



HAL
open science

Stabilization of flicker-like effects in image sequences through local contrast correction

Julie Delon, Agnès Desolneux

► **To cite this version:**

Julie Delon, Agnès Desolneux. Stabilization of flicker-like effects in image sequences through local contrast correction. *SIAM Journal on Imaging Sciences*, 2010, 3 (4), pp.703–734. hal-00407796v2

HAL Id: hal-00407796

<https://hal.science/hal-00407796v2>

Submitted on 17 Dec 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

STABILIZATION OF FLICKER-LIKE EFFECTS IN IMAGE SEQUENCES THROUGH LOCAL CONTRAST CORRECTION *

JULIE DELON † AND AGNÈS DESOLNEUX ‡

Abstract. In this paper, we address the problem of the restoration of image sequences which have been affected by local intensity modifications (local contrast changes). Such artifacts can be encountered in particular in biological or archive film sequences, and are usually due to inconsistent exposures or sparse time sampling. In order to reduce such local artifacts, we introduce a local stabilization operator, called LStab, which acts as a time filtering on image patches and relies on a similarity measure which is robust to contrast changes. Thereby, this operator is able to take motion into account without resting on a sophisticated motion estimation procedure. The efficiency of the stabilization is shown on various sequences. The experimental results compares favorably with state of the art approaches.

1. Introduction. In this paper, we are interested in a particular type of artifacts that can be observed in image sequences and which can be described as fast and unnatural intensity fluctuations from one frame to the other. These artifacts are very common in old movies, especially in movies shot before the twenties, and are generally referred to as “flicker”. The main causes of flicker in these movies are support ageing (the negative of the film contains many chemical products which get older and damaged with time) and exposure inconsistency at the acquisition stage (non constant exposure time from one frame to the other [49] - the film was driven manually at this time). Similar effects can appear in more recent analog or digital videos, such as biological or video surveillance sequences. In these cases, these artifacts are mostly due to a sparse time sampling, a non uniform or a non constant illumination of the frames. Although they arise from different physical processes, we will also refer to these artifacts as flicker, or as flicker-like effects. Removing or at least reducing such artifacts in movies is important in order to improve the subjective quality of films, but it can also be a crucial step before any other post-processing (like compression, segmentation, or image comparison for instance).

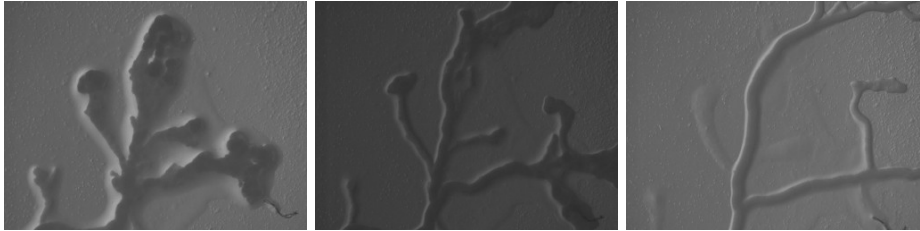
One of the characteristics of flicker-like effects is that they do not create salient geometrical structures in images. In practice, we can almost say that they are invisible on a single frame pulled out of the film, while they cannot be missed in motion, or by comparing several consecutive frames (see Figure 1.1). A consequence of this “transparency” property is that flicker reduction requires the use of successive frames in order to work properly: flicker cannot be removed just by looking at each frame independently of the others (this is not necessarily the case for denoising algorithms for instance). Another characteristic of flicker-like effects is that they can have global and local aspects: depending on the film, they can act as global contrast changes or as several local contrast changes which affect differently various parts of the current frame. In consequence, one of the main difficulties of local flicker reduction is to decide if local changes in the film are due to flicker or to moving objects. This question becomes even more ambiguous when the local effects of flicker have a structured motion, as illustrated by the *taxi* sequence shown in Figure 1.1(c). Now, the main difference between the effects of flicker and moving objects is that the first ones do not modify the salient geometry of the frames. In other words, they can affect the local dynamic of the grey levels, but preserve, at least in first approximation, their local ordering.

1.1. Related works. In this section, we first briefly present different directions that have been explored to modify locally the contrast and dynamic of images or movies. We then focus on

*This work was supported by the French Agence Nationale de la Recherche (ANR), under the grant FREEDOM (ANR07-JCJC-0048-01), “Films, REstauration Et DONnées Manquantes”, and by the GDR ISIS. The authors would like to thank Marc Sandberg, which gave his permission to reproduce here some images from the film *Les Aventures des Pieds Nickelés* (Emile Cohl/Eclair, 1917-1918), Marion Ghibaud and Benoît Ladoux for the *Cells* sequence [18], Marc Durand for the *Physarum* sequence and finally Francois Graner, F. Lense and Y. Bellaïche for the *Drosophila* sequence.

