Occlusion-aware Video Temporal Consistency

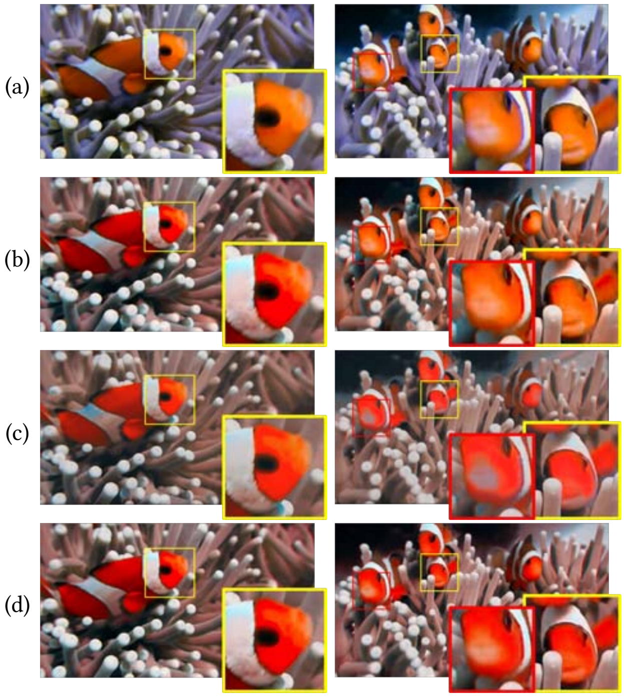
Chun-Han Yao, Chia-Yang Chang, Shao-Yi Chien
Media IC and System Lab
Graduate Institute of Electronics Engineering and Dept. of Electrical Engineering
National Taiwan University
b01901015@ntu.edu.tw,cychang@media.ee.ntu.edu.tw,sychien@ntu.edu.tw

ABSTRACT

Image color editing techniques such as color transfer, HDR tone mapping, dehazing, and white balance have been widely used and investigated in recent decades. However, naively employing them to videos frame-by-frame often leads to flickering or color inconsistency. To solve it generally, earlier methods rely on temporal filtering or warping from the previous frame, but they still fail in the cases of occlusion and produce blurry results. We introduce a new framework for these challenges: (1) We develop an online keyframe strategy to keep track of the dynamic objects, where more temporal information can be acquired than a single previous frame. (2) To preserve image details, local color affine model is employed. The main concept of this post-processing step is to capture the color transformation from editing algorithms and maintain the detail structures of the raw image simultaneously. Practically, our approach takes a raw video and its per-frame processed version, and generates a temporally consistent output. In addition, we propose a video quality metric to evaluate temporal coherence. Extensive experiments and subjective test are done to show the superiority of the proposed framework with respect to color fidelity, detail preservation, and temporal consistency.

KEYWORDS

Video processing; temporal consistency; occlusion; color transfer



Frame 30 Frame 100

Figure 1: An example set of (a) raw video, (b) per-frame processed version by color transfer [35], and the temporally consistent results of (c) Bonneel *et al.* [7] and (d) our framework. As the fish in yellow square being temporally occluded in the frames between 30 and 100, per-frame process causes color discrepancy, and the method of Bonneel *et al.* blurs the details. Our framework is more robust to occlusion and preserves the details with higher color fidelity.

1 INTRODUCTION

As photo editing applications are gaining popularity, numerous image processing techniques have been proposed. On the contrary, there are few algorithms customized for videos. Intuitively, one can employ arbitrary image processing methods on each video frame. Nevertheless, since the luminance and chrominance distribution of video frames may vary significantly, per-frame process is likely to result in artifacts such as flickering and color discrepancy, as shown in Fig. 1.

To address the problem, several existing methods explicitly encode temporal consistency into individual processing algorithm. For instance, Bonneel *et al.* [6] propose video color grading by temporally interpolating the color transfer functions. Aydin *et al.* [3] integrate a spatiotemporal filter into an HDR video tone mapping operator. Ye *et al.* [36] build a causal-anticausal iterative scheme for intrinsic decomposition. These methods are effective but cannot be generalized to other tasks. Some others propose more generic approaches for various applications. Lang *et al.* [20] apply edge-aware filters temporally, which reduces flickering but does not fully eliminate it. Moreover, when optimizing the objective function, it often trades spatial sharpness for temporal smoothness. Bonneel *et al.* [7] propose a gradient-domain technique that is blind to the particular image processing algorithm. By inferring the temporal regularity from the original unprocessed video, it takes a series of processed frames that suffers from flickering and generates a temporally consistent video sequence. While performing well in most cases, it may still lead to low-frequency color drifts between

MM'17, October 23–27, 2017, Mountain View, CA, USA.
© 2017 ACM. ISBN 978-1-4503-4906-2/17/10…$15.00
DOI: https://doi.org/10.1145/3123266.3123363

frames. Since the warping error from the previous output frame is a part of the regularization term, the problem is especially severe when occlusion occurs, as shown in Fig. 1(c).

Inspired by Bonneel *et al.* [7] and the technique of example-based video color transfer proposed by Yao *et al.* [35], we develop a general framework for video temporal consistency. In detail, we select a stack of keyframes with the highest diversity of color distribution, seeking to capture the objects that once appeared but has disappeared or been occluded for a while. To maintain temporal consistency, the PatchMatch method [4] is employed to warp an intermediate frame with the processed keyframe patches. Subsequently, we compensate the warping error of dynamic objects via Laplacian pyramid [8] decomposition and fusion, which preserves the coarse intensity of the intermediate frame and the high-frequency details of the per-frame processed frame. Finally, seeing the strong expressivity of local affine transfer in the works of Gharbi *et al.* [12] and Shih *et al.* [28], local color affine transformations are smoothed by edge-preserving filters and applied as a post-processing step for denoising and deblurring.

Noticing the lack of video quality measurements for temporal consistency, we further refine the quality metric proposed by Yao *et al.* [35], which considers the difference of warping errors between the raw and processed video frames. For performance comparison, we employ our framework on the per-frame processed videos of various applications, including color grading [6], auto-coloring, color harmonization [5], style transfer [35], dehazing [15] [30], and spatially-varying white balance[17]. Experimental results show not only the superiority of our approach but also the validity of the proposed metric.

