# Motion Driven Tonal Stabilization

Oriel Frigo, Neus Sabater, Julie Delon, Pierre Hellier

***Abstract*—This paper addresses the problem of tonal fluctuation in videos. Due to the automatic settings of consumer cameras, the colors of objects in image sequences might change over time. We propose here a fast and computationally light method to stabilize this tonal appearance, while remaining robust to motion and occlusions. To do so, a minimally-viable color correction model is used, in conjunction with an effective estimation of dominant motion. The final solution is a temporally-weighted correction, explicitly driven by the motion magnitude, both visually efficient and very fast, with potential to real time processing. Experimental results obtained on a variety of sequences outperform the current state of the art in terms of tonal stability, at a much reduced computational complexity.**

***Index Terms*—Tonal stabilization, White balance, Exposure control, Video editing.**

## I. INTRODUCTION

The increasing number of amateur video footages facilitated by the proliferation of low-cost video cameras has made it visible a number of video artifacts, some of them being motion and tonal instabilities. While motion stabilization has been studied by several researchers, tonal instability has been far less discussed. Video tonal instability is a particular temporal artifact characterized by fluctuations in the colors of adjacent frames of a sequence. According to [1], in modern videos these instabilities are mainly caused by automatic settings of the camera, notably automatic white balance and automatic exposure. These common features of consumer digital cameras are intended respectively to provide color balanced and well exposed images, while facilitating the user experience. However, these features are mostly appropriated for still images and are not stable in time, resulting in unpleasant tonal instabilities that can be perceived in videos. A notable problem with automatic white balance algorithms is their dependency on illuminant estimation, which is considered an ill-posed problem. Classical assumptions of color constancy algorithms are easily violated in practice and in a context of temporal scene changes, it is likely to result in chromatic instability.

While automatic white balance can be simply turned off in some cases, low end cameras offer no control over setup parameters. In this case, the only alternative to avoid unpleasant tonal fluctuations is to further process the video. This preprocessing can also be crucial for computer vision applications relying on brightness or tonal constancy assumptions.

Generally speaking, tonal stabilization can be described as searching for the transformations that minimize undesired tonal variations in multiple images of a sequence. While surprisingly few works have specifically attempted to correct such color fluctuations in videos [1], [2], numerous works are motivated by similar purposes in different research communities.

Let us start with a word on radiometric calibration approaches, which takes into account operations performed in the camera pipeline in order to retrieve a physically based color calibration model between images. Recent models [3], [4], [5] admit that the output image $u$ of a camera is related to the irradiance vector $\boldsymbol{e}$ measured by the sensor by a combination of linear (white balance, color space conversion) and nonlinear (gamut mapping, camera response function) transformations. If the nonlinear part of the mapping could be estimated for a video sequence, it could be inverted and the resulted sequence could be corrected with linear transformations. Unfortunately, recovering the camera response function necessitates registered images under multiple exposures, as well training sets of RAW-sRGB image pairs [6], [4], both being generally not available for smartphone and point-and-shoot video cameras.

In a related direction, one could argue that color constancy algorithms would be the ideal solution to solve the tonal stabilization problem. Starting with Land and McCain work on the Retinex theory of color vision [7][8][9], color constancy has been extensively studied and remains an active research topic [10] [11]. In digital cameras, the feature intended to approximate color constancy is known as automatic white balance (AWB). The most common approach to perform AWB is to first estimate the color temperature of the illuminant in the scene, and then correct the image by compensating the ratio of the estimated illuminant to the canonical (neutral) illuminant with some variant of a Von Kries diagonal transformation. However, illuminant estimation is a severely under-constrained problem, necessitating additional assumptions intended to cope with its ill-posed nature. Gijsenij *et al* [11] in their extensive survey on computational color constancy, have categorized illuminant estimation approaches into *static* methods, based on fixed parameter settings and low-level statistics, in opposition to *gamut-based* and *learning-based* methods, which tune their parameters to a specific set of images. Automatic white balance in low cost video cameras is likely to be based on static methods, making strong assumptions about the scene content, so that illuminant color can be estimated in real time. Since this automatic feature is the cause of tonal instabilities, recovering the scene illuminant in each frame to correct the sequence is also prone to fail in practice. This is true even if the available computing power makes it possible to use more involved white balance methods than those used directly in the camera.

Another possibility is to draw on color transfer, which aims at modifying the colors of an input image according to the colors of an example image. Unlike color constancy, color transfer methods make very few assumptions about the physics of the scene or the camera (the scene illuminant or the camera response function are not explicitly estimated), being rather a general and pragmatic approach to estimate color transformations between images. Reinhard *et al.* [12] popularized the concept of color transfer 15 years

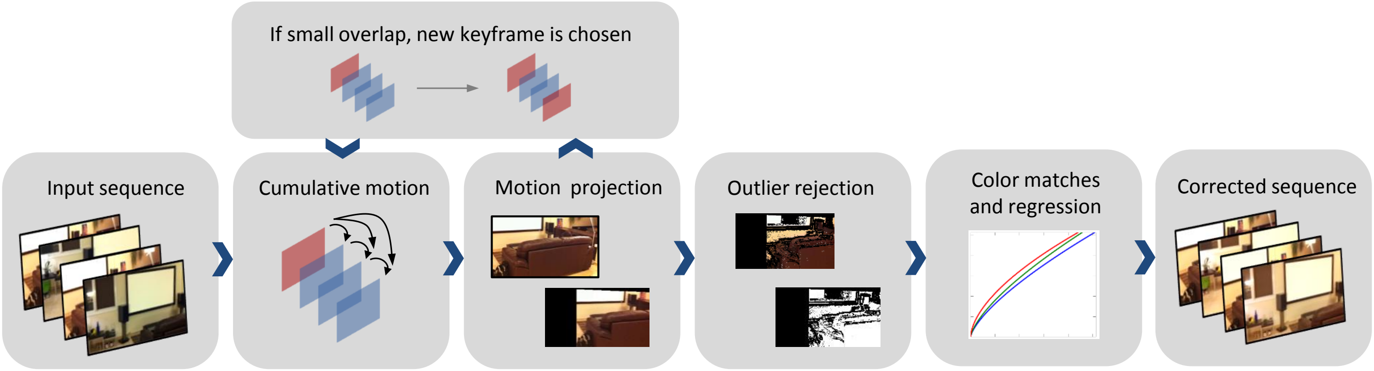

Fig. 1. Overview of proposed method for tonal stabilization of videos.

ago with a very simple affine matching between 3D color distributions. Other works have since proposed more complex global color transfer in terms of nonlinear histogram matching [13], [14], [15], or optimal transportation between compact color signatures [16], [17], [18], [19], [20]. Another class of methods assumes that there are spatial correspondences to be found between the input and example images, these correspondences being used to derive a color transformation [21]. If this assumption reduces the scope of the method for general images, it is particularly relevant for specific user cases such as optimizing color consistency in photos from the same scene [22], [23]. In the case of videos, we also expect to find numerous spatial correspondences between neighbor frames of the sequence. As we will see, the tonal stabilization algorithm presented in this paper draws on a first raw motion estimation between frames to compute a global color correction.

While very few works have attempted to correct color fluctuations in videos, several global [24], [25] or local [26], [27] approaches have been proposed to remove high frequency brightness fluctuations, also known as flicker. In practice, the causes and effects of flicker in videos are quite different from those of tonal instabilities. In old archived films, these high frequency brightness fluctuations are mainly due to physical reasons, which may cause the flicker to vary along the image spatial coordinates. On the contrary, tonal instabilities observed in modern videos (fluctuations in camera exposure or white balance) tend to be global and are low frequency artifacts. As a consequence, parametric and non-parametric deflickering methods are not well suited to correct tonal instabilities. Moreover, the extension of these deflickering methods to color images is far from obvious.

To the best of our knowledge, only two previous works [1], [2] have proposed approaches specifically designed to correct color fluctuations in videos. The first one was proposed by Farbman *et al* in [1]. The method works by aligning tonally each frame of the video with one or several preselected reference (anchor) frames (for instance the first frame of the movie). Adjustment maps between successive frames are computed without explicit motion compensation, the authors claiming that many pixels in the same grid position in two successive frames are likely to correspond to the same surface in the scene. In practice, the method provides a reasonable solution to stabilize tonal fluctuations in static videos, but at the cost of high space and time complexity. Besides, the lack of real motion estimation makes the computation of the adjustment map highly sensitive to noise or fast movements, and lack of accurate correspondences can result in flickering and error propagation.

