Blind Video Temporal Consistency

Nicolas Bonneel[1]* James Tompkin[2] Kalyan Sunkavalli[3] Deqing Sun[2] Sylvain Paris[3] Hanspeter Pfister[2]
[1]CNRS-LIRIS [2]Harvard Paulson School of Engineering and Applied Sciences [3]Adobe Research

(a) Input video (b) Per-frame processing (c) Our result

Figure 1: *With many image filters, such as this automatic color, tone, and contrast adjustment, processing an input video (a) (frames 167-168) frame by frame results in temporal discontinuities (b). We take the two video sequences (a) and (b) and automatically generate a temporally consistent video (c), without knowing the image filter used to produce the unstable video (b). This enables the application of our technique to a wide range of video effects such as color constancy, stylization, color grading, intrinsic decomposition, depth prediction, and dehazing.*

Abstract

Extending image processing techniques to videos is a non-trivial task; applying processing independently to each video frame often leads to temporal inconsistencies, and explicitly encoding temporal consistency requires algorithmic changes. We describe a more general approach to temporal consistency. We propose a gradient-domain technique that is blind to the particular image processing algorithm. Our technique takes a series of processed frames that suffers from flickering and generates a temporally-consistent video sequence. The core of our solution is to infer the temporal regularity from the original unprocessed video, and use it as a temporal consistency guide to stabilize the processed sequence. We formally characterize the frequency properties of our technique, and demonstrate, in practice, its ability to stabilize a wide range of popular image processing techniques including enhancement and stylization of color and tone, intrinsic images, and depth estimation.

CR Categories: I.4.9 [Image Processing and Computer Vision]: Applications I.4.3 [Image Processing and Computer Vision]: Enhancement—Filtering;

Keywords: Video processing; temporal consistency.

*e-mail:nicolas.bonneel@liris.cnrs.fr

1 Introduction

With advances in processing, effective filters for restoration, enhancement, creative edits, and analysis are now common for static images. Videos, on the other hand, do not enjoy the same rich and diverse toolbox. One can naively treat a video sequence as a series of frames and process each one independently with a filter designed for static images. This may work in some simple cases like high-pass and low-pass filtering, but in many other, more sophisticated, cases this produces unsightly results that suffer from flickering. This can occur for various reasons, e.g., a complex optimization technique may fall into different local minima depending on the frame, or a filter may depend on statistics, like the average color, that are not stable throughout the video sequence.

One solution to this problem is to explicitly account for temporal consistency. Several video processing algorithms have been developed along this line, such as color grading [Bonneel et al. 2013], dynamic range compression [Aydin et al. 2014], intrinsic decompositions [Ye et al. 2014; Bonneel et al. 2014; Kong et al. 2014], and tonal stabilization [Farbman and Lischinski 2011]. While effective, these techniques are specific to each task and do not generalize to other problems. Paris [2008] and Lang et al. [2012] propose more generic approaches to extend still image operators to videos. However, these continue to assume a specific filter formulation, which limits their application. For operators outside this set, Lang et al. resort to temporal low-pass filtering. As we shall see, this reduces flickering but does not fully remove it.

We aim for a more general approach to extending image filters to videos, and propose an algorithm that is agnostic to the internal design of the filter, i.e., we treat image filters as black boxes that take input frames and generate processed frames. In that sense, our approach to temporal consistency is *blind* to the image filter being applied. We have two requirements: 1) that the original video is available and that optical or feature flow is recoverable, such that it can be used as a temporal consistency guide, and 2) that the filter

does not generate new content uncorrelated with its input — for instance, painterly rendering filters that procedurally generate brush stroke texture and inpainting techniques that synthesize new content are outside our scope. That said, our approach covers a wide variety of filters such as dehazing, automatic photographic enhancement, color grading, and intrinsic decomposition. We formulate our algorithm in the gradient domain and propose an energy function that amounts to a spatial screened Poisson equation with temporal constraints that we can solve efficiently. We formally characterize the properties of our approach in terms of frequency content. We provide experiments that demonstrate that it can handle a variety of effects, and that it can straight-forwardly extend state-of-the-art image filters to videos.

Contributions

- A technique to remove the flickering due to the frame-by-frame application of an image filter to a video.
- A gradient-domain formulation of the temporal consistency problem that is agnostic to the applied image filter.
- The implicit extension of several state-of-the-art image filters to videos, which are unstable frame-by-frame otherwise.

2 Related work

Some image filters like low-pass filtering and its edge-preserving counterparts produce temporally stable results when applied frame by frame, e.g., [Winnemöller et al. 2006; Chen et al. 2007]. However, many other image filters do not generalize well to videos and need to be adapted.

One popular solution is to focus on a given filter for a specific application. For instance, Bonneel et al. [2013] and Wang et al. [2006] transfer the color grade of one video to another by temporally filtering the color transfer functions. Aydin et al. [2014] make video tone mapping temporally consistent by decomposing HDR content into base and detail layers, and temporally filtering the base layer more aggressively. Bonneel et al. [2014], Ye et al. [2014], and Kong et al. [2014] generate stable video intrinsic decompositions. All these methods work well for their needs, but because the way they enforce temporal consistency relies on the specifics of their target application, they do not generalize to other applications. In comparison, we propose an approach that applies to a large number of image filters.

More general approaches have been proposed to handle entire classes of filters. Paris [2008] extends the Gaussian kernel to the time domain and uses this result to adapt applications such as bilateral filtering and mean-shift clustering to videos. Lang et al. [2012] also extend the notion of smoothing to the time domain by exploiting optical flow and revisit optimization-based techniques like motion estimation and colorization. For other techniques that do not optimize an energy, they resort to temporal low-pass filtering. We shall see in the result section that temporal smoothing reduces high-frequency flickering but low-frequency instabilities remain. Unlike these two approaches, our approach does not require the image filter to have a particular form nor does it adapt to its formulation. Instead, we run the filter without modification as a black box to create a set of filtered frames, then process them to remove temporal artifacts.