2 RELATED WORK

Video temporal consistency can be deemed as two similar problems: (1) color transfer with a geometrically identical but temporally inconsistent guidance or (2) post-processing for enhancing temporal coherence. Therefore, we discuss the related works in these two perspectives respectively.

2.1 Color Transfer Methods

As many image processing tasks can be formulated into a color mapping problem, the topic of image color transfer has been widely researched started from Reinhard *et al.* [26]. Among the existing methods, the idea of decomposing the intrinsic image content and style content prevails recently. Xiao and Ma [33] focus on minimizing both the gradient difference between output and source images and the histogram difference between output and reference images. Aubry *et al.* [2] propose a fast local Laplacian filter for multi-scale image manipulation. Yao *et al.* [35] also separate the coarse and detail layers by building Laplacian pyramids, and extend the color transfer method to videos by inferring temporal warping. With the advances in deep learning techniques, Gatys *et al.* [11], Johnson *et al.* [19], and Li *et al.* [21] further resort to convolutional neural networks to train the deep features of image and style contents. Using the per-frame processed video as a guidance, all above can be extended to videos efficiently with the aid of our framework.

2.2 Video Temporal Consistency

Some video processing techniques achieve temporal coherence by introducing temporal filters to specific applications. Bonneel *et al.* [6] transfer the color grade of one video to another via automatic keyframe selection and transformation interpolation. Aydin *et al.* [3] improve video tone mapping by separating the contents into base and detail layers, and temporally filtering the base layer. Ye *et al.* [36] rely on optical flow information and propose a causal-anticausal, coarse-to-fine iterative scheme to stabilize video intrinsic decomposition.

More general approaches have been proposed to deal with a variety of image filters. Paris [24] extends Gaussian kernel to time domain, and adapts algorithms like bilateral filtering and mean-shift clustering to videos. Lang *et al.* [20] temporally apply edge-aware filters on optimization-based techniques such as motion estimation and colorization. Dong *et al.* [10] segment each frame into regions and adjust the enhancement of these regions spatially and temporally. While temporal filtering reduces high-frequency flickering, low-frequency instability remains. Furthermore, spatial details are often compromised for temporal smoothness.

Bonneel *et al.* [7] present a framework of blind video temporal consistency, serving as a post-processing black box that removes flickering from the processed videos. The core spirit is to optimize temporal coherence and preserve the high-frequency dynamics of the per-frame processed video, where the former is calculated by the warping error between successive frames and the later is represented by a gradient term. This is then solved by minimizing the energy function:

$$\int \|\nabla O_n - \nabla P_n\|^2 + w(x)\|O_n - \text{warp}(O_{n-1})\|^2 dx \qquad (1)$$

where O_n and P_n denote the n^{th} frame in the output and per-frame processed videos respectively, x represents the spatial position, and warp() uses backward flow to advect the previous frame to the current frame. The weight of the temporal consistency term, $w(x)$, is regularized by a parameter λ and the temporal consistency between the input video frames V, as shown in equation (2).

$$w(x) = \lambda \exp(-\alpha\|V_n - \text{warp}(V_{n-1})\|^2) \qquad (2)$$

Despite the effectiveness in regular cases, inferring temporal coherence by the warping error between successive frames has an inherent drawback, that is, single previous frame does not contain sufficient information to cover the whole history of a video. For videos with occlusion, where an object once existed but is missing in the previous frame, the method of Bonneel *et al.* may fail to generate a temporally consistent output.

2.3 Evaluation Metrics

Eventually, a reliable evaluation metric for such algorithms is needed. Chikkerur *et al.* [9] has made a thorough review of existing video quality assessment methods, concluding that multi-scale structural similarity (MS-SSIM) of Wang *et al.* [32], natural visual feature based video quality metric (VQM) of Pinson *et al.* [25], and motion tuned spatio-temporal quality assessment of natural videos (MOVIE) proposed by Seshadrinathan *et al.* [27] generally give the best performance in evaluating natural videos. MS-SSIM measures the structural similarity between input and output frames without

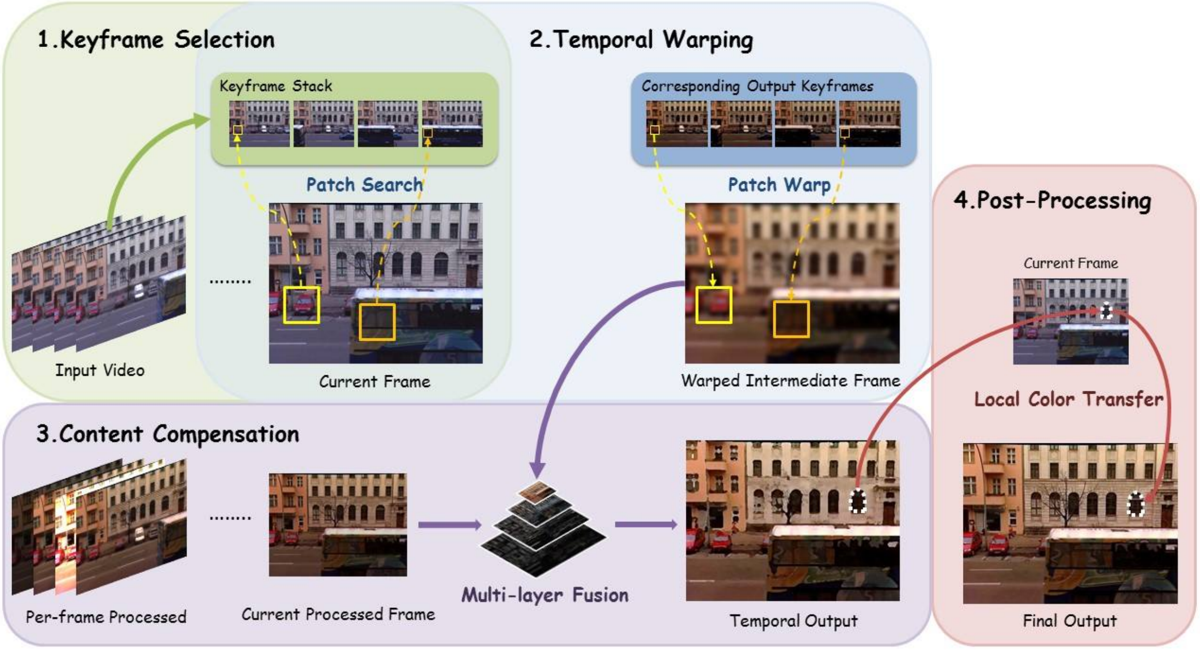


Figure 2: An overview of the proposed framework.

considering temporal consistency. VQM and MOVIE both perform well in evaluating natural videos but do not take the performance of processing techniques into account. Observing that none of these methods focuses on the change of temporal consistency caused by video processing, we refine the quality metric proposed by Yao *et al.* [35], which calculates the warping error of successive frames before and after processing. The high correlation between the proposed metric and the subjective score demonstrates its reliability in reflecting human experience.