The second one, due to Wang *et al.* [2], starts by estimating motion globally between successive frames, by relying on local features correspondences. A nine parameters affine color transformation is then used to model the exposure and white balance changes between two frames in the log domain. These affine color transformations are estimated by least squares for all neighboring frame pairs and accumulated to obtain a ”color state” for each frame of the video, represented as a $4 \times 4$ matrix. To avoid data overfitting, a regularization term is added to force the affine transformations to be close to the identity matrix. In a second step, the frame color state function is smoothed by minimizing an energy which also tends to control the deviation from the original color state. The regularization step necessitates to use PCA on the set of color states in order to avoid smoothing the different parameters of the affine matrices independently. While the results provided by the authors are visually good and much more satisfying than those of [1], the method is surprisingly complex to implement and requires to tune several parameters.

We are aware of existing commercial tools (*Adobe After Effects®, Final Cut Pro®*) able to correct specific brightness fluctuations, such as high frequency flickering commonly seen in time-lapse photography. Nevertheless, these applications remain limited when it comes to correct tonal instabilities. The *Color Stabilizer Tool* [1] from *Adobe After Effects* is based on color sampling of selected points in a reference frame, adjusting the colors neighbor frames so that the color values of selected points remain constant throughout the duration of the layer. The *Flicker Free* plugin[2] for *Final Cut Pro* can be useful to remove high frequency flicker and equalize the exposure of footage, but is not targeted to stabilize white balance fluctuations.

The method prosented in this paper draws on two elementary observations. First, we show that a simplified parametric color transfer model between the frames of a video is enough

[1] https://helpx.adobe.com/after-effects/using/color-correction-effects.html

[2] https://digitalanarchy.com/Flicker/main.html

to correct convincingly color instabilities. Second, we observe that if motion estimation is necessary for tonal stabilization, dominant motion estimation is generally more than enough to infer large sets of correspondences between frames. These are the two key ideas permitting to develop a method which is both visually efficient and very fast, intended to work near-real time on smartphones and similar devices. Furthermore, we propose a temporal weighting scheme, where the intensity of tonal stabilization is directly guided by the motion speed. The present paper extends our work published in [28], presenting extensive experiments on the goodness of fit of the power law color transformation model, and comparing our model with state-of-the-art color transfer models.

The outline of the paper is the following. Our method is presented in Section II, first in the simplified case of registered images, and then generalized to sequences with motion. Section III illustrates the efficiency of the proposed method on different sequences and shows that it compares favorably to [1], [2], and achieve these results at a much reduced computational cost. All results are available on the project website[3].

## II. Proposed Method

In this section, we present the main contributions of our method for video tonal stabilization. Our aim is to conceive a method that is:

1) Accurate enough to correct color instabilities observed between frames in a video;
2) Robust against motion, occlusion and noise;
3) Computationally simple enough to be implemented in a near real time application.

Figure 1 presents a general overview of the proposed method. In order to achieve robustness against motion and occlusion (an important limitation of [1]) while satisfying the simplicity requirement, we make use of dominant motion estimation (and compensation) to estimate the color correspondences between frames. By means of cumulative motion, we are able to quickly register images even if they differ by several frames in time. Color correspondences are then used to estimate a color transformation that is applied to correct the tonal instabilities. The requirement of physical accuracy for the tonal transformation model being in contradiction with the other requirements of robustness and simplicity, our color correction model tries to achieve a good tradeoff between these three properties.

The proposed algorithm makes two hypotheses on the sequences to be corrected. First, it is assumed that there are always spatial correspondences (or redundance in content) between neighboring frames in the sequence (no scene cuts). This is generally true for every sequence composed of a single shot, as long as it does not pass through extreme variations of scene geometry (nearly total occlusion) or radiometry (huge changes in illumination or saturation). Second, we assume that there is a global transformation which can compensate the colorimetric aberrations between the frames. This implies that the observed color instability and consequently the camera response function are global (spatially invariant). In other words, the proposed method is not suitable for correction of local tonal instabilities. This assumptions also holds for every method of the state of the art [1], [2].

In the following subsections, we discuss in detail each step of the proposed method. For the sake of simplicity, we start the discussion with the tonal transformation model, first assuming the simplest case of color correction between registered images. We then present our model to deal with the general case of tonal stabilization of sequences containing motion.

### A. Tonal Transformation Model

Let $\{u_t\}_{t=1,\dots,D}$ be a registered sequence of color frames $u_t : \Omega \to \mathbb{R}^3$ defined over the same spatial domain $\Omega$ and let $u_k$ be a reference keyframe in the sequence. The color channels of $u_t$ are written $(u_t^R, u_t^G, u_t^B)$. In order to tonally stabilize the sequence, for every following frame $u_t, t > k$ we look for a parametric color transformation $T_t : \mathbb{R}^3 \to \mathbb{R}^3$ such that $T_t(u_t) \simeq u_k$. Now, as recalled in the introduction and according to [5] , $u_t$ is related to the irradiance vector $\boldsymbol{e} = (e^R, e^G, e^B)$ measured by the sensor (RAW intensities) by the relation

$$u_t = f \circ h(\boldsymbol{T_s T_w} \mathbf{e}) \tag{1}$$

where $\boldsymbol{T_s}$ is a $3 \times 3$ matrix transformation that accounts for color space conversion, $h : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ is a nonlinear gamut mapping operator, $f : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ is the nonlinear tone mapping of the camera and $\boldsymbol{T_w}$ is a diagonal $3 \times 3$ matrix accounting for white balance and exposure (varying over time)

$$\boldsymbol{T_w} = \begin{bmatrix} \phi & 0 & 0 \\ 0 & \xi & 0 \\ 0 & 0 & \psi \end{bmatrix}. \tag{2}$$

If $u_t$ and $u_k$ are two perfectly registered images of the same scene, taken by the same camera, differing only with respect to white balance and exposure (so that these images have identical irradiance $\boldsymbol{e}$), then $u_t$ and $u_k$ are related by

$$H^{-1}(u_t) = \begin{bmatrix} \frac{\phi_t}{\phi_k} & 0 & 0 \\ 0 & \frac{\xi_t}{\xi_k} & 0 \\ 0 & 0 & \frac{\psi_t}{\psi_k} \end{bmatrix} H^{-1}(u_k), \tag{3}$$

where $H = f \circ h \circ \boldsymbol{T_s}$. In theory, we can achieve tonal stabilization between images $u_t$ and $u_k$ with a simple diagonal transformation performed in the camera sensor space (given by the nonlinear transformation $H^{-1}$). This tonal stabilization model, inspired by radiometric calibration [5] [29], can be seen as an accurate procedure to perform camera color transfer when irradiances are known in the form of RAW images, allowing an estimate of $H$. However, for the problem of tonal stabilization, we are faced with videos taken with low-cost cameras, from which we cannot make the usual assumptions that are necessary to compute radiometric calibration.

The assumption of multiple exposures from the same scene, which is required to estimate the camera response function may not be valid for some sequences, and RAW-sRGB correspondences are also not available in most low-cost cameras.

[3] http://oriel.github.io/tonal_stabilization.html

Alternatively, blind gamma estimation could be obtained when multiple images from the same scene are available, for example in the case of video sequences. In [30] the non-linearity of the camera response is modeled as a gamma correction, defined as a single power transformation applied to all color channels. As discussed in [30], estimating the optimal gamma parameter is an optimization problem that is non-trivial to solve and may need a trial and error approach.

In this paper, we follow another direction and claim that a much simpler model can be used for achieving visually satisfying tonal stabilization when the transformation $H$ is unknown. Several paths can be followed, from fully non parametric color transformations (such as weighted interpolation or optimal transport / histogram specification) to more or less complex parametric models (affine, spline, polynomial). Parametric models have the important advantage of being expressed by smooth and regular functions, well defined for the whole color range (extrapolation is not a problem even if all colors are not observed) and potentially described by few parameters, which reduces the risk of instabilities in time. A purely diagonal model applied to sRGB images is unfortunately not suitable to cope with non linearities inherent to camera response functions. In our quest for a tonal transformation model that is simple enough to be fastly computed, and accurate enough to produce a visually pleasant stabilized sequence, we converged to a separable power law transformation, which is shown to be a good compromise to fit the criteria described above. Perhaps not surprinsingly, this model is quite close to the CDL (Color Decision Lists) model used for color grading in post-production industry [4].

Our deliberately simplified model for $T_t$ assumes a separable transfer function on color channels:

$$T_t = (T_t^R, T_t^G, T_t^B), \text{ where } T_t^c(u) := \alpha_c u^{\gamma_c},\ c \in \{R, G, B\}. \tag{4}$$

The accuracy (in term of goodness of fit) of this power law model will be demonstrated in the experimental section on several image pairs of the same scene, with varying white balance and exposure values. The ability of the model to achieve stabilization results more or equally satisfying than more complex models on image sequences will also be shown.