†LTCI, TÉLÉCOM PARIS (CNRS UMR 5141) 46, RUE BARRAULT, 75634 PARIS CEDEX 13 (JULIE.DELON ‘AT’ ENST.FR).

‡MAP5 (CNRS UMR 8145), UNIVERSITÉ PARIS DESCARTES, 45 RUE DES SAINTS-PÈRES, 75006 PARIS (AGNES.DESOLNEUX ‘AT’ MI.PARISDESCARTES.FR).



(a) Three frames from the *Physarum* sequence.



(b) Three frames from *Les Aventures des Pieds Nickelés*, Emile Cohl/Eclair, 1917-1918 (copyright: Marc Sandberg).



(c) Three frames from the *Taxi* sequence, with some synthetic local flicker.

FIG. 1.1. Frames from three sequences containing (a), (b) real flicker or (c) synthetic flicker. The complete sequences are available at http://www.tsi.enst.fr/~delon/Demos/Flicker_stabilization/.

methods developed specifically to remove flicker-like effects in image sequences.

1.1.1. Contrast modification in images and movies. The process of modifying the contrast or color rendering of an image has been thoroughly studied since the seventies. In particular, several methods have been proposed to enhance globally or locally the contrast of images, while preserving their local geometry and sometimes avoiding noise over-enhancement [23, 37, 38, 12, 43, 24, 30, 42, 5]. The aim of these approaches is to make low contrast structures in the image more easily detectable, and this is of particular importance in domains requiring fine image interpretation, as medical image analysis, or satellite imaging. The most well-known of these enhancement methods is Histogram Equalization, in which the grey level distribution of an image is mapped globally onto a uniform distribution. Among the local methods for contrast enhancement, let us mention Adaptive Histogram Equalization [38, 43], and multiscale methods based on wavelets or curvelets decomposition of the image [24, 30, 42, 5]. These approaches aim at preserving edges while increasing the contrast across them. Now, the first rigorous definition of a local contrast change in an image is due to Caselles *et al* [9, 12] and can be summarized as follows. Recall that a global contrast change is merely an increasing function of the grey values. If one denote by Ω the image domain, Caselles *et al* define *local contrast changes* of an image u as the functions $f : \Omega \times \mathbb{R} \rightarrow \mathbb{R}$, increasing in the second variable and preserving *exactly* the topographic map of u (*i.e.* the set of all the connected components of its level lines). In particular, they require that if $u(x) = u(y) = \lambda$ and if x and y are on the same connected component of a level line, then $f(x, \lambda) = f(y, \lambda)$. Relying on this definition, they propose an algorithm [12] to enhance locally the contrast of images thanks to such local contrast changes.

The methods described above deal with a single image, but many approaches have also been developed to deal with contrast changes between several images. For example, image stitching methods [45, 46, 27], which have become very popular in the last ten years, have to adjust inconsistent exposures from one image to the other in order to avoid shifts in brightness or hue in their final mosaics. In [46], these adjustments are block-based and rely on least square estimation. In [27], exposure differences are eliminated by working with gradient images. Another field of applications dealing with contrast modifications is the field of dynamic enhancement for images and videos. Several approaches have been proposed to combine images of the same scene at different exposures and create a high dynamic range map of the scene [14, 4, 32]. The authors of [25] extend these approaches to increase the dynamic of videos where the exposure of each frame is varying rapidly. Although they control completely the exposure of frames during the capture, the registering and tone-mapping they have to apply to increase the dynamic range of the frames are very close to those involved in flicker reduction. Contrast enhancement can also be encountered in papers dealing with very low light-level videos [1, 31], and with the fusion of flash and no flash images [19, 34].

In a related direction, many works have focused on contrast modification and enhancement in medical and biological sequences analysis. Now, we have to distinguish two different types of contrast variations in such sequences: (a) the ones which are considered as artifacts and have to be removed, and (b) the ones which are important to interpret the sequence, as those observed in MRI contrast-enhanced sequences, where a contrast agent is injected to the patient and the differential enhancement of image intensities assists the clinician to detect suspicious regions (see [51, 22, 8] for instance). In the first case, removing flicker-like effects is important to perform other tasks such as denoising [6], thresholding, motion estimation or automatic segmentation. This is the case in the example of Figure 1.1(a). In the second case, one of the aim is to estimate the local contrast enhancement curves (thanks to a parametric model) and to perform at the same time segmentation, registration and abnormalities detection.