3 PROPOSED FRAMEWORK

Our framework takes a raw video sequence $\{V_1, V_2, \ldots\}$ and its per-frame processed version $\{P_1, P_2, \ldots\}$ as inputs, then generates a temporally consistent output $\{O_1, O_2, \ldots\}$. Specifically, temporal coherence is maintained by the step of temporal warping, and the warping error of dynamic objects are compensated through multi-layer fusion. To handle occlusion, we store the history information in a stack of keyframes $\{K_1, K_2, \ldots\}$. Finally, post-processing is employed for denoising and detail reconstruction, producing the final output sequence $\{O'_1, O'_2, \ldots\}$. A framework overview is shown in Fig.2.

3.1 Keyframe Selection

To store visual information of the temporally occluded or disappearing objects, we maintain a stack of keyframes and keep track of the most significant variations throughout the input sequence. Ideally, the keyframe stack ought to contain a maximum number of diverse patches for temporal warping. Nonetheless, keeping a large stack of patches would cost considerable time for search and maintenance. Alternatively, we select a stack of N keyframes with the maximum mutual distance in color distribution. Since natural video frames without scene change typically do not contain large temporal variations of patch textures, a great difference of color histogram between frames could effectively and efficiently indicate a high temporal variation of patch diversity. To practically do it online, we calculate the χ^2 distances D_n between the color histograms of each input frame V_n and all the stored keyframes V_k as equation (3).

$$D_n = \sum_{i=1}^{N} \frac{[\mathrm{H}(V_n) - \mathrm{H}(V_{k_i})]^2}{\mathrm{H}(V_n) + \mathrm{H}(V_{k_i})} \tag{3}$$

where H() calculates the color histogram of an image, and k_i is the index of keyframes. If D_n is greater than that of any keyframe, the current frame is then substituted for the keyframe with the minimal sum of distances to ensure the diversity of keyframe stack.

The size of keyframe stack, N, is crucial to striking a balance between computational complexity and system performance. A large number of keyframes increases the processing time of warping operation, whereas the insufficiency of keyframes leads to poor temporal consistency. In our experiments with video length of 50-200 frames, the setting of N between [5, 10] produces satisfactory output. For longer videos with massive and complicated occlusions, the number of keyframes should be determined based on factors like video length, number of scene changes, and the complexity of object occlusion.

3.2 Temporal Warping

Based on the keyframe stack, we warp a temporally consistent sequence of intermediate frames $\{I_1, I_2, \ldots\}$ by motion estimation.

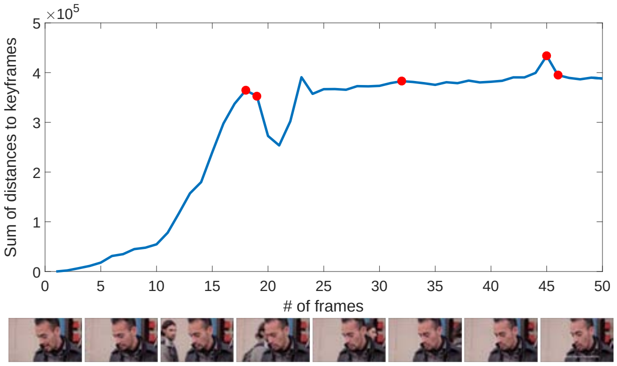


Figure 3: Temporal variation of the histogram distance between the current input frame and keyframes. The red dots indicate the frames selected into the final keyframe stack. Obviously, our approach successfully records the most crucial events throughout the sequence.

The PatchMatch algorithm [4] is applied to find the optimal patch-wise correspondence between the current input frame and the stored keyframes. In detail, for each 7 × 7 patch in the current frame V_n, we search for similar patches p_{k_V} within the keyframe stack, and choose the one with the minimum warping error. The idea can be represented as the following formulation:

$$\hat{v}_\mathrm{I}(x) = \underset{v_k}{\arg\min}\ \|p_\mathrm{V}(x) - \mathrm{warp}_{v_k}(p_{k_\mathrm{V}})\|^2 \tag{4}$$

where $\hat{v}_\mathrm{I}(x)$ is the optimal motion flow for warping at spatial position x, v_k is a three-dimensional vector specifying the horizontal and vertical displacements and the index of keyframe, and $p_\mathrm{V}(x)$ is the unprocessed patch at x. The estimated motion is then used to find the optimal processed patch from the corresponding output patch p_{k_O}:

$$\hat{p}_\mathrm{I}(x) = \mathrm{warp}_{\hat{v}_\mathrm{I}(x)}(p_{k_O}) \tag{5}$$

Finally, these partially overlapping patches, $\hat{p}_\mathrm{I}(x)\ \forall\ x$, are averaged spatially to warp the entire intermediate frame I_n.

Other than PatchMatch, optical flow can be used as an alternative warping operator. We test the performance of PatchMatch and several optical flow techniques, including SIFT flow of Liu *et al.* [22], method of Sun *et al.* [29], and CPM flow of Hu *et al.* [18]. In general, we find that PatchMatch is more robust to rapid motion and resistant to noise, but it introduces a spatially-smoothing effect. Optical flow performs rather well if the raw sequence is originally inconsistent, since it is constrained by gradient and thus more robust to color intensity fluctuations. Comparing the computational cost, while it takes about 30-60 minutes for optical flow to process 50 frames at 960 × 540 resolution, PatchMatch only costs 3-5 minutes.

