automatic workflow can then either be used directly instead of manually generated masks, or can be used as a starting point for further refinements while still significantly reducing the total amount of labor.

Seamless propagation of user defined scribbles involves a multitude of technical challenges. The spatial propagation requires the formulation of one or more edge-stopping conditions such that the propagation can be stopped at the region boundaries intended (but not manually marked) by the user. Moreover, these region boundaries need to be retained as much as possible during the temporal propagation from key-frames to *inter-frames* that contain no user input whatsoever. Another fundamental and significant challenge is achieving high quality temporal propagation using the often error-prone optical flow estimates. Finally, in order to be used with arbitrary length videos at high definition and beyond, the formulation of the spatiotemporal propagation should be parallelizeable and have constant memory complexity. In that sense, the utility of techniques that require the entire video cube to be loaded into memory is limited.

In this work we investigate the practical challenges with the key-frame based spatiotemporal propagation of user scribbles and make the following conributions:

- A spatiotemporal scribble propagation framework with constant memory complexity that improves over the state-of-the-art propagation quality.
- The *hybrid scribble propagation* approach, where multiple scribbles' propagation can be controlled by user-defined edge stopping criteria and later combined into a single map.
- A temporal propagation scheme that effectively avoids "holes" at the inter-frames where no scribble colors from the key-frames could reliably be propagated.

DOI: 10.2312/wiced.20151073

The implementation of our method builds upon the recent work on spatiotemporal filtering [ASC*14]. The memory footprint of our technique is constant as for each processed frame we only keep a user defined number of future and past frames in the memory. Thus our technique scales to arbitrary length videos, unlike methods that load the entire video into the memory such as Levin et al. [LLW04] or Lang et al. [LWA*12]. We demonstrate the use of our method for applications such as mixed HDR video tone mapping, artificial depth of field for video and local video recoloring.

2. Background

In this section we review the previous work in related research areas and give background information to facilitate the discussion presented in later sections.

Scribble Propagation methods aim at propagating the user input delivered in the form of sparse scribbles, such that the resulting propagation covers entire regions and objects. Early work by Levin et al. [LLW04] on re-coloring formulizes a quadratic cost function which can be solved using standard optimization techniques. Scribble propagation by minimizing a cost function has been further investigated in many follow-up works, Grady [Gra06] and Levin et al. [LLW08] are some examples. In the latter work, the cost function is modified to better suit the problem of image matting. Another scribble based interface for interactive matting was proposed by Wang et al. [WC05] which employs an expensive iterative nonlinear optimization process. Yatziv et al. [YS06] proposes a color blending scheme which was followed up by Bai et al. [BS07] that proposes a technique based on the optimal, linear time computation of weighted geodesic distances to the user-provided scribbles. Adelson et al. [LAA08] performs edge aware interpolation by adding a classification step for the scribble pixels. Chen et al. [CZZT12] proposed a manifold preserving edit propagation method which was followed by a more efficient formulation by Ma et al. [Ma114]. In other works, scribbles have been propagated with probabilistic distance transform [LS08], using cellular automation [KV06] and by random walks [KLL09].

Optical Flow based Temporal Propagation methods utilize optical flow for estimating correspondences among consecutive video frames. The accurate estimation of optical flow is one of the most well studied and fundamental problems in video processing. Some of the modern methods [ZBW11, SRB10] still use a variation of the original formulation [HS81], whereas others such as Rhemann et al. [RHB*11] are based on the construction of a cost-volume. Optical flow has been utilized in temporal video filtering for enabling various editing tasks. Lang et al. [LWA*12] proposed a framework that uses edge-aware filtering and propagation in the temporal domain for various video processing applications. Their temporal filtering and propagation utilize the estimated motion path for each pixel throughout a video volume. Since the entire video volume needs to be kept in memory, this method does not scale well to long video sequences at high resolutions. More recently, Aydin et al. [ASC*14] proposed a temporal filtering framework for high dynamic range video, that instead operates on a limited temporal window and was shown to scale well to long video sequences at high definition and beyond.