In order to estimate $\alpha_c, \gamma_c$, we solve for every color channel of the frame $u_t$ the linear least squares fitting problem

$$\arg\min_{\alpha_c, \gamma_c} \sum_{\boldsymbol{x} \in \Omega} \big(\log u_k^c(\boldsymbol{x}) - (\gamma_c \log u_t^c(\boldsymbol{x}) + \log \alpha_c)\big)^2, \tag{5}$$

whose solution is given by

$$\gamma_c = \frac{\mathrm{Cov}\big(\log u_t^c, \log u_k^c\big)}{\mathrm{Var}\big(\log u_t^c\big)}, \alpha_c = \exp(\overline{\log u_k^c} - \gamma_c \overline{\log u_t^c}) , \tag{6}$$

where $\overline{z} = \frac{1}{\#\Omega} \sum_{\boldsymbol{x} \in \Omega} z(\boldsymbol{x})$ is the average value of $z(\boldsymbol{x}),\ \boldsymbol{x} \in \Omega$.
Note that this is not equivalent to solving $\arg\min_{\alpha_c, \gamma_c} \sum_{\boldsymbol{x} \in \Omega} \big(u_k^c(\boldsymbol{x}) - \alpha_c u_t^c(\boldsymbol{x})^{\gamma_c}\big)^2$, since the loss function in Eq. 5 becomes logarithmic and gives more weight to residuals computed from lower values. However, this formulation permits to compute the coefficients $\alpha_c$ and $\gamma_c$ exactly and very quickly (the computation is linear in the number of correspondent points). For higher regression accuracy in terms of linear mean squared error, the solution can be computed with a numerical method such as gradient descent [31], [32].

The model we propose in Eq. 4 is quite similar to the one proposed in the work of color stabilization in [23], where the tonal transformation is also modeled as a combination of a linear term and a power term. Both models can be seen as practical approximations to the unknown inverse camera tonal transformation studied in [5]. But differently to [23], we do not consider the $\gamma_c$ coefficient in our model to be equivalent to the parameter $\gamma$ in the gamma correction of the camera imaging pipeline ($\gamma = \frac{1}{2.2}$ in sRGB) [5]. We rather assume that the power $\gamma_c$ in our model approximates possible non-linear color changes between the images of a sequence, in the spirit of CDL color correction.

Another difference between the two models is that [23] includes a full $3 \times 3$ matrix in the tonal transformation, while our model is separable over color channels. Therefore, the model of [23] can take into account channel correlations and possible color space conversions that can take place when correcting images taken from different cameras. Nevertheless, in our case we consider sequences taken entirely with the same camera, and as we show in our experiments, our tonal transformation model is effective in practice with straightforward optimization in comparison to the model proposed in [23].

### B. *Image Registration*

For estimating tonal transformations between different frames, it is desirable to have a dense set of correspondences, especially in homogeneous areas. In this paper, we choose to rely on global motion estimation. Estimating the dominant motion of the camera is computationally simpler (it can potentially be computed in real time) than dense optical flow, and in our experience, tonal instabilities seen in videos are usually highly correlated with the camera motion. In contrast to tasks that depend heavily on precise motion, we do not need a highly accurate motion description in order to estimate a color transformation that compensates tonal differences between frames. It is much more important that this dominant motion estimation is robust enough to be cumulated in time and permit color correction between non-neighbor frames.

In practice, for all pairs of consecutive frames $u_l$ and $u_{l-1}$ in the sequence, we make use of the robust algorithm of [33] to estimate the affine motion transformation $A_{l,l-1}$ between the frames. This planar affine transformation, defined by 6 parameters, only accounts for the dominant motion of the camera without considering pixel-wise accuracy. Dominant motion has the advantage of being computationally simple to estimate, and is generally sufficient for the task of tonal stabilization.

Now, let $u_k$ be a reference frame and $u_t$ ($t > k$) a subsequent frame in the video. Before applying the transfor-

[4] http://en.wikipedia.org/wiki/ASC_CDL

mation model in Eq. (9), $u_t$ is warped towards $u_k$ with the accumulated transformation

$$A_{t,k} = A_{t,t-1} \circ A_{t-1,t-2} \circ ... \circ A_{k+1,k}. \tag{7}$$

We define the set of spatial correspondences $\Omega_{t,k}$ between $u_k$ and $u_t$ as

$$\Omega_{t,k} = \Bigg\{ \big(\boldsymbol{x}, A_{t,k}(\boldsymbol{x})\big) \in \Omega \times \Omega \ \Bigg| \ \frac{1}{3} \sum_c \Big( (u_k^c(\boldsymbol{x}) \ - \overline{u_k^c}) - (u_t^c\big(A_{t,k}(\boldsymbol{x})\big)) - \overline{u_t^c}) \Big)^2 < \sigma^2 \Bigg\}, \tag{8}$$

where $\sigma^2$ is the empirical noise variance (for instance, estimated with the method in [34]). Note that the constraint in Eq. (8) rules out possible motion outliers as well as occluded points (points visible in only one of the frames).

Note that the obtained dominant motion is affine, representing translations and rotations, however it has the limitation of not taking into account zooming of objects. Nevertheless, in practice we observed that camera zoom is a feature rarely used by users shooting videos with low-cost cameras. Moreover, our method usually leads to a reasonable tonal stabilization even in the case of innacurate motion registrations, by means of our keyframe update approach discussed in the next section.

### C. Motion driven tonal stabilization

Let us describe how keyframes are defined: following an online procedure, the first frame of the sequence $u_1$ is initially defined as keyframe. The following frames $u_t, t > 1$ are stabilized with respect to $u_1$ as long as there are enough correspondences between $u_t$ and $u_1$. In other words, a color correction is computed only if there are enough spatial correspondences to define an accurate and valid model. As soon as the number of correspondences $\#\Omega_{t,1}$ is lower than $\omega \times \#\Omega$, a new keyframe is defined as the previously color corrected frame. This process is repeated till the end of the sequence, assuring tonal coherence across neighbor frames and also across different keyframes in an online manner.

To ensure a fidelity to the colors of the input sequence $\{u_t\}_{t=1,...,D}$, we also introduce a regularization term for the transformation sequence $\{T_t(u_t)\}_{t=1,...,D}$ as following:

$$T_t' = \lambda T_t + (1-\lambda) Id, \tag{9}$$

where $\lambda = \lambda_0 \exp(-\frac{||V_{t,k}||}{p})$ controls the amount of transformation of frame $u_t$, according to the motion amplitude between frame $u_t$ and the keyframe, $||V_{t,k}||$ denotes the norm of the dominant motion vector $V_{t,k}$ (global translation from the spatial coordinates of $u_t$ to the spatial coordinates in $u_k$) and $p$ is the maximum spatial displacement (number of rows + number of columns of the image), $\lambda_0$ is the initial weight (in practice we set $\lambda_0 := 0.9$). In other words, the smaller the motion magnitude, the closer $\lambda$ is to 1 and the greater the transformation. In the following, this temporally-varying regularization is useful to correct the sequence while avoiding overexposure when huge changes in camera exposure occur in the original sequence.

Algorithm 1 sketches the proposed method. Note that the computation of $\Omega_{t,k}$ involves the computation of $A_{t,k}$ and the warping of $u_t$ towards $u_k$ based on $A_{t,k}$.

**Algorithm 1** Motion driven tonal stabilization

```
Input: Sequence of frames $\{u_t\}_{t=1,...,D}$
Output: Tonal stabilized sequence $\{T_t'(u_t)\}_{t=1,...,D}$
$k \Leftarrow 1,\ t \Leftarrow k+1$
$T_1'(u_1) = u_1$
while $t \le D$ do
    Compute $\Omega_{t,k}$
    if $\#\Omega_{t,k} \ge \omega \times \#\Omega$ then
        for $c \Leftarrow \{R, G, B\}$ do
            $\alpha_c, \gamma_c \Leftarrow \arg\min_{\alpha,\gamma} \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \Omega_{t,k}} (u_k^c(\boldsymbol{x}) - \alpha u_t^c(\boldsymbol{y})^\gamma)^2$
            $T_t'(u_t^c) \Leftarrow \lambda \alpha_c (u_t^c)^{\gamma_c} + (1-\lambda) u_t^c$
        end for
        $t \Leftarrow t+1$
    else    # If there are not enough correspondences
        $k \Leftarrow t-1$
        $u_k \Leftarrow T_{t-1}'(u_{t-1})$
    end if
end while
```

For the sake of simplicity, the original frames are downsampled (120 pixels wide) before estimating $T'$. Then, the estimated $T'$ is coded into a Look-Up-Table (LUT) that is applied to the high resolved original frames. These implementation changes guarantee an algorithm with low complexity but do not produce noticeable loss in tonal stabilization accuracy.

## III. EXPERIMENTAL RESULTS

In this section we first study the goodness of fit of the proposed power law model. Then, we show the experimental results obtained with our tonal stabilization algorithm and we compare them with state-of-the-art results. On the one hand, qualitative evaluation based on visual inspection is performed and, on the other hand, quantitative results measuring the amount of tonal variation in the resulting sequence and the fidelity to the original sequence are provided. Both, quantitative and qualitative evaluation prove that the proposed algorithm is accurate and robust with all the tested sequences independently of the amount of tonal instabilities or motion.