Concurrent to this work, Dong et al. [2015] present a technique to stabilize video frames processed by an unknown image filter that can be expressed as nonlinear curves applied to regions of the original video frames. In contrast, our technique is not restricted to a specific formulation and can handle applications like intrinsic images or depth prediction that violate this assumption.

In parallel, techniques have been developed to remove inconsistency in input videos, rather than in videos processed by a filter. These range from capture-time issues like in-camera auto white balancing, to temporal aliasing from mismatched camera/lighting/projector frame rates, to physical phenomena such as the irregular aging of film stock, e.g., [van Roosmalen 1999; Pitié et al. 2004; Pitié et al. 2006; Delon and Desolneux 2010; Farbman and Lischinski 2011; RE:Vision 2015]. These approaches are complementary to our work. They perform well on these tasks, but make domain-specific assumptions such as gray-scale degradation [Pitié et al. 2004], global effects [Farbman and Lischinski 2011], or known degradation models [Pitié et al. 2006] that limit their applicability to other applications. On the other hand, our approach focuses on a different scenario in which the temporal artifacts are generated by an image filter applied on each frame independently. Instability of the input video itself is not covered by our work.

Our approach is similar in spirit to gradient-domain image processing techniques that aim to preserve high-frequency scene content while allowing low-frequency adjustments toward another goal — an approach that has been successful for seamless compositing [Pérez et al. 2003], HDR compression [Fattal et al. 2002], and several other filters including video deblocking [Bhat et al. 2010]. Our work differs from these techniques in that it focuses on video temporal consistency and is agnostic to the applied filter.

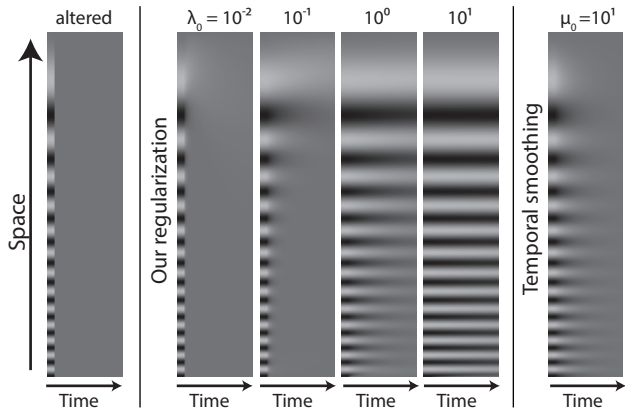


Figure 2: *We perform a synthetic experiment on a 1D video: each frame is a vertical line of pixels. The first frames show a linear chirp, i.e., a signal of increasing spatial frequency, and the remaining frames are uniformly gray. With a low λ_0 value, there is no temporal consistency and the output video transitions instantly back to a uniform color. For intermediate values, the temporal consistency is enforced more strongly on the low frequencies which remain visible for a long time, while the high-frequency region at the bottom is allowed to be temporally discontinuous and quickly becomes uniform. For a large λ_0 value, temporal consistency is enforced on the whole spectrum and the chirp is propagated to all frames. These results are predicted by our analysis (§ 3.2).* Right: *In contrast, temporal smoothing enforces consistency uniformly over the spectrum.*

3 Restoring Temporal Consistency

Our algorithm takes as input an unprocessed video $\{V_0, V_1 \dots\}$ and the same video $\{P_0, P_1 \dots\} = \{f(V_0), f(V_1) \dots\}$ processed frame by frame by some image filter f. The filter f has introduced temporal artifacts that we seek to remove to create a temporally stable video $\{O_0, O_1 \dots\}$.

Spatially, the artifacts in P can be either global or local. For instance, intrinsic image decompositions are defined up to a global

multiplicative factor and algorithms often set this factor arbitrarily, leading to random offsets in each frame. Algorithms that rely on sophisticated optimization schemes are prone to local minima, which makes them overly sensitive to initial conditions and can generate discontinuous local variations between adjacent frames. Further, many optimization schemes are spatially regularized, so variations typically impact an entire object or image region at once — they rarely occur at the scale of a few pixels. In the temporal domain, the profile of these artifacts is arbitrary: they can vary slowly, be random at each frame, or be anywhere in between. We design our algorithm with these characteristics in mind.

One naive approach would be to enforce perfect temporal consistency by warping the first frame by optical flow to recreate each subsequent frame. However, this ignores the inevitable imprecision of accumulated flow fields, and would eventually cause large errors. Further, this scheme does not account for issues like occlusions and appearance variations, e.g., due to lighting changes. In other words, enforcing temporal consistency can come at the expense of scene dynamics. Our solution balances these two aspects.

3.1 Joint optimization of temporal consistency and scene dynamics

We describe our approach in a causal setting: we consider the n^{th} frame ($n > 1$) assuming that the previous frames have already been processed. This processes frames one at a time, which keeps the memory requirement small and enables the processing of arbitrarily-long videos without resorting to complex memory management schemes [Paris 2008].

We formulate the temporal consistency objective with a simple least-squares energy: $\arg\min_{O_n} \int \|O_n - \text{warp}(O_{n-1})\|^2 \text{d}x$, where x represents the spatial position in the frame, and $\text{warp}()$ uses backward flow to advect the previous frame toward the current frame.

For the scene dynamics term, one naive option would be a simple data attachment term: $\arg\min_{O_n} \int \|O_n - P_n\|^2 \text{d}x$. However, P implicitly suffers from temporal inconsistency, and so this term would go against our objective and transfer instability to the output video O. Ideally, we would like to attach O only to the part of P that represents the scene and discard the part that is inconsistent. To achieve this, we draw inspiration from the work of Elder [1999], who showed that a scene is well represented by its edges. We are also inspired by Poisson Image Editing [Pérez et al. 2003], which reproduces the appearance of image regions by copying their gradients. Thus, instead of requiring pixel values to be similar, we require their gradients to be similar. We minimize: $\arg\min_{O_n} \int \|\nabla O_n - \nabla P_n\|^2 \text{d}x$. Intuitively, this can be seen as a data attachment on the scene edges, where the gradients are approximations of the edges. We further analyze this aspect in Section 3.2.