Edge-Aware Filtering has been an active research area which is directly related to scribble propagation, since detecting visually important edges is a key challenge for both problems. Also, Lang et al. [LWA*12] showed that an edge-aware filtering method can be trivially modified to perform propagation through a simple normalization step, which we utilize and extend in this work. Early work in edge-aware filtering by Perona and Malik [PM90] introduced anisotropic diffusion which was followed by the bilateral filter [TM98] as an alternative. In Black et al. [BSMH98], anisotropic diffusion is modified in the framework of robust statistics which then inspired a different analysis of bilateral filtering [DD02]. The trilateral filter [CT05] is built from the modified forms of the bilateral filter [TM98], which smooths the image towards piecewise constant solutions. Farbman et al. [FFLS08] presented an alternative edge-preserving operator that is based on a weighted least squares framework [LBB88]. At the cost of long computational times due to sparse linear system solutions, this edge-preserving operator provides more robust and versatile results in the applications that have been so far based on the bilateral filter. A more efficient filtering framework based on edge avoiding wavelets has been presented by Fattal [Fat09]. Criminisi et al. [CSRP10] proposed a generalized geodesic transform for general edge-aware image and video editing. Gastal and Oliveira [GO11] presented a new approach for edge preserving filtering of images and videos based on a novel *domain transform*. More recent work on edge-aware filtering focused on finding more sophisticated metrics for localizing edges and estimating their visual significance. Karacan et al. [KEE13] utilizes *region covariances* for distinguishing texture from image edges, however their method is slow due to the increased computational complexity.

3. Scribble Propagation Framework

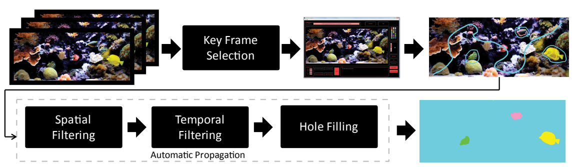

Figure 1: *The data flow diagram of the framework. User input is required for key frame selection and scribble drawing, whereas the remaining spatiotemporal propagation steps are performed automatically by our system.*

The main processing steps of our framework are illustrated in Figure 1. Given an input video, we require the user to select a number of key-frames and to manually draw scribbles on those key-frames. The resulting scribbles, as well as the input video is then processed by our automatic propagation method, which yields the resulting spatiotemporally propagated scribble colors. In order to facilitate the practical use of our method, we created a GUI where the user can interactively load key-frames of a video, draw scribbles and conveniently run our propagation method (see the supplementary material for the illustration). In the reminder of this section we discuss the main building blocks of our automatic propagation method.

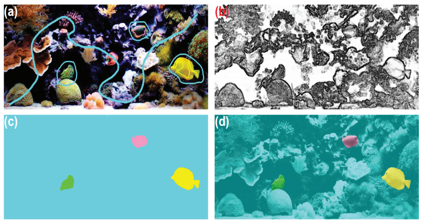

Figure 2: *Our method propagates the input scribbles (a) using the permeability map of the current frame (b). The scribble colors are propagated without generating holes (c) while also respecting the image edges (d).*

3.1. Spatial Propagation

Our method uses the permeability-guided filtering (PGF) technique of Aydin et al. [ASC*14] as a starting point. The PGF approach involves iterative application of a 1-dimensional filter along image rows and image columns in a separable way (see supplementary material).

We consider a *scribble image* as an image with two colors, black and scribble color. Associated with a scribble image is always a binary image, which has ones at the scribble pixels and zero at the remaining pixels. To propagate a scribble image S based on a natural image I by the PGF approach, first permeability values are computed from I. Permeabilities computed from natural images are mostly smaller than 1. Consequently, a naive application of the PGF method on S based on these permeability values leads to a decrease of the diffused scribble color value as the scribble color is propagated to other pixels. To prevent the fading of propagated scribble colors, we employ a normalization term similar to Lang et al. [LWA*12] (See the supplementary material to visualize the effect of the normalization term). It is employed as follows in the filtering equation

$$S_p^{(k+1)} = \begin{cases} \frac{\sum_{q=1}^{N} \pi_{pq} S_q^{(k)}}{\sum_{q=1}^{N} \pi_{pq} B_q^{(k)}} & : \quad \sum_{q=1}^{N} \pi_{pq} B_q^{(k)} > \varepsilon \\ 0 & : \quad \text{else} \end{cases} , \quad (1)$$

where k indicates the iteration, $S_p^{(k)}$ is the scribble intensity value of the pixel p, π_{pq} is the permeability coefficient in the PGF technique in [ASC*14], and ε is a threshold which controls the sensitivity of the propagation to edges. $B_q^{(k)}$ is the binary value of the pixel q on the scribble image and $B_q^{(k+1)}$ is calculated by treating the values of $B_q^{(0)}$ as decimal with the range $[0,1]$, so that the values of $B_q^{(k+1)}$ are decimal.