3.3 Content Compensation

In the patch-warping step, the intermediate frames are warped by averaging the patches with a fixed size and a shape of square, which may blur the edges and damage the detail structures. Moreover, for the objects newly appearing in the current frame, it would be difficult to find patches similar enough to warp them. Hence, we preserve the high-frequency dynamics of the processed frame in the step of content compensation.

Bonneel *et al.* [7] maintain the high-frequency details by solving large linear systems, which costs a great amount of processing time. Instead, we refer to exposure fusion of Mertens *et al.* [23] and construct Laplacian pyramids [8] as a fast multi-scale decomposition. Briefly speaking, layers of a pyramid $\{LP^1, LP^2, \ldots\}$ are defined by the difference between successively up/down-sampled versions of the original image, say, I^l, and $I^{l+1} = \mathrm{downSample}(I^l)$:

$$LP^l = I^l - \mathrm{upSample}(I^{l+1}) \tag{6}$$

where downSample() and upSample() are Gaussian filter operators for sub-sampling and interpolation respectively. To reconstruct the image, a Laplacian pyramid can be *collapsed* by recursively applying $I^l = LP^l + \mathrm{upSample}(I^{l+1})$ until I^1 is obtained.

In our framework, Laplacian pyramids are built to decompose the intermediate and processed frames into layers of different frequencies. Afterwards, a multi-layer fusion is applied on the pyramids, enhancing the detail layers by the processed frame while keeping the coarse layers of the intermediate frames. As shown in equation (7), each layer of the output pyramid LP_o is a weighted fusion of the intermediate and processed pyramids, LP_I and LP_p.

$$LP_o^l = (1 - w^l(x))\ LP_I^l\ +\ w^l(x)\ LP_p^l \quad \forall\ \ l \in \{1, 2, \ldots, L\} \tag{7}$$

The number of layers L is determined depending on the size of video frames. For the test cases with size 960 × 540, we choose L between [4, 6] to give ideal results. The spatial weighting of each layer, $w^l(x)$, is also calculated by down-sampling the bottom layer, which is determined as the normalized warping error:

$$w^1(x) = \frac{\|p_\mathrm{V}(x) - \mathrm{warp}(\hat{p}_{k_\mathrm{V}}(x))\|^2}{\max\limits_{y} \|p_\mathrm{V}(y) - \mathrm{warp}(\hat{p}_{k_\mathrm{V}}(y))\|^2} \tag{8}$$

where y is any spatial position in the frame, and $\hat{p}_{k_\mathrm{V}}$ is the optimal unprocessed patch for warping. If an object has appeared previously, the corresponding patches are expected to be sufficiently similar, so we put more weight on the intermediate frame to guarantee temporal coherence. For the regions with higher warping error, where we assume to be the boundaries or new objects, the patches from the per-frame processed video are weighted higher. Eventually, LP_o is collapsed to generate the output frame O_n.

3.4 Post-processing

While the coarse appearance of output sequence $\{O_1, O_2, \ldots\}$ is temporally smoothed, the video quality may still be unsatisfying compared with the raw sequence in the following cases. (1) The patches and motion flow for warping are noisy. (2) Object boundaries are blurred by the per-frame processing algorithm. (3) Large camera motion or drastic fluctuation of color distribution causes a high warping error, which is particularly severe around the edges.

To further improve color fidelity and detail sharpness, the video frames are segmented into small regions according to their color distribution, and a final output video $\{O'_1, O'_2, \ldots\}$ is generated by applying local affine transformations between the RGB channels of $\{V_1, V_2, \ldots\}$ and $\{O_1, O_2, \ldots\}$. For each segment in V_n and O_n, the pixel values are stored in matrices M_V and M_o, and M_V is augmented with one dimension of 1s. The transformation can be written as:

$$M_{o'} = M_o \, M_{\mathrm{v}}^{+} \, M_{\mathrm{v}} = A \, M_{\mathrm{v}} \tag{9}$$

where $M_{o'}$ is the final output pixel values, M_{v}^{+} is the pseudoinverse of M_{v}, and $A \in R^{3\times4}$ is the affine transformation.

For video segmentation, we experiment on image-based methods like SLIC superpixel of Achanta *et al.* [1] and temporally regularized approaches such as multi-level segmentation of Grundmann *et al.* [14] and supervoxel proposed by Xu *et al.* [34]. Methods of temporal segmentation often sacrifice spatial smoothness to maintain temporal continuity, yet post-processing is aimed to refine details. Therefore, SLIC superpixel is more suitable for our framework. The size of superpixel should be carefully determined. Large superpixels eliminate noise but lead to poor color fidelity. Small superpixels enhance edges but cost more computation time.

Considering that naive application of any color transfer method would cause artifacts around the region boundaries, Gong *et al.* [13] decompose shading adjustment from color transfer for post-processing. Similar in spirit, we smooth the coefficients of affine transformation, A, by an edge-preserving filter. In our testing, guided filter [16] performs better than bilateral filter [31] in noise reduction and boundary preservation.

4 PROPOSED EVALUATION METRIC

When evaluating the performance of temporal consistency algorithms, the quality of both the raw input (V) and the processed video (O) should be taken into consideration. If the successive frames of the raw video are temporally coherent, then a smoother output sequence is expected, and vice versa.

Yao *et al.* [35] propose a temporal consistency metric (TCM) by calculating the warping error between frames. We further modify its formulation and normalize the value into [0,1]. First, the motion field between successive input frames, say V_n and V_{n-1}, are estimated by PatchMatch or optical flow. Then, we use the estimated motion to warp two intermediate frames from V_{n-1} and O_{n-1} respectively. Finally, the refined TCM is defined based on the ratio of the two warping errors, as shown in equation (10).

$$\mathrm{TCM}_n = \exp\left(- \left| \frac{\|O_n - \mathrm{warp}(O_{n-1})\|^2}{\|V_n - \mathrm{warp}(V_{n-1})\|^2} - 1 \right| \right) \tag{10}$$

Note that the motion field used in the warp() operator is the same for V_{n-1} and O_{n-1} so as to examine the same local correspondence. If the current input frame defers significantly from the previous input frame, the warping error of V_{n-1} will be rather large, so a higher warping error of O_{n-1} should be tolerated. Trivially, since the ratio of the two warping errors is better close to 1, larger TCM value indicates higher algorithm performance.