We combine the two terms after modulating the influence of the temporal consistency term by a weight, $w(x)$, that measures the input video consistency V. We set the first frame as a boundary condition and compute O as the minimum of the least-squares energy:

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

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

$$O_0 = P_0 \qquad \text{(1c)}$$

The weighting function $w(x)$ is key to our approach because it relaxes the temporal consistency requirement when the input video V itself is not consistent or the warp inaccurate, which we detect when the warped previous frame does not explain well the current frame (Eq. 1b). In other words, we only impose temporal consistency when the input video is consistent. We use the first frame as a boundary condition (Eq. 1c) which corresponds to the common scenario where users tune the image filter f to achieve the desired result on the first frame and would like to propagate the same quality of results to the rest of the video. It would be straightforward to adjust our scheme to use any frame as reference and to process the video forwards and backwards in time from it. Our formulation is parametrized by λ, which controls the regularization strength (see analysis Sec. 3.2) and α which indicates our confidence in the warp.

To minimize Equation 1a, we use the Euler-Lagrange formula [Weinstock 1974] to derive a differential property that O_n must satisfy at the minimum:

$$-\Delta O_n + w(x) O_n = -\Delta P_n + w(x)\,\text{warp}(O_{n-1}) \qquad (2)$$

where all the quantities on the right-hand side, P_n, $w(x)$, and O_{n-1} are known. This equation is known as the *screened Poisson equation*; it is a variant of the standard Poisson equation with a 0^{th}-order term added. There exist efficient schemes to solve it, e.g., in the frequency domain [Bhat et al. 2008], and it has been used for various image editing applications [Bhat et al. 2010]. In direct relation to our work, Bonneel et al. [2014] used this equation for temporally-consistent intrinsic video decomposition by augmenting it with application-specific terms. In comparison, our approach applies to a large number of applications and does not require a specific formulation of the filter applied to each frame.

3.2 Frequency-domain Analysis

Our approach has a varying impact upon different spatial frequencies in the video, with the low frequencies being more constrained to be temporally consistent than the high frequencies. We analyze Equation 2 in the frequency domain using $\mathcal{F}(\cdot)$ for the Fourier transform and ξ for the spatial frequency. For the sake of simplicity and in this section only, we assume that the weighting function w is constant and equal to λ_0, i.e., $w(x) = \lambda_0$ for all x. Applying the identity $\mathcal{F}(\Delta \cdot) = -4\pi^2\xi^2\mathcal{F}(\cdot)$ to Equation 2 gives:

$$4\pi^2\xi^2\mathcal{F}(O_n) + \lambda_0\mathcal{F}(O_n) = 4\pi^2\xi^2\mathcal{F}(P_n) + \lambda_0\mathcal{F}(\text{warp}(O_{n-1}))$$

which leads to:

$$\mathcal{F}(O_n) = \frac{4\pi^2\xi^2\mathcal{F}(P_n) + \lambda_0\mathcal{F}(\text{warp}(O_{n-1}))}{4\pi^2\xi^2 + \lambda_0} \qquad (3)$$

which is the basis of our analysis.

First, we look at the effect of λ_0. For large values, i.e., $\lambda_0 \to +\infty$, we have $\mathcal{F}(O_n) \approx \mathcal{F}(\text{warp}(O_{n-1}))$, that is, we warp the previous frame — this is the naive solution for temporal consistency that we previously discussed, which does not account for scene dynamics. For small values, i.e., $\lambda_0 \to 0$, for $\xi \neq 0$, we have $\mathcal{F}(O_n) \approx \mathcal{F}(P_n)$, that is, we copy the frequency content of the frame processed by the image filter without imposing any temporal consistency. The DC component ($\xi = 0$) is treated differently though. If $\lambda_0 = 0$, we have a $0/0$ indeterminacy that corresponds to the well-known ill-posedness of the Poisson equation in the absence of boundary conditions. More interestingly, if $\lambda_0 \neq 0$, we have $\mathcal{F}(O_n)(0) = \mathcal{F}(\text{warp}(O_{n-1}))(0)$, i.e., we copy the DC component of the previous frame. That is, even with a small temporal weight, as long as it is non-zero, our approach removes the flickering due to a constant spatial offset.

Next, we assume a general non-zero value of λ_0 and analyze the influence of the unstabilized filtered frame P_n vis-a-vis the stabilized previous frame O_{n-1}. The influence of P_n is proportional to the square of the frequency ξ. As a result, the low frequencies

of the stabilized frame O_n are dominated by the previous frame O_{n-1}, but the high frequencies are closer to those of the output P_n of the image filter. This means that the temporal consistency is more strongly enforced on low frequencies. We illustrate this property on a synthetic example in Figure 2. Recall that our goal is to remove temporal inconsistencies that are mostly constant or large-scale while preserving the scene structure captured by the image edges. By enforcing temporal consistency more strongly on the low frequencies and preserving the high frequencies, our approach fulfills our objective.

In comparison, temporal smoothing can be modeled as averaging the current frame with the previous frame, i.e., $O_n = (P_n + \mu_0 \operatorname{warp}(O_{n-1}))/(1+\mu_0)$ where μ_0 controls the smoothing strength [Paris 2008]. This directly translates to $\mathcal{F}(O_n) = (\mathcal{F}(P_n) + \mu_0 \mathcal{F}(\operatorname{warp}(O_{n-1})))/(1+\mu_0)$. All frequencies are affected equally by the previous frame, i.e., temporal consistency is enforced uniformly over the spectrum.

We analyze the behavior of our temporal consistency algorithm for different scales of temporal noise with a synthetic experiment. We use a 46-frame clip from the Sintel sequence [Butler et al. 2012] — a synthetically rendered animation with ground truth reflectance and shading. We generate random noise, add it to a random per-frame offset, and spatially filter them with a Gaussian filter with varying standard deviations to produce noise with different characteristic scales.

We add this noise layer to the ground-truth reflectance sequence to simulate the temporal artifacts caused by frame-by-frame processing. We scale the noise layer to keep the PSNR of the deteriorated reflectance fixed at 25dB for all noise scales. We simulate common user practice of setting the filter 'as desired' on the first frame by adding no noise to it. We compute the optical flow on the original sequence using the technique of Sun et al. [2014]. Figure 3 illustrates the performance of our temporal consistency technique as the the scale of the added noise is changed.