By applying the aforementioned normalization scheme to the PGF approach, we convert the filtering method to a scribble propagation method, while also retaining the edge-awareness property of the original formulation given by the use of permeability weights. Figure 2 shows a typical result from our permeability based propagation framework, where the permeability weights are computed by utilizing color gradients. The limitations of the color gradient based edge metric have been discussed in several works in the field of edge-aware filtering (e.g. [SSD09, KEE13] among others). Their findings revealed that such a simple metric fails to deal with edges in strongly textured regions. This limitation also carries over to our propagation framework and has a negative impact on the overall accuracy of its results. In the next section we address this limitation by revisiting the color gradient based edge metric.

3.2. Hybrid Scribble Propagation

Our experiments with various edge metrics revealed that no single metric works universally. This observation led us to revisit the standard scribble that is only associated with a unique color value and propose an extension where the scribble is also associated with an edge metric that will be used for the scribble's propagation. In our implementation, the association of a scribble with a particular edge metric is given as user input similar to the scribble color. Equipped with this information, our method performs *hybrid scribble propagation* where the propagation of each scribble is controlled by its designated edge metric. In the reminder of this section we introduce two new edge metrics that can be utilized for hybrid propagation, namely *local entropy* and *local L-moments*.

In image processing, we can associate the term entropy, from information theory, with the randomness of the color values. The neighboring pixels whose neighborhood are in the same texture should result in similar entropy values. Thus we can use the difference between the local entropies of the neighboring pixels as our permeability measure. Our method's color permeability measure can be modified to use entropy as follows:

$$\tilde{\pi} = \left(1 + \left|\frac{E_{local,p} - E_{local,p'}}{\sigma}\right|^{\alpha}\right)^{-1} \quad (2)$$

where $E_{local,p}$ and $E_{local,p'}$ denote the local entropy of neighboring pixels p and p'.

Figure 3-middle column shows two examples where the

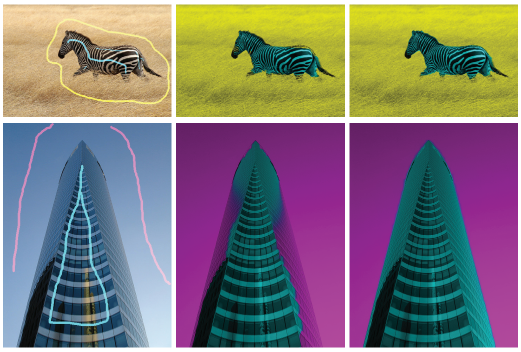

Figure 3: *Examples of entropy scribbles on the foreground mixed with color gradient scribbles on the background. Note that the use of color gradient scribbles uniformly over whole images result in small errors in the zebra image (top center) and significant leaks for the skyscraper image (bottom center). The hybrid entropy-color gradient scribbles result in improved results in both cases (right column).*

traditional scribble propagation using color gradients fails. In the *zebra* image the background scribbles slightly leak into the zebra's head and back, where in the *skyscraper* image, the leak from the background is more pronunced. In both cases we can improve the propagation result by applying a hybrid propagation where we the utilize entropy metric in the foreground and the color gradient metric in the background (Figure 3-right column) using the same set of scribbles (Figure 3-left column).