### A. Goodness of fit

In order to evaluate the accuracy of our power law model, we have estimated the mean $R^2$ (coefficient of determination) along color channels:

$$R^2 = \frac{1}{3} \sum_c \left( 1 - \frac{\sum_{\boldsymbol{x} \in \Omega_p} (\log u_k^c(\boldsymbol{x}) - T_t^c(\log u_t^c(\boldsymbol{x})))^2}{\sum_{\boldsymbol{x} \in \Omega_p} (\log u_k^c(\boldsymbol{x}) - \overline{\log u_k^c})^2} \right). \tag{10}$$

where $\Omega_p$ is the set of points selected from a color chart in the image. In particular, we consider images captured with a smartphone from the same scene containing a Macbeth color chart (see Fig. 2). Each picture is adjusted (using the camera settings) to have a different exposure or white balance (WB), so that we can analyze the tonal changes by studying the color transfer function between the reference picture (sunlight WB, medium exposure) and the other ones. More specifically, we use the median color value of each color in the color chart to estimate a power law transformation. In the right column of Fig. 2, we plot the functional relationship (in logarithmic domain) between the colors extracted from the color chart of the reference picture and the correspondent colors from the other pictures. As an indication of goodness of fit, the computed $R^2$ value is shown for each plot (the closer is $R^2$ to 1, the better the observed tonal transformation fits the model). The coefficient is larger than 0.9 for all the computed regressions, which shows that the relationship between reference and test color intensities is approximately linear in a logarithmic scale and in general the model fits the data.

Note that the images with the color chart are useful to evaluate our model but they are not enough challenging to compare our results with other methods in the case of videos. In fact, all methods are comparable with this data since the presence of all the colors in the chart help the camera automatic white balance algorithm to work properly, without producing strong tonal instabilities.

Finally, we note that the tonal fluctuations caused by automatic camera settings in videos are far less intense than the tonal changes presented in Fig. 2, which were produced by manually adjusting the camera settings.

### B. Qualitative evaluation

In practice, our method has been tested on 18 different video sequences. While some sequences have been kindly provided by the authors of [1], we have completed our dataset with video sequences acquired with smart phones from different manufacturers. Complete video sequences (originals and results) are available at the project website[5]. We strongly recommend the reader to look at the electronic version of the paper and the videos in our website to appreciate the results.

First, we have considered video sequences in which camera motion is not complicated as in the sequence "sofa" (see Fig. 3). In the original sequence one same object appears with different colors (e.g., sofa) while in our resulting sequence all colors are stable.

We present our tonal stabilization result for a sequence with fast motion in Figure 4. The video, taken while driving in a highway, is particularly challenging because of the driving speed, the fast motion of objects, the rain droplets falling and the wiper blade movements.

We note that camera zoom is challenging to be estimated by our dominant motion model. Nevertheless, our tonal stabilization method compensates innacurate motion estimation with the keyframe update approach. We have observed in practice that for sequences where the registation is not accurate, the number of motion outliers increases and keyframe updates are triggered with more frequence. For example, in the driving sequence in Fig. 4 we can see objects changing in scale, nevertheless our model still correct tonal instabilities in the sequence and does not generate artifacts.

In order to evaluate our results with respect to state-of-the-art tonal stabilization methods, we have considered the methods of Farbman et al. [1] and Wang et al. [2]. In our comparison, the results from [2] have been provided by the authors while the results from [1] come from our implementation of their method, which is coherent with the results published in their paper and website. In particular, Fig. 5 compares our results with the sequence "building" that has been acquired with a Samsung Galaxy S smart phone. Note that the results from [1] are not perfectly stabilized due to the important camera motion of the sequence, and the results from [2] are stabilized but the sequence is much whiter and parts of it are completely saturated which is not visually pleasant. Fig. 6 compares with the sequence "graycard" the same algorithms which turn out to have the same behavior in terms of remnant tonal variation for [1] and wash-out look (white) for [2]. On the contrary, in the two sequences "building" and "graycard", our results are both stable and color coherent with a good dynamic range.

Note that all video sequences are processed with the same set of parameters, i.e., $\omega = 0.25$ and $\lambda_0 = 0.9$. Concerning the parameter robustness in our method we have observed that varying $\omega$ has very little impact on the results. On the contrary, the choice of $\lambda$ (cf. Eq. 9) determines the amount of fidelity to the original sequence. Indeed, $\lambda = 1$ corresponds to a strict tonal stabilization and $\lambda = \lambda_0 \exp(-\frac{||V_{t,k}||}{p})$ weights temporally (in function of motion) the contribution to the original sequence. While temporal weighting reduces the strict tonal preservation, it may be argued that it produces a visually pleasant result by preserving some degree of the original colors. In particular, when the original sequence presents high tonal variations, temporal weighting avoids to create a resulting sequence too different from the original in which colors may appear saturated or not natural. Fig. 7 shows an example of a sequence with important tonal instabilities in the original sequence and the results with the different choices of $\lambda$. The strict tonal stabilization result ($\lambda = 1$) saturates some parts of the resulting frames (e.g. t= 170) which motivates our choice of $\lambda$.

### C. Quantitative evaluation

*1) Comparison of color transformation models:* In order to evaluate our power law color transformation, we compare it to the parametric color transformation proposed by [23]. We made experiments with the tonal transformation model proposed in [23] in two ways, based on trustworthy color correspondences between the reference and test images:

- estimating all the 10 coefficients of the model by gradient descent,
- estimating the 9 coefficients of the $3 \times 3$ matrix by gradient descent, fixing $\gamma := \frac{1}{2.2}$

To guarantee that no outliers are used for coefficient estimation, the color correspondences used to compare our method to [23] are taken from the mean 24 colors of the Macbeth color chart, in particular, we use the same test images and