At the smallest scale, our technique improves the reconstruction of the reflectance by about 4.5dB — this is largely the result of correcting for the random per-frame offset. As the scale of the noise increases, our technique does better, up to a PSNR of about 33dB. We also compare to the temporal smoothing technique of Lang et al. [2012]. Since they use a forward-backward propagation scheme, we adapted Roo and Richardt's [2014] implementation of Lang et al.'s filtering by providing their algorithm with both forward and backward flows computed using the same technique [Sun et al. 2014]. We also added another noise-free frame after a second (frame 30) to simulate the user checking the quality of the output periodically. This has little influence on our approach, but helps Lang et al.'s technique by providing it a reference point to propagate backwards.

For high-frequency noise ($\sigma < 4$ pixels), temporal smoothing performs better but, as the scale of the noise increases ($\sigma > 4$ pixels), our approach becomes better. This suggests that temporal smoothing is better suited for filtering high-frequency perturbations such as sensor noise, but that our approach is better at handling medium and low frequency inaccuracies like the ones introduced by unstable image filters. We show in the validation section that we obtain similar results on real-world videos.

4 Results

This section provides implementation details, comparisons with state-of-the-art techniques, and applications of our technique to several video processing applications. Our temporal consistency can be better observed in the accompanying supplementary video, which also includes additional results.

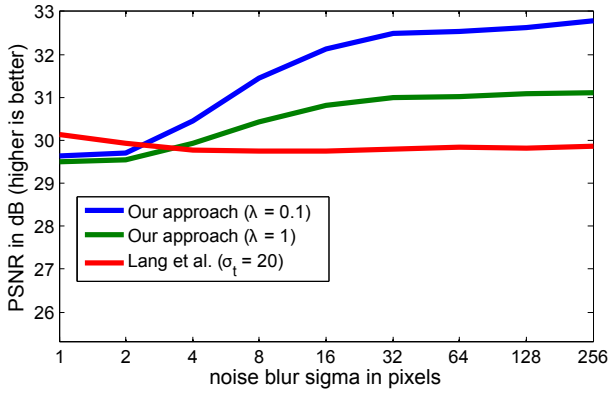


Figure 3: *We analyze our algorithm behavior on a synthetic reflectance-only video by adding a random per-frame offset and noise filtered by a Gaussian filter with increasing standard deviation. The PSNR of the deteriorated reflectance is kept constant at 25.3dB across noise scales. At the finest scale of noise, our temporally coherent reconstruction improves the PSNR by 4.3db to 29.6dB. The temporal smoothing algorithm of Lang et al. performs better (30.1dB). As the scale of the filter increases beyond 4 pixels, this trend changes and our approach performs better with a difference that reaches $+4dB$ for 32 pixels and higher.*

4.1 Implementation

The choice of the $\operatorname{warp}()$ operator in Equation 1a is critical, as inaccurate correspondences across the input video V can result in flickering or bleeding in the stabilized result O. After testing several optical flow techniques [Liu 2009; Werlberger et al. 2010; Sun et al. 2014], we found that the method of Sun et al. [2014] produced satisfactory results on a wide range of videos, despite occasionally introducing minor bleeding. Its main drawback is computational cost, taking 1–2 hours for 100 frames at 1024×576 resolution. We also considered several nearest neighbor field techniques [Barnes et al. 2009; Besse et al. 2012; HaCohen et al. 2011]. PatchMatch [Barnes et al. 2009] provides a complementary option which generates satisfactory results on many examples, including on videos which are challenging for optical flow, and at a fraction of the cost: less than 30 seconds for 100 frames. However, PatchMatch sometimes introduces minor flickering when the estimated correspondence field is discontinuous between two successive frames. In general, both methods were able to produce high-quality results, although which method achieves the most stable output depends on the specific video and application: videos with rapid motion are often better handled with PatchMatch, while applications which remove texture cues such as depth prediction and intrinsic decomposition are better with optical flow. We generated the results in Figures 1, 4, 6, 7, and 8 using PatchMatch, and optical flow was used for Figures 3, 5, and 10.

Having pre-computed correspondences which define the $\operatorname{warp}()$ operator, we solve the linear system of Equation 2 using a fast multiscale solver with Gauss-Seidel iterations. Our approach takes less than 0.40 seconds per frame at 1024×576 resolution. The temporal weight (Eq. 1b) is computed using $\alpha = 0.2$ for all our experiments. To set λ, we start with a value of 1.0, which works in about 75% of cases. We reduce it when we observe bleeding due to optical flow inaccuracy. In practice, a λ value between $[0.05; 1.0]$ produced results without spatial bleeding or temporal flickering.

4.2 Comparisons

We compare our approach to the unaltered filtering algorithm of Lang et al. [2012] on a typical video processing task – automatically enhancing the color and tone of the video frames using a combination of Adobe Photoshop's 'Auto Color', 'Auto Contrast', and 'Auto Tone' tools. Applying these tools on a per-frame basis results in strong high-frequency flickering (Fig. 1) and slow color drifts (Fig. 4). Without access to the original enhancement algorithm, Lang et al.'s approach amounts to temporal smoothing. As shown in Figure 4, using low values of smoothing in their algorithm does not fully remove the temporal flickering, which matches our analysis on synthetic data (Fig. 3). On the other hand, using stronger spatial smoothing leads to undesirable spatial blurring, which we hypothesize is caused by optical flow errors. In all fairness, Lang et al. focused on reformulating energy-based filters and proposed the temporal smoothing that we use here only as a fallback. Comparisons with Lang et al. [2012] on all sequences can be found in supplementary material. We also compare to applying an existing video enhancement tool — Adobe Premiere's 'Auto Colors' enhancement — to the same scene (Fig. 4(f)). We enabled the 'Temporal Smoothing' option, and obtained unstable results even with large smoothing windows. In comparison, our approach generates a temporally-consistent output without any loss of spatial detail.