To enable the hybrid propagation, we compute the non-normalized scribble colors and the binary normalization masks (S and B from Equation 1) separately for both the color gradient scribbles and the entropy scribbles. We can formally express the hybrid propagation by extending Equation 1 as follows:

$$S_p^{(k+1)} = \begin{cases} \frac{\sum_{m\in M}\left[\sum_{q\in\Omega}\pi_{pq}^{m}S_q^{(k)}\right]_m}{Q} & : \quad Q > \varepsilon \\ 0 & : \quad \text{else} \end{cases}, \tag{3}$$

$$Q := \sum_{m\in M}\left[\sum_{q\in\Omega}\pi_{pq}^{m}B_q^{(k)}\right]_m, \tag{4}$$

where M is the set of edge metrics (in this case $M = \{\text{ColorGradient}, \text{Entropy}\}$), and π_{pq}^{m} is a permeability computed with edge metrics m.

In addition to the entropy metric, we also investigated the use of *local L-moments* for hybrid propagation. In our implementation we consider the first four L-moments and compute them on a 5×5 neighborhood around each pixel. The L-moment metric can be integrated into our method by modifying the permeability formulation as follows:

$$\tilde{\pi} = \left(1 + \left|\frac{\sqrt{\sum_{i=1}^{4}(L_{i_p} - L_{i_{p'}})^2}}{\sigma}\right|^{\alpha}\right)^{-1} \tag{5}$$

where L_{i_p} is the i^{th} L-moment for the neighborhood of pixel p and $L_{i_{p'}}$ is the i^{th} L-moment for the neighborhood of pixel p' which is the right neighbor of pixel p.

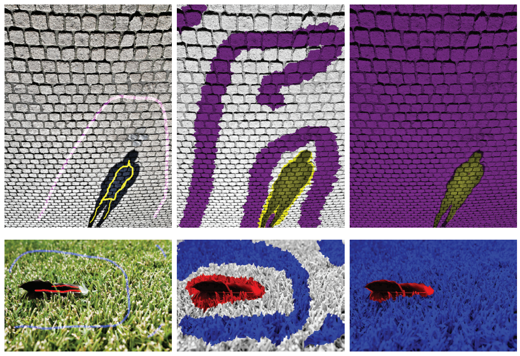

Figure 4: *Examples of L-moment scribbles on the background mixed with color gradient scribbles on the foreground. Note that in both examples, the hybrid scribbles (left column) yield significantly better results compared to uniformly used color gradient scribbles (middle column).*

As with the entropy metric, we can perform hybrid propagation using local L-moments as described in Equation 3 by defining M accordingly. Such a hybrid propagation result that involves the L-moment metric for the background and the color gradient metric for the foreground is shown in Figure 4. Notice how the L-moment metric improves the propagation of the background scribbles (right column) compared to the color gradient metric (center column) using the same set of scribbles (left column).

3.3. Temporal Propagation

Once the spatial propagation scheme from Section 3.1 is applied to the user selected key-frames, the next step in our pipeline (Figure 1) is the automatic temporal propagation of the dense scribble colors from the key-frames to the *inter*-frames. To that end, we build upon the temporal filtering from [ASC*14] where they simply utilize the same *permeability* concepts (that were previously utilized in the spatial dimension) in the temporal dimension. Formally, this can be expressed by defining pixels p and p' as *temporal neighbors*, instead of spatial neighbors. We utilize the resulting temporal filtering framework for propagating scribble colors to the inter-frames applying the normalization procedure described in Equation 1 along a single dimension on the temporal axis.

While in temporal filtering, the temporal neighborhood can simply be defined as the corresponding pixels between consecutive frames, in the case of key-frame based propagation the temporal neighborhood of a pixel in an inter-frame is instead defined by the corresponding pixels in the nearest two key-frames. Therefore, in our application we follow motion paths to find the corresponding pixels in the nearest key-frames, and propagate the scribble colors as follows:

$$S_{t,p} = \begin{cases} \dfrac{\pi_{pst}\, S_{pst-key,p} + \pi_{ftr}\, S_{ftr-key,p}}{\pi_{pst} + \pi_{ftr}} & : \pi_{pst} + \pi_{ftr} \neq 0 \\ 0 & : \text{otherwise} \end{cases} \tag{6}$$

where $S_{t,p}$ is the scribble value for the pixel p in frame t, $S_{pst-key,p}$ is the scribble value for the corresponding pixel in the past key frame, $S_{ftr-key,p}$ is the scribble value for the corresponding pixel in the future key frame. The coefficients π_{past} and π_{future} are temporal permeabilities computed using the permeability formulation in [ASC*14] in the temporal dimension.