For a better evaluation on videos with temporal occlusions or quality degradation, one can trade computational cost with performance by storing the history information. In the steps of motion estimation and warping, optimal patches for warping can be searched within all the previous frames instead of merely the exactly previous one. It is more complicated and time-consuming to handle cross-frame matching and warping, yet it provides a more reliable evaluation.

5 EXPERIMENTAL RESULTS

Section 5.1 shows a variety of video processing applications. The results are plotted in Fig. 4, where videos 1-6 are provided by the supplemental materials of Bonneel *et al.* [7], and the rest are generated by running the source code of different methods. In section 5.2 we compare the performance of our algorithm with the existing methods by subjective test and the proposed TCM metric.

5.1 Applications

The proposed video temporal consistency framework has a wide variety of applications, including all processing techniques that can be deemed as color transfer. In this section we demonstrate the results of our framework applying on several video processing techniques. It can be seen that these techniques benefit from our framework and the visual experience is improved significantly.

Color Grading By using a reference image or video to guild color grading, one can produce the intended color style in any video sequence. Bonneel *et al.* [6] propose a video color grading algorithm by matching the luminance histogram and chrominance covariance matrix of the segmented foreground and background components. We apply its per-frame process on the test videos shown in Fig. 7 and 9, demonstrating that the results of our general framework is even more temporally consistent than the task-specific approach.

Auto-coloring Automatic enhancement of the color and tone is a typical video processing task. Video 5 shown in Fig. 4 is processed by a combination of Adobe Photoshop's 'Auto Color', 'Auto Contrast', and 'Auto Tone' tools. The high-frequency flickering caused by per-frame processing is greatly removed in the resulting video.

Color harmonization This is a color-matching technique for palette registration of multiple images. Bonneel *et al.* use sliced Wasserstein barycenter [5] to harmonize three videos per frame and generate the output of test video 4 shown in Fig. 4.

Style transfer While similar to color grading, style transfer methods focus on matching the detail structures and texture between images. Yao *et al.* [35] propose an example-based approach to transfer the color style between the reference image and input video. The resulting videos are shown in Fig. 1 and 8.

Dehazing Dehazing is another common color-transferring task which makes images clearer. Algorithms of He *et al.* [15] and Tang *et al.* [30] are applied on test videos 2 and 3 respectively, as shown in Fig. 4. Through the refinements of our framework, they both perform well on videos.

Spatially-varying white balance Hsu *et al.* [17] propose a white balance algorithm by clustering images into regions of different albedo. Since fine-tuning the parameters for each video frame to produce temporally consistent result would be an arduous work, our framework serves as a better solution. The results are shown in video 6 of Fig.4.

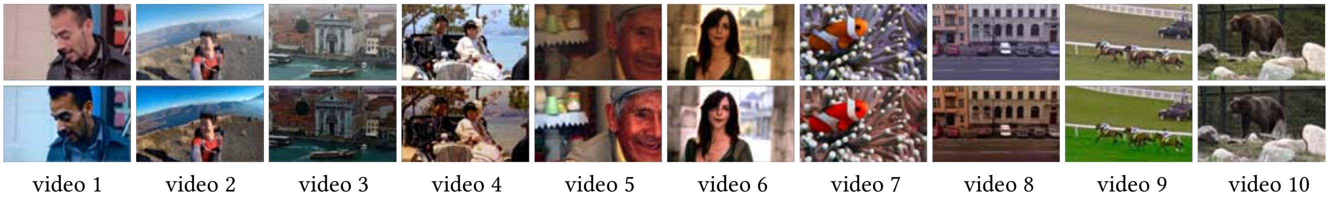


Figure 4: Test videos 1-10. The upper row shows the raw videos, and the lower row shows the processed ones. Videos 1 and 8 captures static scene and huge objects passing in the background. Videos 2, 4, and 9 tracks foreground contents with fast-changing background. Videos 3, 5, 6 focus on content details in slowly shifting scene. Video 7 and 10 contain highly dynamic objects and complex occlusion effects.

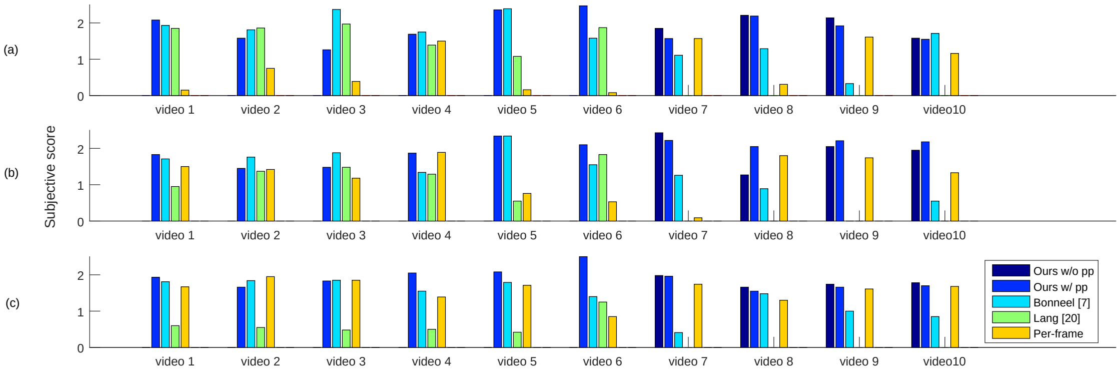


Figure 5: Performance comparison by subjective test with respect to (a) temporal consistency, (b) color fidelity, and (c) detail preservation, which demonstrates the superiority of our framework (either w/ or w/o post-processing). Note that we only compare 4 output versions of each test video for the convenience of subjective ranking. In general, per-frame processed videos suffer from flickering, the method of Lang *et al.* [20] blurs detail structures, and the results of Bonneel *et al.* [7] have poor color fidelity. Our framework performs better overall, especially on the videos with occlusion (videos 1, 6, 7, and 8).