We also compare to three algorithms explicitly designed to be temporally consistent: the user-guided intrinsic videos technique of Bonneel et al. [2014] (Fig. 5), the video color grading method of Bonneel et al. [2013] (Fig. 6), and the tone mapping of Aydin et al. [2014] (see supplemental material). For intrinsic images, we apply the state-of-the-art technique of Bell et al. [2014] to every frame and use our technique to make the resulting sequence temporally consistent. For color grading, we apply our technique to the frames produced by the single image color transfer technique proposed by Bonneel et al. [2013]. For HDR compression, we used an HDR video sequence produced by Kronander et al. [2013] ; comparisons with other tonemapping operators evaluated by Kronander et al. and Aydin et al. are also shown in supplemental material. In all these cases, we are able to produce results that qualitatively have the same temporal characteristics as these sophisticated video processing algorithms, in spite of having no knowledge about the underlying filter.

In addition, we compare our technique to the commercially available RE:Vision DE:Flicker software [2015]. DE:Flicker operates only on the processed video, P, with no knowledge of the input video, V, and so can be more widely applied than our technique. The approach used in DE:Flicker is unpublished. We evaluated this software on our processed sequences with two types of users: naive users, i.e., us, where we tested on all of our sequences, and expert users, i.e, the authors of DE:Flicker, where three sequences were tested. Please see supplementary material for these results. Naive application often fails to remove all inconsistency, whereas our approach is broadly successful. Expert use is able to reduce more inconsistency (see Old Man autocolors), but still has problems on difficult cases (see Bedroom intrinsic decomposition).

4.3 Applications

The strength of our technique lies in the fact that it makes very few assumptions about the underlying image processing applied to the video frames, and we have explored applying it to a wide range of applications. In particular, we identified several sources of temporal inconsistency across these applications. High temporal frequency flickering is often caused by sensitivity to parameters, problem ill-posedness, or rapid scene content changes. Low temporal frequency artifacts are often caused by smooth variations in scene content. Our algorithm appropriately deal with both kinds of artifacts.

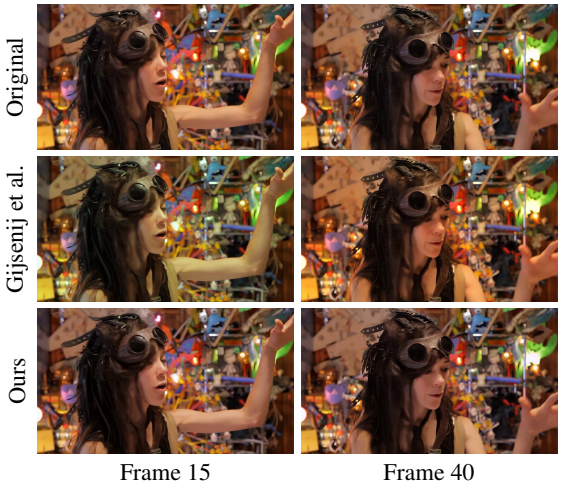


Figure 7: *The gamut mapping of Gijsenij et al. [2010] is not stable, producing different white balance on two different frames. Our approach directly propagates the white balance from previous frames.*

Intrinsic Images The intrinsic decomposition problem consists in decomposing a photograph into a product of reflectance and shading layers. This problem is naturally ill-posed: multiplying one layer by a constant and dividing the other by the same constant results in the same product. In addition, in spite of the use of non-local reflectance priors, low-frequencies are often harder to solve for, and result in low frequency spatial artifacts. The state-of-the-art intrinsic image decomposition algorithm by Bell et al. [2014] produces large temporal inconsistencies when applied per frame, which we regularized with our approach (Fig. 5). We also successfully regularized the intrinsic image decomposition of Zhao et al. [2012] applied per frame (see supplemental material). We produce results with similar quality as the temporally-consistent approach of Bonneel et al. [2014], which is tailored to the intrinsic decomposition problem. While their method is temporally stable by design, it cannot benefit from the improvements of the Bayesian approach of Bell et al., nor any future work on this problem, without modification. Our approach allows for the easy exploration of different image-based techniques as it treats the underlying processing function as a black box.

Color Grading By-example color grading allows the transfer of color style between photographs, which is often performed by matching color statistics. Bonneel et al. [2013] proposes a model consisting of a 1D luminance histogram and a 2D chrominance covariance matrix matching in each segment of foreground-background segmented frames. This model produces temporal inconsistencies that are regularized in a second differential geometric step. We obtain similar temporal consistency from a more general framework (Fig. 6).

Color Constancy These algorithms find the white point of an image and balance it accordingly. This process can be unstable as the white point determined by the system can vary significantly from frame to frame. Color correction requires a linear transform of the red, green, and blue components of each pixel. We applied two gamut mapping methods of Gijsenij et al. [2010; 2012], based on edge derivatives (Fig. 7) and edge weighting (see additional material). Both produce relatively low temporal frequency inconsistencies that our technique eliminates.

(a) Original frames (b) Processed frames (c) Lang ($\sigma = 2$) (d) Lang ($\sigma = 10$) (e) Our result (f) Adobe Premiere

Figure 4: *We process the original video (frames 75 and 90) (a) using the Auto-Color, Auto-Contrast, and Auto-Tone tools in Adobe Photoshop (b). The method of Lang et al. [2012] does not eliminate low temporal frequency variations for short temporal kernels ($\sigma = 2$)(c) and creates spatial blurring for longer kernels ($\sigma = 10$)(d). Our method produces temporally consistent results while retaining the spatial details (e). Automatic color enhancement using video tools like Adobe Premiere does not produce temporally consistent results either (note the brightness change; see the accompanying video for a better depiction).*

(a) Original frames (b) Reflectance frames (Bell et al. [2014]) (c) Our reflectance (d) Bonneel et al. [2014]

Figure 5: *The intrinsic decomposition problem is inherently unstable. Processing the original frames (a) using the single-image technique of Bell et al. [2014] produces reflectance frames with temporal inconsistencies (emphasized in red) (b). In spite of having no knowledge about the intrinsic image problem, our approach produces stabilized reflectance frames (c), that are qualitatively similar to the interactive solution of Bonneel et al. [2014] that explicitly encodes temporal coherence.*