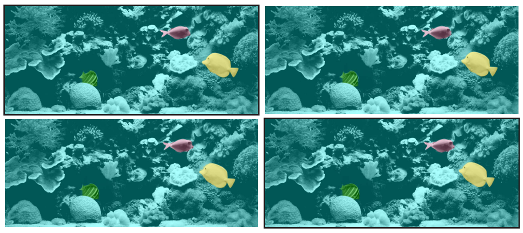

Figure 5: *A successful temporal propagation example. Black borders indicate the key-frames.*

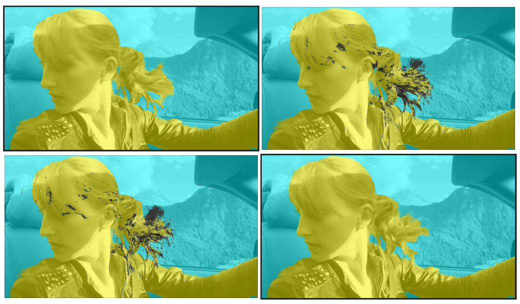

Figure 6: *An example where temporal propagation fails due to incorrect optical flow. Black borders indicate the key-frames. Notice the regions of the inter-frames where no scribble color could be propagated (indicated with black).*

Since so far our method relies solely on the accuracy of the motion paths for temporal propagation, propagation errors such as holes in the inter-frames are inevitable. These holes appear when no reliable path exists between a pixel in an inter-frame and the corresponding pixels in the nearest key-frames. Using a sophisticated optical flow estimate (we utilize Zimmer et al. [ZBW11]) can alleviate this problem to a certain degree. An example where relying on optical flow for propagation yields satisfactory results is presented in Figure 5. However, our experience suggests that such cases are rare and additional means of dealing with propagation errors are necessary for facilitating the use of our method in practice. An example where the optical flow inaccuracy affects the propagation results is shown in Figure 6.

For removing the holes in the inter-frames pixels (where no motion path from either key-frames reaches) we propose a simple hole filling scheme that is applied after the initial propagation. Since we know that the optical flow estimate is inaccurate at such holes, instead of relying to the optical flow estimate we simply assume that these pixels have straight temporal paths. In other words, we are assuming that these pixels' colors do not change over time, and therefore the corresponding pixels in the consecutive frames are located in the same spatial location. Accordingly for each frame we update the scribble colors at the holes relying on a photo-constancy measure:

$$S_{t,p} = \begin{cases} S_{t+1,p} & : \Delta I_{p_{t,t+1}} < \Delta I_{p_{t,t-1}} \quad \text{and} \quad \Delta I_{p_{t,t+1}} < c \\ S_{t-1,p} & : \Delta I_{p_{t,t+1}} > \Delta I_{p_{t,t-1}} \quad \text{and} \quad \Delta I_{p_{t,t-1}} < c \\ 0 & : \textit{otherwise} \end{cases} \tag{7}$$

where $\Delta I_{p_{m,n}} = |I_{p,m} - I_{p,n}|$ is the color difference between the pixel p of frame n and m, and c is the photo-constancy threshold.

Obviously the assumption that the pixel colors remain static over time is often violated. However, as the example in Figure 7-a shows that by relying to the spatial color similarity of objects, we can transfer a notable amount of scribble colors from consecutive frames (Figure 7-b) using Equation 7.

While the photo-constancy based scribble color transfer method from consecutive frames is effective, it does still leave holes at the pixels where the photo-constancy assumption is violated. Therefore, after the temporal color transfer, we perform an additional edge-aware spatial propagation step (as described in Section 3.1) for performing a similar color transfer from the spatial neighbors. This final step removes any remaining holes in our final propagation results (Figure 7-c).

To summarize, our final temporal propagation method starts initially by performing an optical flow based propagation of scribble colors from key-frames to the inter-frames. Then for each frame starting from the first, we first apply the temporal color transfer relying on photo-constancy, and next apply a spatial color transfer using edge-aware propagation. After the completion of both color transfer steps, we move to the next frame until completion.