[5] http://oriel.github.io/tonal_stabilization.html

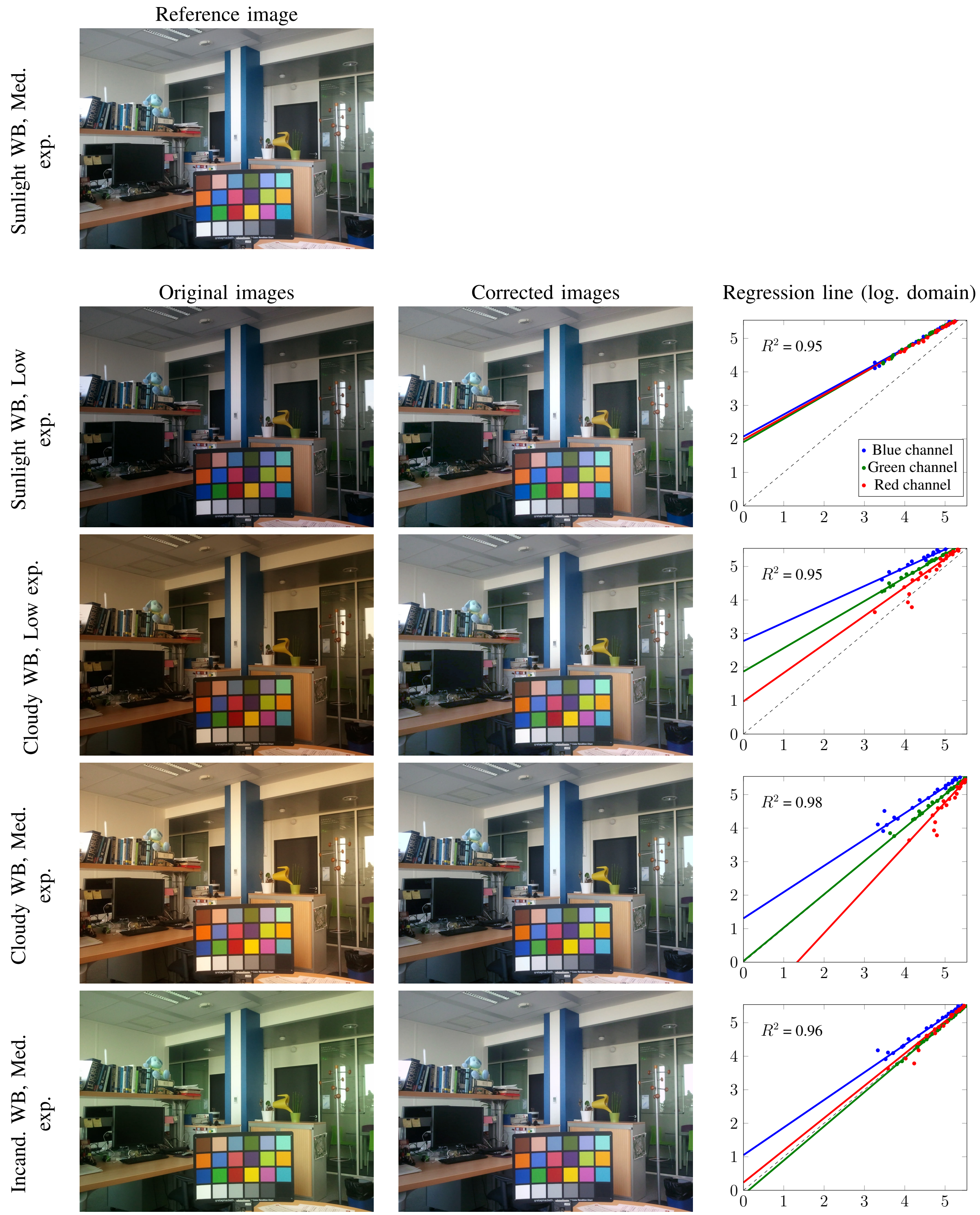

Fig. 2. Illustration of the goodness of fit of the proposed tonal transformation model. In this experience, a sequence of approximately registered images is taken from the same scene, and the color transformation is estimated from correspondent points (taking into account the small dominant motion between the images as explained in Sec. II-B) and masking the Macbeth color chart. On the first row, a reference image (keyframe $u_k$) is shown. For the following second to fifth rows, on the left column, tonally unstable images ($u_t$) taken with different exposures and white balance are shown. On the middle column, the color corrected images are shown, where it can be noted that tonal instability is largely reduced. Finally, on the right column, the extracted points from the color chart and the estimated regression lines are plotted in the logarithmic domain. The dashed black line corresponds to the identity, the $x$-axis corresponds to $\log u_t^c(\Omega_p)$, while the $y$-axis corresponds to $\log u_k^c(\Omega_p)$ for the plotted points and $\log T_t^c$ for the plotted lines, where $\Omega_p$ is the set of color chart coordinates. The regression line has a close fit to the color points, reminding that $R^2$ values which are close to $1$ are an indication of a good fit.

TABLE I
COMPARISON OF DIFFERENT COLOR TRANSFORMATION MODELS. MEAN PSNR IS COMPUTED FROM IMAGES OF FIGURE 2.

| | Sunlight WB, Low exp. | Cloudy WB, Low exp. | Cloudy WB, Med. exp. | Incand. WB, Med. exp. |
|---|---|---|---|---|
| Original | 13.40 | 12.46 | 18.68 | 23.73 |
| Our Power Law | **37.62** | **28.84** | 29.27 | 34.39 |
| [23], fixed $\gamma$ | 26.34 | 25.86 | 30.37 | 34.37 |
| [23], estimated $\gamma$ | 26.81 | 26.62 | **30.67** | **36.30** |

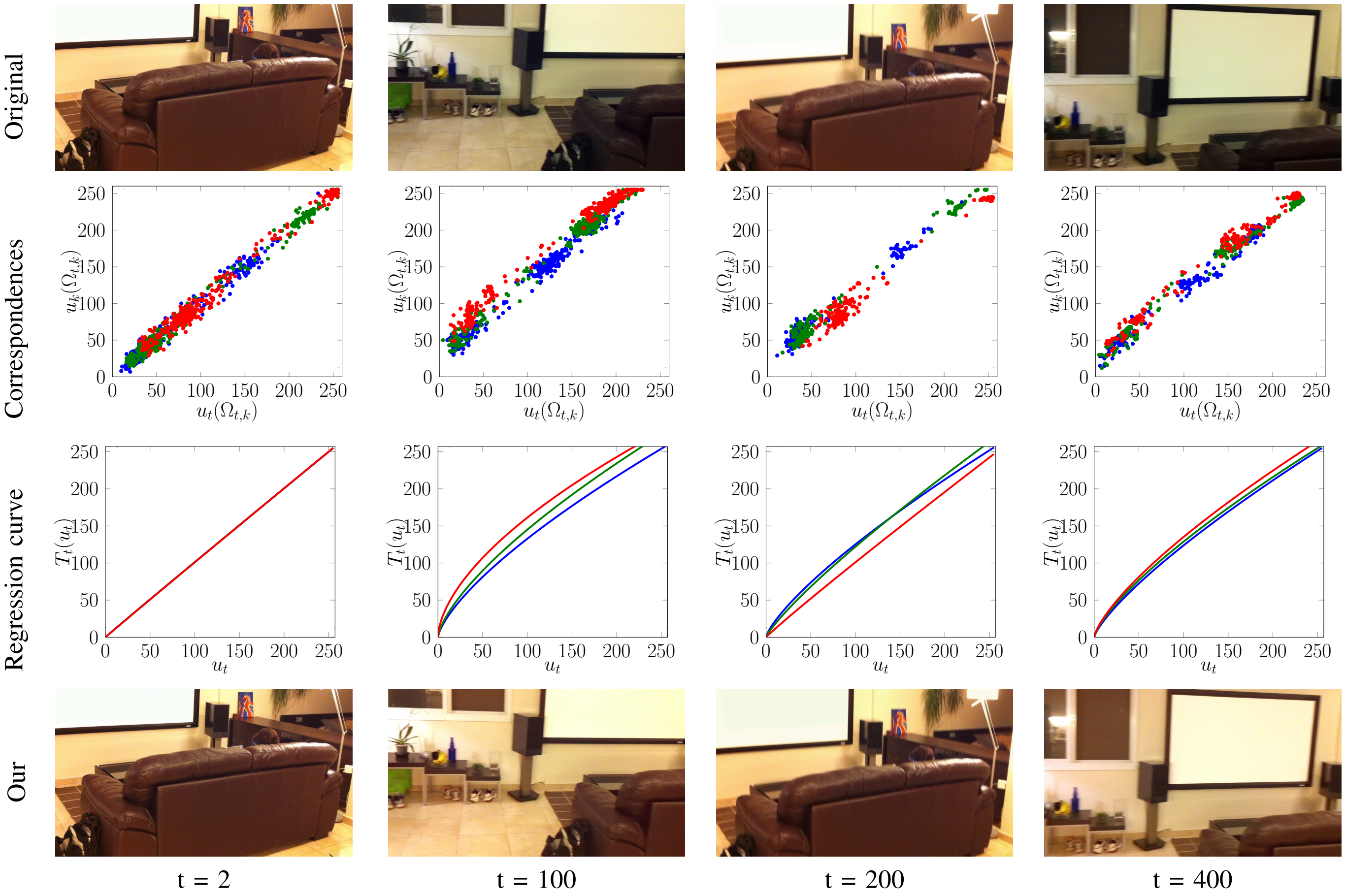

Fig. 3. Tonal stabilization of the sequence "sofa". Top row: frames extracted from the original sequence, t = (2, 100, 200, 400). Second row: plot of point correspondences between the original frame $u_t$ and the keyframe $u_k$. Third row: estimated power law tonal transformation for each frame. Bottom row: same frames from top row, after tonal stabilization with our method. Note that objects appearing with different colors in the original sequence have the same color in our results. The color of the plotted points and curves correspond to the sRGB color channel (Red, Green, Blue) of the image.

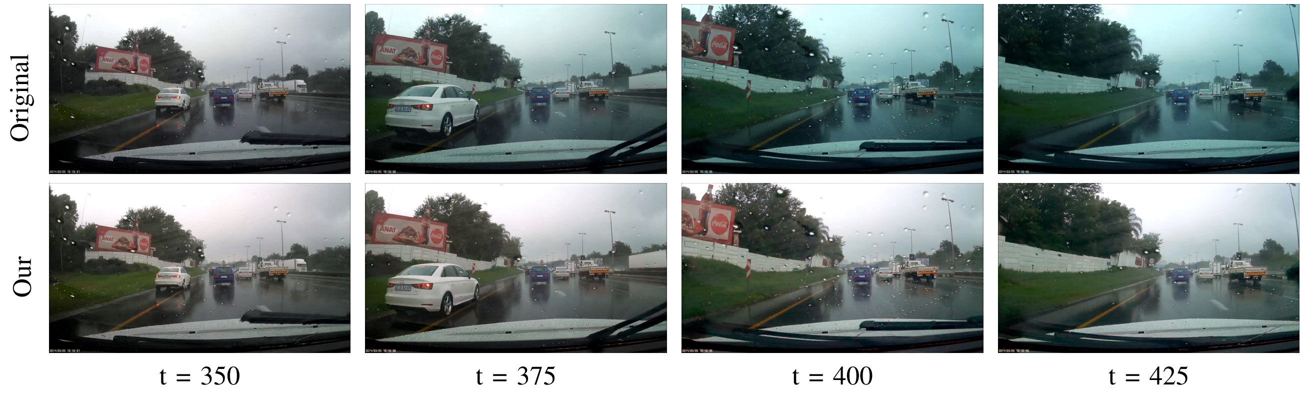

Fig. 4. Tonal stabilization of the sequence "driving". Top row: frames extracted from the original sequence, t = (350, 375, 400, 425). Second row: same frames from top row, after tonal stabilization with our method. This video, taken while driving in a highway, is particularly challenging because of the driving speed, the fast motion of objects, the rain droplets falling and the wiper blade movements. Still, our method produces satisfactory tonal stabilization for this sequence, with no artifact creation.