(a) Original frames (b) Single-image color grading (c) Bonneel et al. [2014] (d) Our result

Figure 6: *Bonneel et al. [2013] introduced a single-image color transformation model, which when applied per frame, produces temporal inconsistencies (see brightening in the background) (b). They eliminate this inconsistency by filtering color-transforms in a higher-dimensional space (c); we achieve a similar consistency with our algorithm that is blind to the color transfer (d).*

Spatially-varying White Balance We experimented with the two-illuminant white balancing scheme of Hsu et al. [2008]. This algorithm clusters a photograph into regions with the same albedo and uses them to recover the spatially-varying weights of the two illuminants. This algorithm is sensitive to the clustering step and initial light color parameters. While it is easy to manually determine the best parameters for a particular frame, adjusting these parameters for all frames is cumbersome and would not result in a temporally consistent solution. Instead, we adjusted the parameters for the first frame of the video sequence, and then relied on our algorithm to

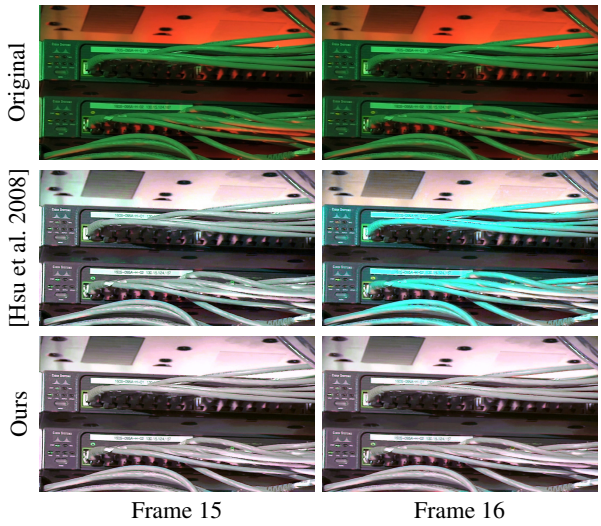


Figure 8: *The local white balancing algorithm of Hsu et al. [2008] applied to video frames is sensitive to initial parameters. Our solution regularizes the output of this algorithm.*

temporally regularize the output of Hsu et al.'s algorithm (Fig. 8).

Color Harmonization This consists in matching and averaging color statistics of multiple images to register their color palettes. It can be used to simulate different photos being taken on the same device, or same setup, or during the same day, even when they were not. We used the sliced Wasserstein barycenter of Bonneel et al. [2015] to harmonize colors of three videos per frame. This resulted in minor flickering which our techniques stabilizes.

Style Transfer Generalizing example-based color grading, Aubry et al. [2014] introduced a method to transfer the style of a particular photograph to an input image. We used their algorithm to process our videos per frame. This yields minor flickering, which we are able to remove with our method (Fig. 9).

HDR Compression We processed HDR frames with the tone mapping operators of Paris et al. [2011], and Durand and Dorsey [2002] on videos by Kalantari et al. [2013] and Kronander et al. [2013] These methods produce mostly low spatial frequency flickering, which is easily removed by our method. We also processed LDR frames with the "HDR Toning" filter of Adobe Photoshop (Fig. 9), and using Adobe Lightroom (highlights, clarity and shadows settings), and again removed the temporal inconsistencies.

Dehazing We applied the algorithms of Tang et al. [2014] and He et al. [2009] to every video frame. While we found the former more temporally stable, both methods benefit from our approach (Fig. 9).

Depth prediction Our algorithm also performs well on vision-related tasks, such as the problem of recovering depth from a single input image. We successfully regularize the method of Eigen et al. [2014] applied per-frame (Fig. 9). Because of the problem difficulty, the resulting depth maps are of low resolution and only produce spatially low frequency artifacts that are easy to remove with our approach.

Figure 9: *We experimented with a number of other filters which produce a flickering too subtle for side-by-side comparison. Among those, the style transfer approach of Aubry et al. [2014], two dehazing methods [He et al. 2009] (a) and [Tang et al. 2014] (b), Photoshop's HDR toning effect, the tone mapping of Paris et al. [2011], and depth prediction [Eigen et al. 2014]. We refer the reader to our accompanying video to appreciate temporal consistency.*

4.4 Discussion

Our approach addresses the problem of temporal instability introduced by applying unstable image filter to videos. It is designed to exploit a pair of unprocessed–processed videos and is not meant to handle the single-sequence scenario. For instance, it cannot remove flickering due to problems at capture time such as sensor noise or temporal aliasing of time-lapse videos, since, in these cases, there is no temporally-consistent input video to be used as a reference.

We found that our approach does not work well on matting because instabilities occur on fuzzy object boundaries which is also where optical flow techniques tend to fail (Fig. 10). Further, as discussed previously, artistic filters that create content uncorrelated with their input are also problematic, such as painterly stylization (Fig 10).

From a practical point of view, although the λ parameter is easy to set and the choice between PatchMatch and optical flow is easy to make (trying PatchMatch first for its computational efficiency), we would

like to automate these steps in future work. In some cases when we use PatchMatch, we observed some mild posterization of the result due to the spatial discontinuities of the estimated correspondence field. For instance, this can be seen on the wall behind the man in Figure 4. However, these issues happen only on a small number of cases and, as shown in our result video, our algorithm stabilizes many image filters that would be unusable on videos otherwise. While the optical flow computation is orthogonal to our technique, we tested the recent PCA and Layered PCA flow methods of Wulff and Black [2015]. We observed that they have similar accuracy to Sun et al. [2014], but are up to $100\times$ faster. Exploring such fast flow algorithms is a promising avenue of future work that can improve the usability of our technique.