Figure 7: *Final propagation result. The initial propagation's (a) holes due to incorrect optical flow are partially filled by propagating through straight motion paths (b). Next, any remaining holes are filled with a final spatial propagation pass (c).*

4. Results

In this section we present example results of our method, compare these results with the state-of-the-art and provide insights on the effect of optical flow to the quality of propagation results

The results of a highly challenging sequence are presented in Figure 8 where the actors move fast and unpredictably. A comparison of our method with Lang et al. [LWA*12] (using the same scribbles every 10^{th} frame) shows leaked scribble colors in their results. Our method, while not perfect (notice the right leg of the magenta colored actor), yields a notably better result. A comparison of our method with and without utilizing optical flow can be also seen. The version of our method that does not use optical flow simply assumes that the scene is static (i.e. all motion paths are straight in time) and relies on the hole filling step from Section 3.3. As such, its results are similar to our full method that uses optical flow except the regions with rapid motion. In such regions the propagation fails since there is very little overlap between the pixels on an object at consecutive frames.

Figure 8: *A comparison of our result with Lang et al. [2012]. Note the leakage near the boundaries of the performers.*

5. Applications

Using our scribble propagation framework one can generate dense masks from a small number of scribbles with little manual effort. Even though the quality of these masks is not at the same level with hand drawn rotomasks, we found that their precision is sufficient for achieving high quality result for numerous video editing tasks. In this section we demonstrate three such examples Note that, our method can also be easily applied to other applications such as meta-data propagation where small deviations form object boundaries can be tolerated.

5.1. Mixed HDR Video Tone Mapping

Multiple new HDR video tone mapping methods have been proposed recently. One aspect of HDR video tone mapping that has been missing so far is the ability of applying different tone mapping operators (TMOs) to different regions of a video. This is a very relevant problem in practice: one commonly faced problem is that the local TMOs, which enhance small scale details, work nicely for background regions (Figure 9-a). But when applied to human skin they are known to generate a "dirty face" effect. Thus, for such regions the more natural look provided by global TMOs is preferred (Figure 9-b), which in turn does not work as good for the background regions. Utilizing our method (Figure 9-c), one can run both types of TMOs, and blend the results of both TMOs to combine locally tone mapped backgrounds with natural looking skin regions (Figure 9-d).

5.2. Artificial Depth of Field for Video

A significant challenge of artificial depth of field is the estimation of the scene depth which is often an error prone process. Instead, our method provides an easier alternative where one can draw focus layers conveniently using scribbles at the key frames of a video. These focus scribbles are then spatiotemporally propagated to the whole video and can be used by a generic lens blur algorithm (Figure 10-right). Note that our method allows changing focus from an object to another during the course of the video as demonstrated in Figure 10 where the focus switches from the blue car to the yellow car.

5.3. Local Video Color Editing

Local color editing has been a classical application showcased by prior scribble propagation methods. As the final application of our method, we demonstrate how the spatiotemporally propagated scribbles can be used to edit the color of selected objects in a video sequence (Figure 11).

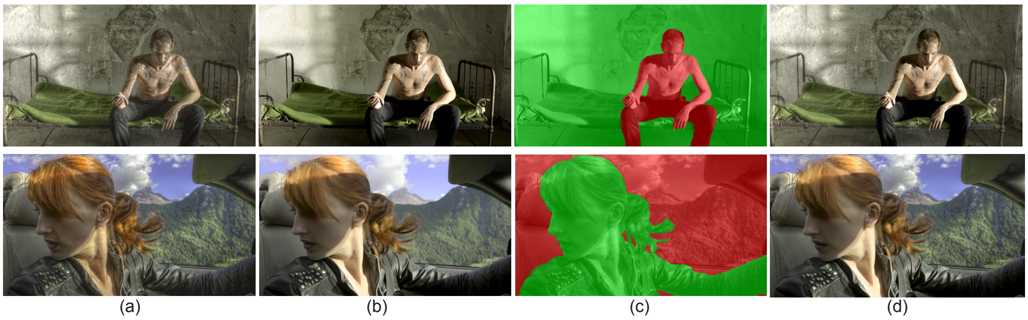

Figure 9: *Examples of mixed HDR video tone mapping (d). The result of a local TMO [MMS06] (a) is mixed with the result of a global TMO [RSSF02] (b) using the map generated using our method (c).*