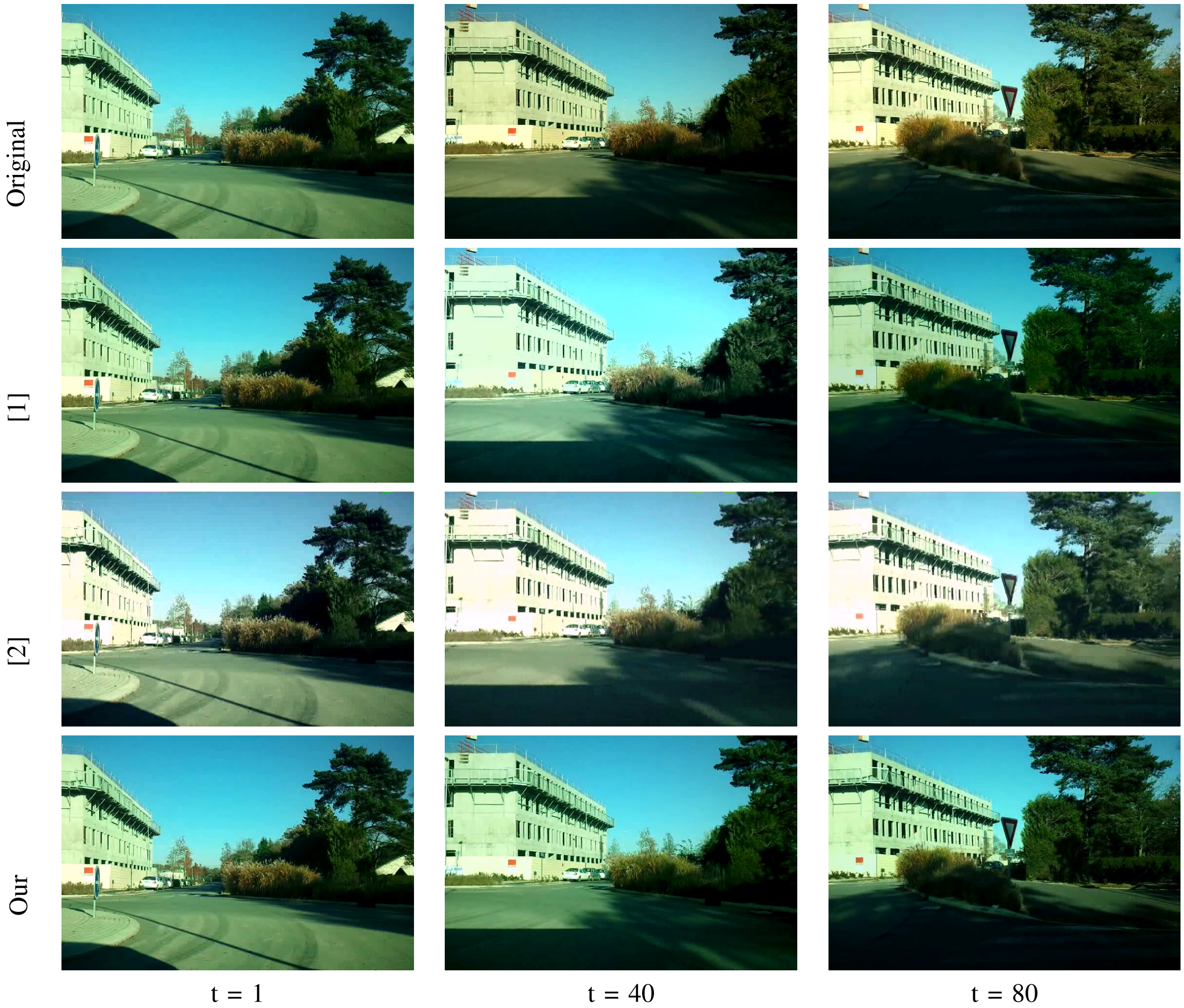

Fig. 5. Comparison of tonal stabilization for the sequence "building" with the methods of [1] and [2]. This figure shows three frames of the sequence (t=1,40,80). Our algorithm is able to stabilize tonal variations without generating artifacts, while the results from [1] are not perfectly stabilized and the results of [2] tend to saturate parts of the building.

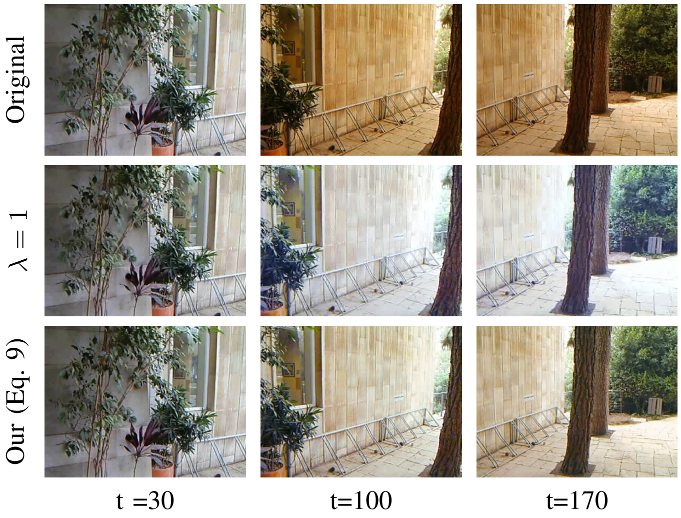

Fig. 7. Illustration of tonal stabilization in sequence "entrance". First row: Frames extracted from the original sequence. Second row: stabilizing with $\lambda = 1$ (no temporal weighting) ensures tonal coherence between all the frames, but at the cost of clipping intensities. Bottom row: Tonal stabilization with temporal weighting ($\lambda$ decreases exponentially in function of motion).

camera configurations as shown in Figure 2. According to our experiments with four different white balance and exposure configurations, summarized in Table I, for two of the shooting configurations the color correction model proposed by [23] is in average more accurate than our model (in terms of PSNR), while for two other test images our model is in average more accurate. In particular, for this set of four images, we observed that [23] performed better when white balance changed between reference and test images, while our power law performed better when exposure changed between reference and test images. However, we note that the number of test images is not large enough to draw a solid conclusion about the two models.

We note that the model proposed by [23] is more complete (with 10 coefficients) and arguably closer to the radiometric calibration pipeline [5] than our model (with 6 coefficients). Nevertheless, we observed in practice that the estimation of optimal coefficients of model [23] is less straightforward and less numerically stable than ours, since a non-convex optimization is solved. In conclusion, we may argue that both models are practical approximations of the physical camera