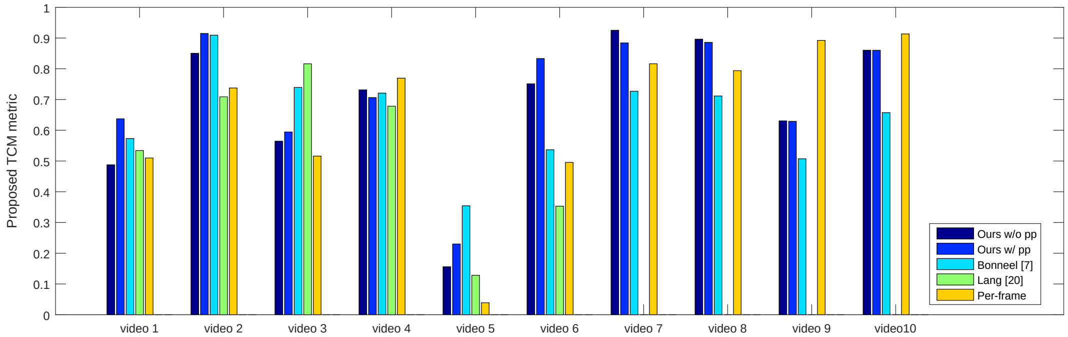


Figure 6: Performance comparison by the proposed TCM metric, which is highly consistent with the subjective scores of temporal consistency. For the raw videos with rapid motion or color fluctuation (videos 2, 9 and 10), TCM may wrongly predict visual experience based on inaccurate motion estimation.

5.2 Comparisons

Considering that Lang *et al.* [20], Bonneel *et al.* [7], and our methods share common applications and means of usage, we compare them as well as per-frame processing in several video streams. The result of subjective measurement is plotted in Fig. 5. For each raw-processed video pair, we ask 20 people to rank 4 versions of output video in aspects of temporal consistency, color fidelity, and detail preservation, separately. 3 scores will be given to the version

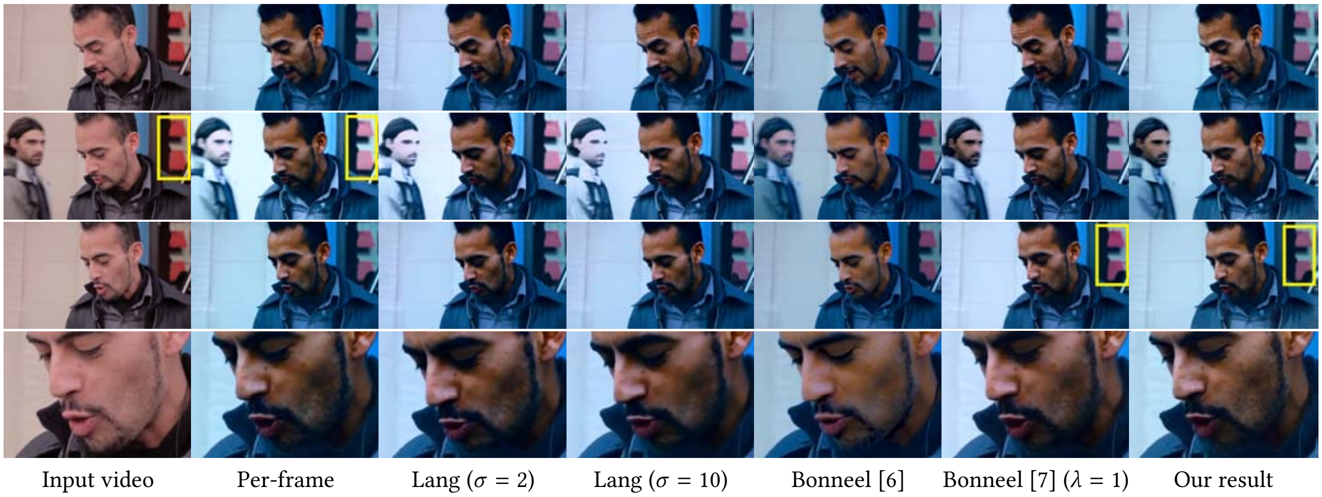


Figure 7: The resulting frames 1, 16, and 40 of color grading application. In the per-frame processed video, the entire background turns pale suddenly when the pedestrian appears, especially the marked red blocks. The method of Lang *et al.* [20] with short kernels ($\sigma = 2$) does not remove low-frequency inconsistency, and long kernels ($\sigma = 10$) create spatial blurring. The approach of Bonneel *et al.* [6] also over-smooths the details on the man's face. Comparing Bonneel *et al.* [7] and our methods, they both successfully eliminate the sudden paleness in the middle frame. However, observing from the final colors of the red blocks, our approach maintains better temporal consistency after occlusion.