Figure 10: *Artificial depth of field example. Given an input video (left column), our method can be used to generate maps that indicate camera focus, which in turn can be used to create an artificial depth of field effect (right column).*

Figure 11: *Local color editing example. The maps generated efficiently by our method can also be used for editing the colors of selected objects.*

6. Discussion

We built a prototype of our method in Matlab, where the users can draw scribbles and view the propagation results conveniently through a GUI. In our research prototype the computation of a spatial propagation step takes on average 1.03 seconds for a color video frame at HD resolution on a standard PC. Note that the runtime of our method is agnostic to the number of input scribbles. The temporal propagation for the same video frame takes on average 5.29 seconds.[†]. Our method operates on a limited temporal window and therefore is highly parallelizeable, and we believe an implementation that exploits this parallelism can significantly lower our current run-times. Also, a notable advantage of our method is its constant memory complexity due to the window based formulation.

† All reported run-times also include time spent at I/O.

Our method is not without limitations. Although in practice the proposed temporal and spatial scribble color transfer methods eliminate holes in the inter-frames accurately up to a certain size, larger holes due to low quality optical flow estimates can result in the propagation of erroneous scribble colors to the holes. In this work, we also did not discuss the use of edge metrics beyond color gradients in the temporal domain. Investigating new metrics for temporal edges would be an interesting future direction for our research.

7. Conclusion

We presented a semi-automatic spatiotemporal scribble propagation framework, where we examined the use of the recent advances in spatiotemporal filtering, identified their shortcomings for the scribble propagation problem, and proposed novel solutions to these shortcomings. We introduced the concept of *hybrid scribble propagation* to facilitate the propagation of scribble colors in textured regions, and presented a temporal propagation scheme that avoids holes through temporal and spatial color transfer from neighboring regions. Finally, through our research prototype we demonstrated the practical use of our method by presenting example results of mixed HDR video tone mapping, artificial depth of field for video andm local video color editing.

8. Acknowledgements

We thank the anonymous reviewers, Simone Croci, Steven Poulakos, Pascal Frossard, Markus Gross, Valentina Storti, Börkur Sigurbjörnsson, Martin Fisch, Nina Matthews, the MUSCADE project, Blender Foundation | www.blender.org, Flickr and ESPN. This work was partly supported by the EU FP7 project ACTION-TV (http://www.action-tv.net), funded under the European Commission's 7th Framework Program (Grant Number: 611761).

References

[ASC*14] AYDIN T. O., STEFANOSKI N., CROCI S., GROSS M., SMOLIC A.: Temporally coherent local tone mapping of hdr video. In *ACM TOG* (2014). 2, 3, 4, 5

[BS07] BAI X., SAPIRO G.: A geodesic framework for fast interactive image and video segmentation and matting. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on* (2007), IEEE, pp. 1–8. 2

[BSMH98] BLACK M. J., SAPIRO G., MARIMONT D. H., HEEGER D.: Robust anisotropic diffusion. *Image Processing, IEEE Transactions on 7*, 3 (1998), 421–432. 2

[CSRP10] CRIMINISI A., SHARP T., ROTHER C., PEREZ P.: Geodesic image and video editing. *ACM TOG 29*, 5 (2010), 134. 2

[CT05] CHOUDHURY P., TUMBLIN J.: The trilateral filter for high contrast images and meshes. In *ACM SIGGRAPH 2005 Courses* (2005), ACM, p. 5. 2

[CZZT12] CHEN X., ZOU D., ZHAO Q., TAN P.: Manifold preserving edit propagation. *ACM TOG 31*, 6 (2012), 132:1–132:7. 2

[DD02] DURAND F., DORSEY J.: Fast bilateral filtering for the display of high-dynamic-range images. In *ACM TOG* (2002), vol. 21, ACM, pp. 257–266. 2

[Fat09] FATTAL R.: Edge-avoiding wavelets and their applications. In *ACM TOG* (2009), vol. 28, p. 22. 2

[FFLS08] FARBMAN Z., FATTAL R., LISCHINSKI D., SZELISKI R.: Edge-preserving decompositions for multi-scale tone and detail manipulation. In *ACM TOG* (2008), vol. 27, p. 67. 2