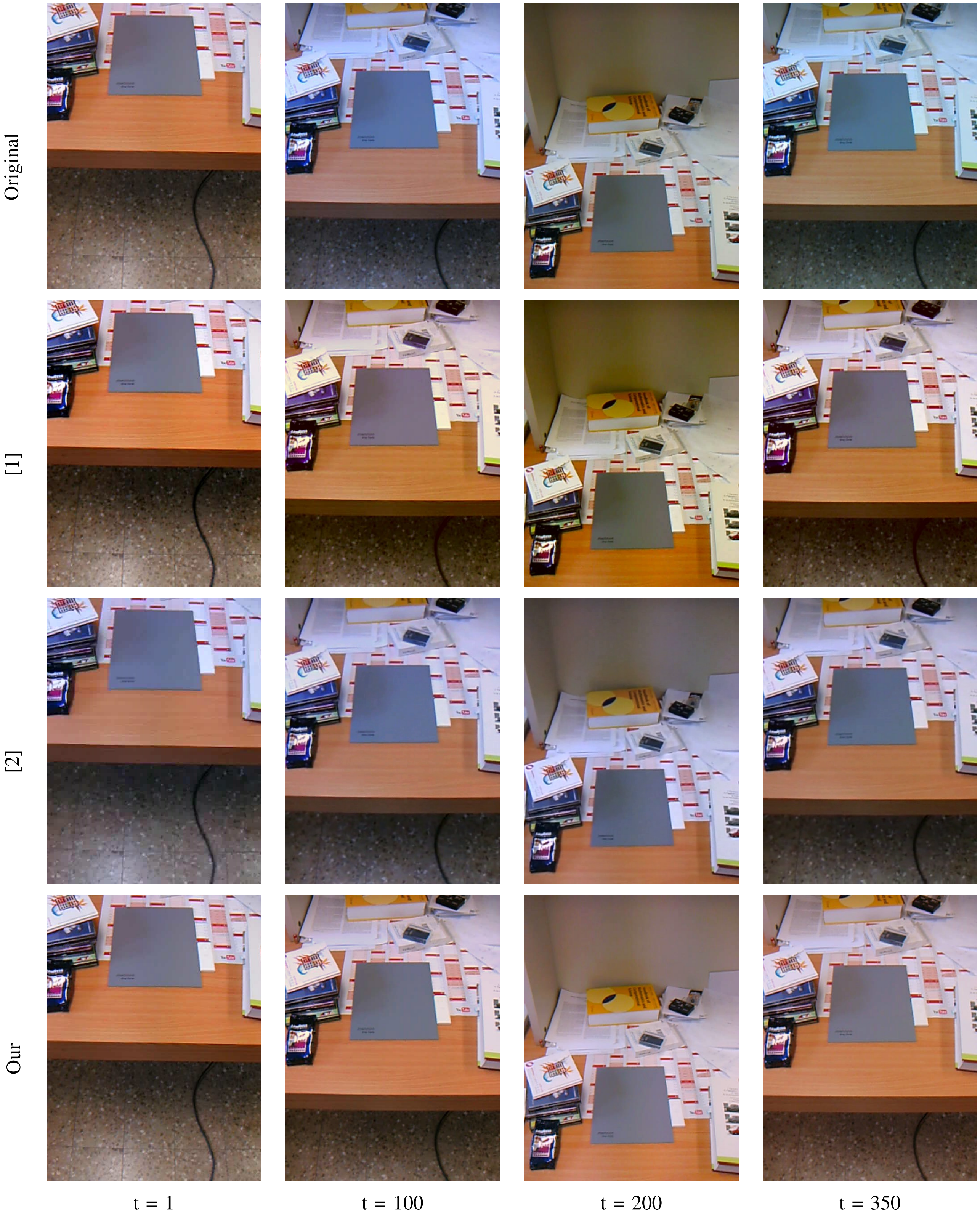

Fig. 6. Tonal stabilization of the sequence “graycard”. First row: frames extracted from the original sequence (t=1, 100, 200, 350). Second row: results from [1]. Notice the yellowish color for t=200. Third row: results from [2] with a wash-out appearance. Bottom row: our stabilized results without any visual artifact.

model, where model proposed by [23] seems to be best targeted for color stabilization among photographies taken with arbitrary cameras, while our model seems to be best targeted for video tonal stabilization, where a sequence of images is typically taken from the same camera and lower computational complexity is an important requirement.

*2) Comparison of video tonal stabilization methods:* In an effort to quantitatively assess the performance of our algorithm we propose to study the tonal variation of a homogeneous patch with respect to the reference (first) frame through the sequence. This is, considering the resulting sequence, we compute the color differences (in CIELAB color space) between a homogeneous patch in the reference frame $P_0$ and its corresponding patches through the resulting sequence $P_t$, $t = 1, \dots, D$. Ideally, the patch tonal variation remains constant and equal to zero. However, this criterion is not sufficient to evaluate a tonal stabilization algorithm. For instance, a resulting sequence of completely homogeneous color frames would satisfy this criterion but would not be a good (pleasant) result. Because of this reason we also study the fidelity to the original sequence by computing the color difference between the same aforementioned patches $P_t$ and the same patches on the original sequence $P_t^o$, for all $t = \{1, \dots, D\}$. A resulting sequence with a large deviation from the original sequence would produce undesired artifacts. With these two criteria being defined we consider a tonal stabilized sequence being a good result when the patch tonal variation is as constant and small as possible and *at the same time* the fidelity to the original sequence is as much preserved as possible.

Obviously, the two error curves (tonal variation and fidelity to original) can be computed, provided the video sequence has a homogeneous patch, as it is the case for the sequence ”building” or the sequence ”greycard”. Fig. 8 shows the error curves for these two sequences. We observe that the patch tonal variation is reduced with our method and the method of [2] when compared to the patch tonal variation of the original sequence but this is not the case for the results of [1]. Also, our method produces the closest results in terms of color fidelity to the original sequence. Notice that for the sequence ”graycard” the fidelity to original is smaller for [2] between t=75 and t=150. We explain this behavior because we choose the first frame as reference. Indeed we obtain a smaller fidelity to original for the first frames (from t=1 to t=75), but then, when there is a big instability of the original sequence for these frames (see red curve of the tonal variation) our algorithm compensates this difference. As we have explained in Fig. 7 we believe that our weighting strategy provides more natural and artifact-free results.

### D. Implementation

Besides the qualitative and quantitative evaluation our method is also more efficient in comparison to the state-of-the-art. Our prototype implementation in Python processes a $1920 \cdot 1080$ resolution video in a rate of 11 frames per second considering image reading and writing and 20 frames per second without reading and writing[6]. On the contrary, the C++ implementation of [2] processes 1 frame per second (depending on video length) and a Python implementation of [1] processes 0.6 frames per second. We believe that an optimized implementation of our method could approach real time processing which is a major advantage and proves the feasibility of embedding robust tonal stabilization algorithms on smart phones.

## IV. CONCLUSION

In this work, we have proposed an efficient tonal stabilization method, aided by global motion estimation and a parametric tonal transformation. We have shown that a simple six-parameters color transformation model is enough to provide tonal stabilization caused by automatic camera parameters, without the need to rely on any *a priori* knowledge about the camera model.

The proposed algorithm is robust for sequences containing motion, it reduces tonal error accumulation by means of long-term tonal propagation, and it does not require high space and time computational complexity to be executed.

One of the main advantages of the proposed method is that it could be applied in practice as an online algorithm, that has potential for real time video processing. The actual unoptimized software implementation is already near real-time processing. In addition, the proposed algorithm is suitable for an implementation on a chip, opening applications such as tonal compensation for video conferences or for live broadcast.

Experiments have demonstrated that the method performs favorably compared to state-of-the-art methods, both in terms of stabilization quality, fidelity to original colors, computational complexity and memory footprint.

## V. ACKNOWLEDGEMENTS

This work has been partially funded by the French Research Agency (ANR) under grant nro ANR-14-CE27-001 (MIRIAM). We thank the authors of [2] for providing their results in our sequences and [1] for helping with implementation details of their method.

## REFERENCES

[1] Z. Farbman and D. Lischinski, “Tonal stabilization of video,” *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2011)*, vol. 30, no. 4, pp. 89:1 – 89:9, 2011.

[2] Y. Wang, D. Tao, X. Li, M. Song, J. Bu, and P. Tan, “Video tonal stabilization via color states smoothing.” *IEEE transactions on image processing*, vol. 23, no. 11, pp. 4838–4849, 2014.

[3] A. Chakrabarti, D. Scharstein, and T. Zickler, “An Empirical Camera Model for Internet Color Vision,” *Procedings of the British Machine Vision Conference 2009*, pp. 51.1–51.11, 2009. [Online]. Available: http://www.bmva.org/bmvc/2009/Papers/Paper364/Paper364.html

[4] H. Lin, S. J. Kim, S. Susstrunk, and M. S. Brown, “Revisiting radiometric calibration for color computer vision,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 129–136.

[5] S. J. Kim, H. T. Lin, Z. Lu, S. Süsstrunk, S. Lin, and M. S. Brown, “A new in-camera imaging model for color computer vision and its application,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 2289–2302, 2012.

[6] M. Grossberg and S. Nayar, “What Can Be Known about the Radiometric Response from Images?” in *Computer Vision - ECCV 2002*, 2002, pp. 189–205. [Online]. Available: http://www.springerlink.com/index/up1wneb7ug3l3wpx.pdf