5 Conclusion

We have described a blind algorithm to stabilize the output of image processing filters applied frame by frame to videos. Our approach relies on a standard least-squares energy that can be solved with a linear system. We have analyzed the properties of our scheme in the Fourier domain and showed that despite its apparent simplicity, it performs a sophisticated differentiation between high and low frequencies that enables the stabilization of the video without degrading its content. Our experiments show that our technique performs significantly better than temporal smoothing and is able to produce high-quality results on a wide variety of applications independently of their inner workings, thereby helping bring the video processing toolbox closer to parity with that of images.

Acknowledgements

We thank Eugene Hsu and Ivaylo Boyadzhiev for their white balance implementation [Hsu et al. 2008], Xue Bai and Kang In Kim for their suggestions, Pierre Jasmin of RE:Vision for his expertise in processing sequences with DE:Flicker [RE:Vision 2015], and Adobe for their donation. James Tompkin and Hanspeter Pfister thank NSF CGV-1110955, with the work also sponsored by the Air Force Research Laboratory and DARPA Memex program. Finally, we thank the reviewers for their feedback, and the authors of the video footage: G. Mougin (Fig. 1), Unbound (Fig. 4), Millgate (Fig. 5 and 9), G. Henkel (Fig. 6), S. Pyeatte (Zina Nicole Lahr, Fig. 7 and 9), B. Fitzgerald (Fig. 9), P. Shin (Fig. 9) and B. Bourgon (Fig. 8).

References

AUBRY, M., PARIS, S., HASINOFF, S., KAUTZ, J., AND DURAND, F. 2014. Fast local Laplacian filters: Theory and applications. *ACM Trans. on Graphics (SIGGRAPH)*.

AYDIN, T. O., STEFANOSKI, N., CROCI, S., GROSS, M., AND SMOLIC, A. 2014. Temporally coherent local tone mapping of hdr video. *ACM Trans. Graph. 33*, 6 (Nov.), 196:1–196:13.

BARNES, C., SHECHTMAN, E., FINKELSTEIN, A., AND GOLDMAN, D. B. 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. on Graphics (SIGGRAPH) 28*, 3.

BELL, S., BALA, K., AND SNAVELY, N. 2014. Intrinsic images in the wild. *ACM Trans. on Graphics (SIGGRAPH) 33*, 4.

BESSE, F., ROTHER, C., FITZGIBBON, A., AND KAUTZ, J. 2012. PMBP: PatchMatch belief propagation for correspondence field estimation. In *BMVC - Best Industrial Impact Prize award*.

BHAT, P., CURLESS, B., COHEN, M., AND ZITNICK, C. L. 2008. Fourier analysis of the 2D screened Poisson equation for gradient domain problems. In *ECCV*, 114–128.

BHAT, P., ZITNICK, C. L., COHEN, M., AND CURLESS, B. 2010. GradientShop: A gradient-domain optimization framework for image and video filtering. *ACM Trans Graph (SIGGRAPH) 29*, 2.

BONNEEL, N., SUNKAVALLI, K., PARIS, S., AND PFISTER, H. 2013. Example-based video color grading. *ACM Trans. on Graphics (SIGGRAPH) 32*, 4.

BONNEEL, N., SUNKAVALLI, K., TOMPKIN, J., SUN, D., PARIS, S., AND PFISTER, H. 2014. Interactive intrinsic video editing. *ACM Trans. on Graphics (SIGGRAPH Asia) 33*, 6.

BONNEEL, N., RABIN, J., PEYR'E, G., AND PFISTER, H. 2015. Sliced and radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision 51*, 1, 2245.

BUTLER, D. J., WULFF, J., STANLEY, G. B., AND BLACK, M. J. 2012. A naturalistic open source movie for optical flow evaluation. In *European Conf. on Computer Vision (ECCV)*, 611–625.

CHEN, J., PARIS, S., AND DURAND, F. 2007. Real-time edge-aware image processing with the bilateral grid. *ACM Trans. on Graphics (SIGGRAPH)*.

DELON, J., AND DESOLNEUX, A. 2010. Stabilization of flicker-like effects in image sequences through local contrast correction. *SIAM Journal on Imaging Sciences 3*, 4, 703–734.

DONG, X., BONEV, B., ZHU, Y., AND YUILLE, A. L. 2015. Region-based temporally consistent video post-processing. In *IEEE Conference on Computer Vision and Pattern Recognition*.

DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. In *Proc. of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, ACM, SIGGRAPH '02, 257–266.

EIGEN, D., PUHRSCH, C., AND FERGUS, R. 2014. Depth map prediction from a single image using a multi-scale deep network. In *NIPS'14*, 2366–2374.

ELDER, J. H. 1999. Are edges incomplete? *Int. J. Comput. Vision 34*, 2-3 (Oct.), 97–122.

FARBMAN, Z., AND LISCHINSKI, D. 2011. Tonal stabilization of video. *ACM Trans. on Graphics (SIGGRAPH) 30*, 4, 89:1 – 89:9.

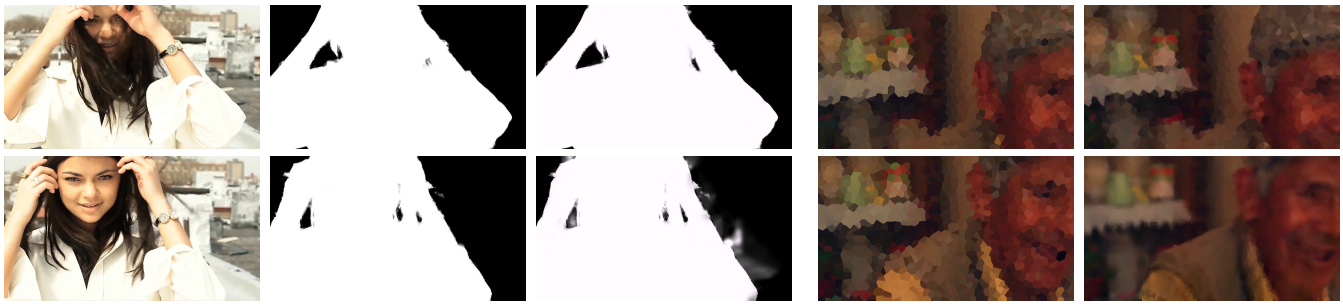
FATTAL, R., LISCHINSKI, D., AND WERMAN, M. 2002. Gradient domain high dynamic range compression. *ACM Trans. on Graphics (SIGGRAPH)*.