[GO11] GASTAL E. S., OLIVEIRA M. M.: Domain transform for edge-aware image and video processing. In *ACM TOG* (2011), vol. 30, ACM, p. 69. 2

[Gra06] GRADY L.: Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 28*, 11 (2006), 1768–1783. 2

[HS81] HORN B. K., SCHUNCK B. G.: Determining optical flow. In *1981 Technical Symposium East* (1981), International Society for Optics and Photonics, pp. 319–331. 2

[KEE13] KARACAN L., ERDEM E., ERDEM A.: Structure-preserving image smoothing via region covariances. *ACM TOG 32*, 6 (2013), 176. 2, 3

[KLL09] KIM T. H., LEE K. M., LEE S. U.: Edge-preserving colorization and recoloring based on coupled map lattices. *IEEE International Conference on Image Processing* (2009), 1661–1664. 2

[KV06] KONUSHIN V., VEZHNEVETS V.: Interactive image colorization and recoloring based on coupled map lattices. In *Graphicon* (2006), pp. 231–234. 2

[LAA08] LI Y., ADELSON E., AGARWALA A.: Scribbleboost: Adding classification to edge-aware interpolation of local image and video adjustments. In *Computer Graphics Forum* (2008), vol. 27, Wiley Online Library, pp. 1255–1264. 2

[LBB88] LAGENDIJK R. L., BIEMOND J., BOEKEE D. E.: Regularized iterative image restoration with ringing reduction. *Acoustics, Speech and Signal Processing, IEEE Transactions on 36*, 12 (1988), 1874–1888. 2

[LLW04] LEVIN A., LISCHINSKI D., WEISS Y.: Colorization using optimization. In *ACM TOG* (2004), vol. 23, pp. 689–694. 2

[LLW08] LEVIN A., LISCHINSKI D., WEISS Y.: A closed-form solution to natural image matting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 30*, 2 (2008), 228–242. 2

[LS08] LAGODZINSKI P., SMOLKA B.: Digital image colorization based on probabilistic distance transformation. In *ELMAR* (2008), vol. 2, pp. 495–498. 2

[LWA*12] LANG M., WANG O., AYDIN T., SMOLIC A., GROSS M. H.: Practical temporal consistency for image-based graphics applications. *ACM TOG 31*, 4 (2012). 2, 3, 5

[Ma114] Efficient manifold preserving edit propagation with adaptive neighborhood size. *Computers and Graphics 38* (2014), 167 – 173. 2

[MMS06] MANTIUK R., MYSZKOWSKI K., SEIDEL H.-P.: A perceptual framework for contrast processing of high dynamic range images. *ACM TAP 3*, 3 (2006), 286–308. 7

[PM90] PERONA P., MALIK J.: Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 12*, 7 (1990), 629–639. 2

[RHB*11] RHEMANN C., HOSNI A., BLEYER M., ROTHER C., GELAUTZ M.: Fast cost-volume filtering for visual correspondence and beyond. In *IEEE CVPR* (2011), pp. 3017–3024. 2

[RSSF02] REINHARD E., STARK M., SHIRLEY P., FERWERDA J.: Photographic tone reproduction for digital images. *ACM TOG 21*, 3 (2002), 267–276. 7

[SRB10] SUN D., ROTH S., BLACK M. J.: Secrets of optical flow estimation and their principles. In *IEEE CVPR* (2010), IEEE, pp. 2432–2439. 2

[SSD09] SUBR K., SOLER C., DURAND F.: Edge-preserving multiscale image decomposition based on local extrema. *ACM TOG 28*, 5 (2009), 147. 3

[TM98] TOMASI C., MANDUCHI R.: Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on* (1998), IEEE, pp. 839–846. 2

[WC05] WANG J., COHEN M. F.: An iterative optimization approach for unified image segmentation and matting. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* (2005), vol. 2, IEEE, pp. 936–943. 2

[YS06] YATZIV L., SAPIRO G.: Fast image and video colorization using chrominance blending. *Image Processing, IEEE Transactions on 15*, 5 (2006), 1120–1129. 2

[ZBW11] ZIMMER H., BRUHN A., WEICKERT J.: Optic flow in harmony. *International Journal of Computer Vision 93*, 3 (2011), 368–388. 2, 5