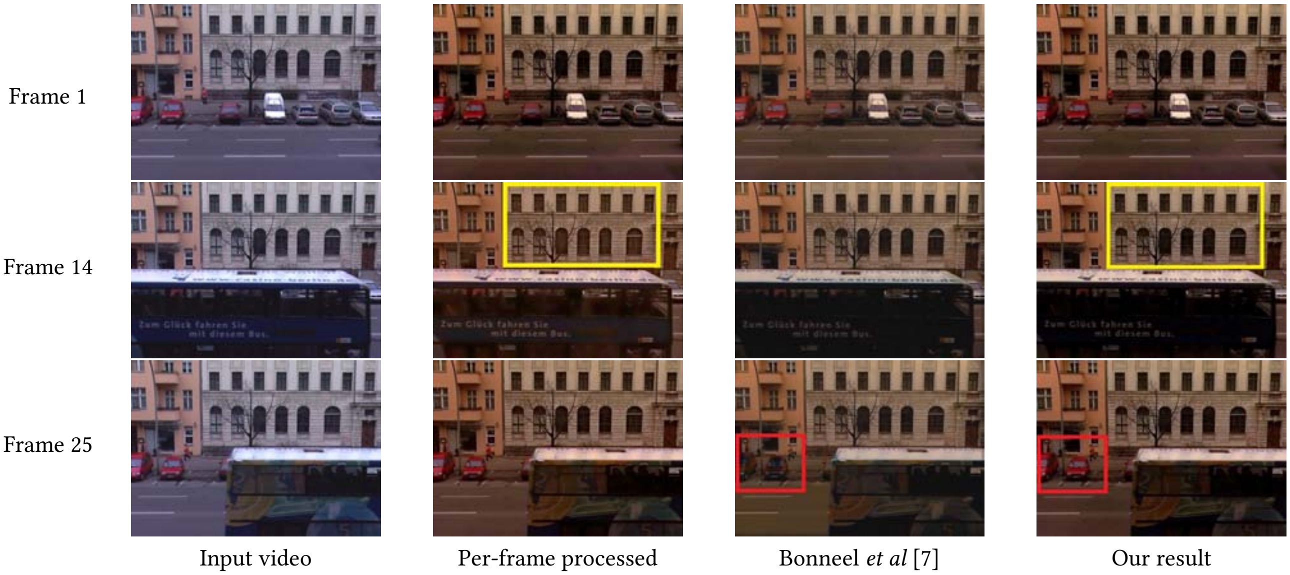


Figure 8: The results of Bonneel *et al.* [7] and our frameworks applied on style transferred video [35]. Obviously, per-frame process causes abrupt brightness on the windows as the bus comes across the scene. In addition, after temporally occluded by the bus, the marked red cars are wrongly colored by the method of Bonneel *et al.*.

with the best quality, 2 for the second place, and so forth. If there is a draw, then the versions with the same quality will share the scores evenly. It can be observed that all three methods averagely produce more consistent outputs than the per-frame processed video, whereas our framework outperforms the method of Bonneel *et al.* [7] in color fidelity, and preserves finer details than the results of Lang *et al.* [20]. Since local color affine model is applied frame-by-frame to mainly refines boundary sharpness and color transformation, the step of post-processing slightly affects temporal coherence, but produces better details and colors. Fig. 6 shows the result of TCM evaluation, which further demonstrates the superiority of our framework. On the other hand, we calculate the numerical correlation scores between the results in Fig. 5 and 6 with respect to consistency, color, and detail, respectively. The

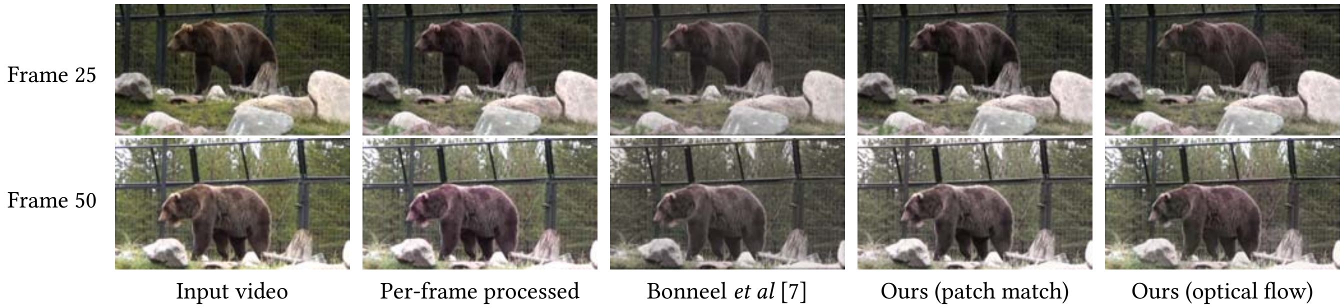


Figure 9: We seek to produce a winter style on the raw video by color transfer [35]. Nevertheless, as the color distribution of the raw sequence is originally fluctuating, it is difficult to track object motion accurately. Hence, most proposed methods fail to give a consistent output. We experiment on substituting the warping operator by optical flow, which is constrained by gradient instead of intensity. The result shows that it indeed is more robust to such cases.

Pearson correlation scores are 0.844, 0.858, 0.821, and the Spearman correlation scores are 0.728, 0.788, 0.731. The high correlation substantiates the effectiveness of the proposed metric in reflecting human visual perception.

In Fig. 7 and 8 we show the resulting frames of different methods. Apparently, the entire frame turns pale suddenly as a large object movement changes the color distribution of the scene rapidly, either the pedestrian in Fig. 7 or the bus in Fig. 8. The approach of Lang *et al.* [20] with short kernels ($\sigma = 2$) fails to eliminate temporal variations, yet longer kernels ($\sigma = 10$) bring a spatially-blurring effect. The method of Bonneel *et al.* [6] also over-smooths the details. Bonneel *et al.* [7] and our approaches both fix the inconsistent problem successfully. However, after the color histogram recovers, some temporally occluded objects may have inconsistent colors, like the red blocks in Fig. 7 and the red cars in Fig. 8. Since method of Bonneel *et al.* uses only the patches of a single previous frame, it is prone to color discrepancy after occlusions. On the contrary, the step of keyframe storage in our framework prevents such problem.

Other issues such as low-frequency color shift and the validity of transferring colors are demonstrated in Fig. 1. Our framework is shown to be robust to not only high-frequency flickering, but also long-term color drift. The drawback of our method is discussed in Fig. 9. With an originally incoherent input video, the Patch Match approach would probably find erroneous patch correspondence. We resort to optical flow for it is gradient-constrained and thus more reliable in such cases.

6 CONCLUSIONS

Temporal coherence in video streams is crucial to visual experience, and it has a wide variety of applications. By inputting an original video and its processed version, which suffers from temporal inconsistency, the proposed framework is able to remove artifacts from the processed video effectively. Furthermore, it is generic to all processing techniques that can be formulated as a color transfer task. Compared with the existing methods, our approach is more robust to occlusion and achieves higher temporal consistency. The proposed TCM metric is also shown to be an appropriate evaluation of video temporal consistency since it matches the results of human visual perception. Although the parameters in our framework, N and L, are already easy to determine, future work on automatic parameter setting can be done for a more convenient use.

7 ACKNOWLEDGEMENT

This work was supported by Himax Technologies, Inc.

REFERENCES

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2010. *Slic superpixels.* Technical Report.

[2] Mathieu Aubry, Sylvain Paris, Samuel W Hasinoff, Jan Kautz, and Frédo Durand. 2014. Fast local laplacian filters: Theory and applications. *ACM Transactions on Graphics* 33, 5 (2014), 167.