GIJSENIJ, A., GEVERS, T., AND VAN DE WEIJER, J. 2010. Generalized gamut mapping using image derivative structures for color constancy. *Int. J. Comput. Vision 86*, 2-3, 127–139.

GIJSENIJ, A., GEVERS, T., AND VAN DE WEIJER, J. 2012. Improving color constancy by photometric edge weighting. *IEEE Trans on Pattern Analysis and Machine Intelligence 34*, 5, 918–929.

HACOHEN, Y., SHECHTMAN, E., GOLDMAN, D. B., AND LISCHINSKI, D. 2011. Non-rigid dense correspondence with applications for image enhancement. *ACM Trans. on Graphics (SIGGRAPH) 30*, 4, 70:1–70:9.

HE, K., SUN, J., AND TANG, X. 2009. Single image haze removal using dark channel prior. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1956–1963.



(a) Original frames (b) Matte frames (Shahrian et al. [2013]) (c) Our result (d) Processed frames (e) Our result

Figure 10: Left: *The matting problem (here, Shahrian et al. [2013]) (b), produces temporal artifacts on object boundaries, precisely where the optical flow is unreliable due to occlusions, which leads to bleeding (c).* Right: *Some NPR effects, such as the 'Crystallize' tool of Adobe Photoshop, produce temporally inconsistent edges which do not gracefully blend in time (d). This leads our method to lose the NPR style (e).*

HSU, E., MERTENS, T., PARIS, S., AVIDAN, S., AND DURAND, F. 2008. Light mixture estimation for spatially varying white balance. *ACM Trans. on Graphics (SIGGRAPH)*, 70:1–70:7.

KALANTARI, N. K., SHECHTMAN, E., BARNES, C., DARABI, S., GOLDMAN, D. B., AND SEN, P. 2013. Patch-based High Dynamic Range Video. *ACM Trans. Graph. (SIGGRAPH Asia) 32*, 6.

KONG, N., GEHLER, P. V., AND BLACK, M. J. 2014. Intrinsic video. In *Eur. Conf. Comp. Vision (ECCV)*, vol. 8690, 360–375.

KRONANDER, J., GUSTAVSON, S., BONNET, G., AND UNGER, J. 2013. Unified HDR reconstruction from raw CFA data. *IEEE Int. Conference on Computational Photography (ICCP)*.

LANG, M., WANG, O., AYDIN, T., SMOLIC, A., AND GROSS, M. 2012. Practical temporal consistency for image-based graphics applications. *ACM Trans. Graph. (SIGGRAPH) 31*, 4, 34:1–34:8.

LIU, C. 2009. *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. PhD thesis, Massachusetts Institute of Technology.

PARIS, S., HASINOFF, S. W., AND KAUTZ, J. 2011. Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid. *ACM Trans. on Graphics (SIGGRAPH)*, 68:1–68:12.

PARIS, S. 2008. Edge-preserving smoothing and mean-shift segmentation of video streams. In *ECCV*.

PÉREZ, P., GANGNET, M., AND BLAKE, A. 2003. Poisson image editing. *ACM Trans. on Graphics (SIGGRAPH) 22*, 3.

PITIÉ, F., DAHYOT, R., KELLY, F., AND KOKARAM, A. 2004. A new robust technique for stabilizing brightness fluctuations in image sequences. In *Statistical Methods in Video Processing*. Springer, 153–164.

PITIÉ, F., KENT, B., COLLIS, B., AND KOKARAM, A. 2006. Localised deflicker of moving images. In *IEEE European Conference on Visual Media Production*.

RE:VISION, 2015. De:flicker v.1.3.0. `http://www.revisionfx.com/products/deflicker/`.

ROO, J. S., AND RICHARDT, C. 2014. Temporally coherent video de-anaglyph. In *SIGGRAPH Talks*.

SHAHRIAN, E., RAJAN, D., PRICE, B., AND COHEN, S. 2013. Improving image matting using comprehensive sampling sets. In *IEEE Conf. Comp. Vision and Pattern Recognition*, 636–643.

SUN, D., ROTH, S., AND BLACK, M. J. 2014. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *Int. J. Comput. Vision 106*, 2, 115–137.

TANG, K., YANG, J., AND WANG, J. 2014. Investigating haze-relevant features in a learning framework for image dehazing. In *IEEE Conf. Comp. Vision and Pattern Recognition*, 2995–3002.

VAN ROOSMALEN, P. M. B. 1999. *Restoration of archived film and video*. TU Delft.

WANG, C.-M., HUANG, Y.-H., AND HUANG, M.-L. 2006. An effective algorithm for image sequence color transfer. *Mathematical and Computer Modelling 44*, 78, 608 – 627.

WEINSTOCK, R. 1974. *Calculus of variations : with applications to physics and engineering*. Dover books on advanced mathematics. Dover. Originally published by McGraw-Hill, in 1952.

WERLBERGER, M., POCK, T., AND BISCHOF, H. 2010. Motion estimation with non-local total variation regularization. In *IEEE Conference on Computer Vision and Pattern Recognition*.

WINNEMÖLLER, H., OLSEN, S. C., AND GOOCH, B. 2006. Real-time video abstraction. *ACM Trans. on Graphics (SIGGRAPH)*, 1221–1226.

WULFF, J., AND BLACK, M. J. 2015. Efficient sparse-to-dense optical flow estimation using a learned basis and layers. In *IEEE Conference on Computer Vision and Pattern Recognition*.

YE, G., GARCES, E., LIU, Y., DAI, Q., AND GUTIERREZ, D. 2014. Intrinsic Video and Applications. *ACM Trans. Graph. (SIGGRAPH) 33*, 4.

ZHAO, Q., TAN, P., DAI, Q., SHEN, L., WU, E., AND LIN, S. 2012. A closed-form solution to retinex with nonlocal texture constraints. *IEEE Trans. Pattern Anal. Mach. Intell. 34*, 7, 1437–1444.