[6] Processed by Intel(R) Core(TM) i5-3340M CPU @ 2.70GHz, 8GB RAM

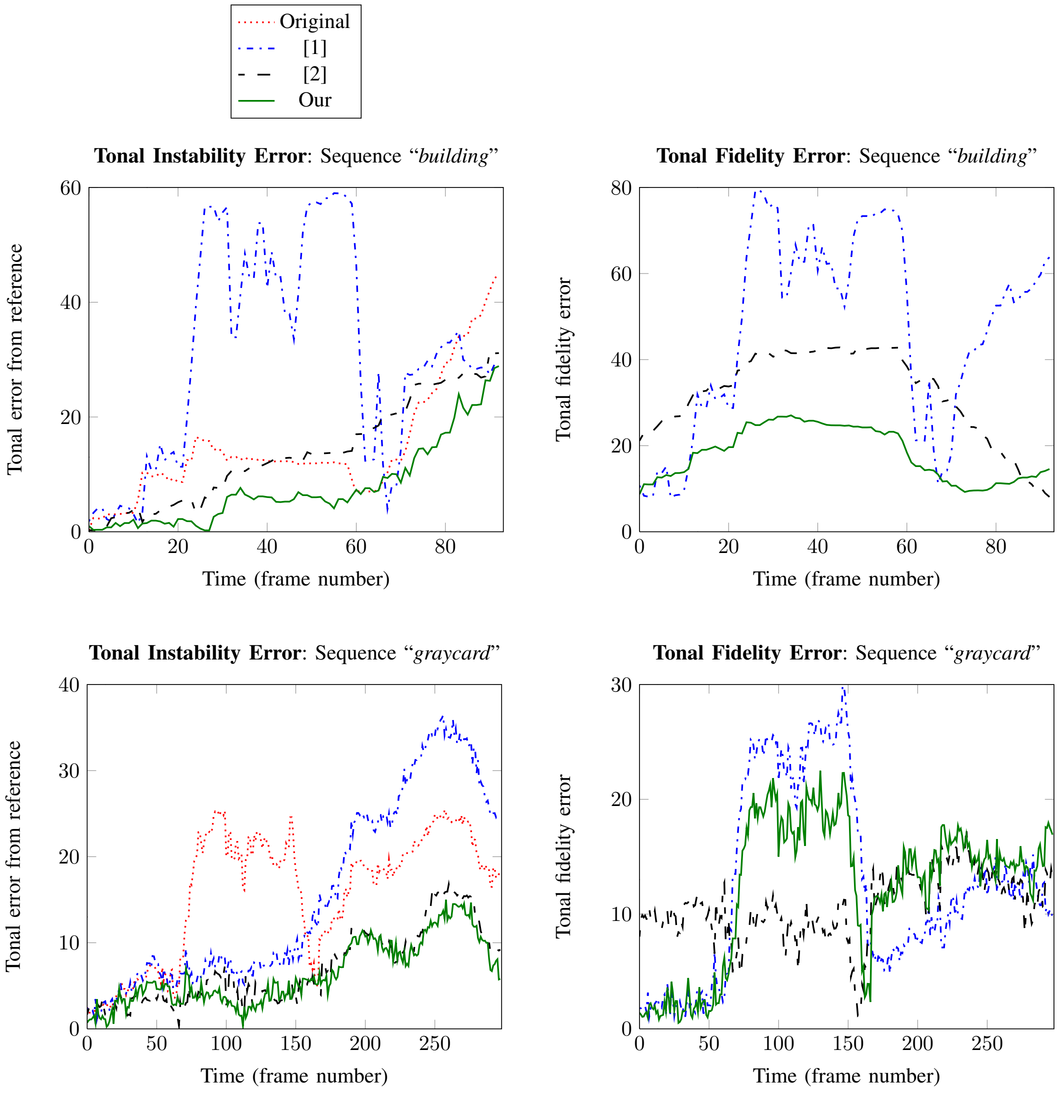

Fig. 8. Quantitative evaluation of the sequence "building" (top) and "graycard" (bottom). For each sequence, we show (i) the tonal instability error, computed as the color distance of a tracked patch to the reference (first) frame, and (ii) the tonal fidelity error, computed as the color distance at each instant, between the corrected frame and the original frame, which indicates the degree of fidelity between the original and the corrected sequence. The color distances are computed as the euclidian distance in perceptual color space CIELAB. Overall, our method compares favorably with the methods of [1] and [2], both in terms of *reduction of tonal instability* as in terms of *fidelity to original colors*.

[7] E. H. Land, J. J. McCann *et al.*, "Lightness and retinex theory," *Journal of the Optical society of America*, vol. 61, no. 1, pp. 1 – 11, 1971.

[8] E. H. Land, "The Retinex Theory of Color Vision The Retinex Theory of Color Vision," vol. 237, no. 6, 1977.

[9] D. H. Brainard and B. A. Wandell, "Analysis of the retinex theory of color vision," *JOSA A*, vol. 3, no. 10, pp. 1651 – 1661, 1986.

[10] S. D. Hordley, "Scene illuminant estimation: Past, present, and future," *Color Research & Application*, vol. 31, no. 4, pp. 303 – 314, 2006.

[11] A. Gijsenij, T. Gevers, and J. V. D. Weijer, "Computational color constancy: Survey and experiments," *Image Processing, IEEE Transactions on*, vol. 20, no. 9, pp. 2475 – 2489, 2011.

[12] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 34 –41, sep/oct 2001.

[13] L. Neumann and A. Neumann, "Color style transfer techniques using hue, lightness and saturation histogram matching," in *Computational Aesthetics in Graphics, Visualization and Imaging 2005*, B. G. W. P. L. Neumann, M. Sbert, Ed., 5 2005, pp. 111–122. [Online]. Available: http://www.cg.tuwien.ac.at/research/publications/2005/Ineumann-2005-cst/

[14] F. Pitié, A. C. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Comput. Vis. Image Underst.*, vol. 107, no. 1-2, pp. 123–137, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.cviu.2006.11.011

[15] N. Papadakis, E. Provenzi, and V. Caselles, "A variational model for histogram transfer of color images," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1682–1695, 2011.

[16] D. Freedman and P. Kisilev, "Object-to-object color transfer: Optimal flows and smsp transformations," in *IEEE Computer Vision and Pattern Recognition*, 2010, pp. 287–294.

[17] S. Ferradans, N. Papadakis, J. Rabin, G. Peyr, and J.-F. Aujol, "Regularized discrete optimal transport," in *Scale Space and Variational Methods in Computer Vision*, ser. Lecture Notes in Computer Science, A. Kuijper, K. Bredies, T. Pock, and H. Bischof, Eds. Springer Berlin Heidelberg, 2013, vol. 7893, pp. 428–439.

[18] N. Murray, S. Skaff, L. Marchesotti, and F. Perronnin, "Toward automatic and flexible concept transfer," *Computers & Graphics*, vol. 36, no. 6, pp. 622–634, 2012.

[19] F. Wu, W. Dong, Y. Kong, X. Mei, J.-C. Paul, and X. Zhang, "Content-based colour transfer," *Computer Graphics Forum*, pp. no–no, 2013. [Online]. Available: http://dx.doi.org/10.1111/cgf.12008

[20] O. Frigo, N. Sabater, V. Demoulin, and P. Hellier, "Optimal trans-

portation for example-guided color transfer," in *Proceedings of Asian Conference on Computer Vision*, 2014.

[21] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski, "Non-rigid dense correspondence with applications for image enhancement," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 70:1–70:10, Jul. 2011. [Online]. Available: http://doi.acm.org/10.1145/2010324.1964965

[22] ——, "Optimizing color consistency in photo collections," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 38:1–38:10, Jul. 2013. [Online]. Available: http://doi.acm.org/10.1145/2461912.2461997

[23] J. Vazquez-Corral and M. Bertalmio, "Color stabilization along time and across shots of the same scene, for one or several cameras of unknown specifications," *Image Processing, IEEE Transactions on*, vol. 23, no. 10, pp. 4564–4575, Oct 2014.

[24] É. Decencière, "Restauration automatique de films anciens," Theses, École Nationale Supérieure des Mines de Paris, Dec. 1997. [Online]. Available: https://pastel.archives-ouvertes.fr/pastel-00003316

[25] J. Delon, "Movie and video scale-time equalization application to flicker reduction," *Image Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 241–248, Jan 2006.

[26] F. Pitié, R. Dahyot, F. Kelly, and A. Kokaram, "A new robust technique for stabilizing brightness fluctuations in image sequences," in *Statistical Methods in Video Processing*. Springer, 2004, pp. 153–164.

[27] J. Delon and A. Desolneux, "Stabilization of flicker-like effects in image sequences through local contrast correction," *SIAM Journal on Imaging Sciences*, vol. 3, no. 4, pp. 703–734, 2010. [Online]. Available: http://dx.doi.org/10.1137/090766371

[28] O. Frigo, N. Sabater, J. Delon, and P. Hellier, "Motion driven tonal stabilization," in *Image Processing (ICIP), 2015 IEEE International Conference on*, Sept 2015, pp. 3372–3376.

[29] Y. Xiong, K. Saenko, T. Darrell, and T. Zickler, "From pixels to physics: Probabilistic color de-rendering," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012, pp. 358–365.

[30] J. Vazquez-Corral and M. Bertalmio, "Simultaneous blind gamma estimation," *IEEE Signal Processing Letters*, vol. 22, no. 9, pp. 1316–1320, Sept 2015.

[31] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quart. J. Appl. Maths.*, vol. II, no. 2, pp. 164–168, 1944.

[32] D. W. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *SIAM Journal on Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[33] J. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of visual communication and image representation*, vol. 6, no. 4, pp. 348–365, 1995.

[34] M. Colom and A. Buades, "Analysis and Extension of the Percentile Method, Estimating a Noise Curve from a Single Image," *Image Processing On Line*, vol. 3, pp. 332–359, 2013.