[3] Tunç Ozan Aydin, Nikolce Stefanoski, Simone Croci, Markus Gross, and Aljoscha Smolic. 2014. Temporally coherent local tone mapping of HDR video. *ACM Transactions on Graphics* 33, 6 (2014), 196.

[4] Connelly Barnes, Eli Shechtman, Dan B Goldman, and Adam Finkelstein. 2010. The generalized patchmatch correspondence algorithm. In *European Conference on Computer Vision.* 29–43.

[5] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. 2015. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision* 51, 1 (2015), 22–45.

[6] Nicolas Bonneel, Kalyan Sunkavalli, Sylvain Paris, and Hanspeter Pfister. 2013. Example-based video color grading. *ACM Transactions on Graphics* 32, 4 (2013), 39–1.

[7] Nicolas Bonneel, James Tompkin, Kalyan Sunkavalli, Deqing Sun, Sylvain Paris, and Hanspeter Pfister. 2015. Blind video temporal consistency. *ACM Transactions on Graphics* 34, 6 (2015), 196.

[8] Peter Burt and Edward Adelson. 1983. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications* 31, 4 (1983), 532–540.

[9] Shyamprasad Chikkerur, Vijay Sundaram, Martin Reisslein, and Lina J Karam. 2011. Objective video quality assessment methods: A classification, review, and performance comparison. *IEEE Transactions on Broadcasting* 57, 2 (2011), 165–182.

[10] Xuan Dong, Boyan Bonev, Yu Zhu, and Alan L Yuille. 2015. Region-based temporally consistent video post-processing. In *IEEE Conference on Computer Vision and Pattern Recognition.* 714–722.

[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition.* 2414–2423.

[12] Michaël Gharbi, YiChang Shih, Gaurav Chaurasia, Jonathan Ragan-Kelley, Sylvain Paris, and Frédo Durand. 2015. Transform recipes for efficient cloud photo enhancement. *ACM Transactions on Graphics* 34, 6 (2015), 228.

[13] Han Gong, Graham D Finlayson, and Robert B Fisher. 2016. Recoding Color Transfer as a Color Homography. In *British Machine Vision Conference.*

[14] Matthias Grundmann, Vivek Kwatra, Mei Han, and Irfan Essa. 2010. Efficient hierarchical graph-based video segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition.* 2141–2148.

[15] Kaiming He, Jian Sun, and Xiaoou Tang. 2011. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 12 (2011), 2341–2353.

[16] Kaiming He, Jian Sun, and Xiaoou Tang. 2013. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 6 (2013), 1397–1409.

[17] Eugene Hsu, Tom Mertens, Sylvain Paris, Shai Avidan, and Frédo Durand. 2008. Light mixture estimation for spatially varying white balance. *ACM Transactions on Graphics* 27, 3 (2008), 70.

[18] Yinlin Hu, Rui Song, and Yunsong Li. 2016. Efficient coarse-to-fine patchmatch for large displacement optical flow. In *IEEE Conference on Computer Vision and Pattern Recognition*. 5704–5712.

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*. 694–711.

[20] Manuel Lang, Oliver Wang, Tunc Aydin, Aljoscha Smolic, and Markus Gross. 2012. Practical Temporal Consistency for Image-based Graphics Applications. *ACM Transactions on Graphics* 31, 4 (2012), 34:1–34:8.

[21] Chuan Li and Michael Wand. 2016. Combining markov random fields and convolutional neural networks for image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2479–2486.

[22] Ce Liu. 2009. *Beyond pixels: exploring new representations and applications for motion analysis*. Ph.D. Dissertation. Massachusetts Institute of Technology.

[23] Tom Mertens, Jan Kautz, and Frank Van Reeth. 2009. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum* 28, 1 (2009), 161–171.

[24] Sylvain Paris. 2008. Edge-preserving smoothing and mean-shift segmentation of video streams. In *European Conference on Computer Vision*. 460–473.

[25] Margaret H Pinson and Stephen Wolf. 2004. A new standardized method for objectively measuring video quality. *IEEE Transactions on Broadcasting* 50, 3 (2004), 312–322.

[26] E Reinhard, M Adhikhmin, B Gooch, and P Shirley. 2001. Color transfer between images. *IEEE Computer Graphics and Applications* 21, 5 (2001), 34–41.

[27] Kalpana Seshadrinathan and Alan Conrad Bovik. 2010. Motion tuned spatio-temporal quality assessment of natural videos. *IEEE Transactions on Image Processing* 19, 2 (2010), 335–350.

[28] Yichang Shih, Sylvain Paris, Frédo Durand, and William T Freeman. 2013. Data-driven hallucination of different times of day from a single outdoor photo. *ACM Transactions on Graphics* 32, 6 (2013), 200.

[29] Deqing Sun, Stefan Roth, and Michael J Black. 2014. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision* 106, 2 (2014), 115–137.

[30] Ketan Tang, Jianchao Yang, and Jue Wang. 2014. Investigating haze-relevant features in a learning framework for image dehazing. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2995–3002.

[31] Carlo Tomasi and Roberto Manduchi. 1998. Bilateral filtering for gray and color images. In *IEEE International Conference on Computer Vision*. 839–846.

[32] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. 2003. Multiscale structural similarity for image quality assessment. In *Asilomar Conference on Signals, Systems and Computers*, Vol. 2. 1398–1402.

[33] Xuezhong Xiao and Lizhuang Ma. 2009. Gradient-Preserving Color Transfer. *Computer Graphics Forum* 28, 7 (2009), 1879–1886.

[34] Chenliang Xu, Caiming Xiong, and Jason J Corso. 2012. Streaming hierarchical video segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 626–639.

[35] Chun-Han Yao, Chia-Yang Chang, and Shao-Yi Chien. 2016. Example-based video color transfer. In *IEEE International Conference on Multimedia and Expo*. IEEE, 1–6.

[36] Genzhi Ye, Elena Garces, Yebin Liu, Qionghai Dai, and Diego Gutierrez. 2014. Intrinsic video and applications. *ACM Transactions on Graphics* 33, 4 (2014), 80.