1.1.2. Flicker reduction in archived film sequences. Most of the papers dealing with the reduction of flicker-like effects are devoted to the restoration of archived film sequences. These papers can be roughly classified in two categories, those considering these effects as global, and those considering that they affect frames non-uniformly. The earliest papers on flicker reduction [39, 44, 15] model the flicker as a global affine change of contrast on the frames. If we denote with $(u_t^0)_{t \in \mathbb{R}}$ the original (unknown) image sequence and $(u_t)_{t \in \mathbb{R}}$ the observed image sequence, this model can be written $u_t = \alpha_t u_t^0 + \beta_t$, where α_t and β_t are two unknown constants, depending only on t . Unfortunately, this simple affine model fails at capturing the non linear aspects of flicker-like effects. A first non linear regularization approach is proposed in [41]. One limitation of these different restoration schemes is that they act recursively or rely on reference frames, supposed to be undegraded. These schemes are thus oversensitive to initialization and can become unstable (subject to errors accumulation when dealing with long sequences). A more sophisticated model of flicker degradation is developed by Vlachos in [49]. He shows in particular that the flicker degradations due to exposure inconsistencies of photographic films are highly non linear and well modeled by contrast changes of the form $Id + \psi$, where ψ is a cubic polynomial. Despite an accurate modeling, the proposed correction algorithm is also limited by its recursive aspect. In order to avoid this shortcoming, following papers have favored methods not depending on the time origin and direction [33, 17]. The idea is to define a time dependent grey level distribution as a midway of all grey level distributions of the frames u_s for s in $[t - \delta t, t + \delta t]$, and to fit the contrast of u_t on this distribution. The authors of [33] define the midway distribution at time t as a direct average of all grey level distributions in $[t - \delta t, t + \delta t]$. The main drawback of this approach is that direct histogram averaging can lead to severe artifacts in the frames, as its is shown in [17]. In order to overcome this difficulty, Delon shows in [17] how to average correctly a set of histograms, and derives a regularization operator, called *Scale Time Equalization*, from a sequence of axioms. Roughly speaking, it is shown that the midway distribution at time t should be defined as the *harmonic mean* of all frame distributions in $[t - \delta t, t + \delta t]$.

Global methods are attractive for their simplicity and their robustness to motion and to the

presence of small defects (dirt, scratches). However, in practice, the effects of flicker can vary within an image, and the zones which are the most affected can change from a frame to the other (see Figure 1.1(c)). In the literature, one of the most popular model of local flicker degradation involves locally affine contrast changes [47, 40, 50] and can be written $u_t(x) = a_t(x)u_t^0(x) + b_t(x)$, $\forall x \in \Omega$, where a_t and b_t are smooth functions of the space variable x . This model can be generalized to non linear degradations [35, 36, 21, 20] by the formula $u_t(x) = f_t(x, u_t^0(x))$, where $f_t(x, \cdot)$ is increasing and $f_t(\cdot, \lambda)$ is assumed to be smooth. Several methods have been proposed to correct these local degradations. A detailed survey of existing approaches can be found in the recent paper [20]. In most of these approaches, the image domain is divided into a fixed grid of overlapping blocks. A correction is then computed and applied on each block, using corresponding blocks in other frames and eventually some weighted interpolation between blocks to avoid blocking boundaries artifacts. The main differences between these works lie in the way these corresponding blocks are chosen and in the way the flicker is compensated. Several authors [47, 40, 50] propose a recursive affine correction, the blocks of the frame $t-1$ being used to correct the blocks of the frame t . In [47, 40], motion estimation is avoided by making use of robust estimators or of rejection criteria (to detect blocks containing motion or blotches and discard them in flicker estimation). Whereas the authors of [50] make use of an explicit block-matching motion estimation. Unfortunately, the recursive aspect of these methods makes them unstable and sensitive to initialization. An alternative approach to local flicker stabilization is to use all frames in a sliding temporal window $[t - \delta t, t + \delta t]$, and reliability weights to compensate local flicker degradations between frames [35, 36, 21, 20]. In [35], robust estimators are used to estimate the flicker parameters while discarding outliers (blotches and motion) from this estimation. In [36], a real parameter $\gamma(x)$, varying slowly in space, controls the intensity of flicker around each pixel x . The estimation of γ should theoretically involve motion estimation. However, considering that “the motion estimation is deteriorated by the presence of flicker” [36], the authors replace this estimation by a temporal filtering of γ . They propose in an optional second stage to perform a conventional motion estimation on the deflickered sequence, and then use it to register the frames before estimating again the flicker parameter γ . More recently, the authors of [21, 20] directly use motion-compensated sequences to estimate a parametric intensity error profile between frames, generalizing the approach of [49] to local flicker degradations.

1.1.3. Motion estimation in presence of local contrast changes. The question of motion estimation is probably the main difficulty in movie and video restoration, in particular in the restoration of flicker-like effects. Indeed, as mentioned before, it is sometimes especially difficult to distinguish contrast irregularities due to flicker from those due to moving objects, or, in the case of old movies, from those due to other impairments (blotches, scratches, etc.). In the literature on flicker reduction, some authors choose [47, 40, 35] to not estimate motion, but to consider it as an outlier: in these works, motion is assumed to be well handled by rejection criteria or a robust estimation of flicker parameters. Obviously, if the sequence contains a dominant motion or if several important objects are moving, these approaches turn out to be inadequate.

Now, if one want to estimate the motion flow in presence of flicker, it is important to propose similarity measures which are robust to contrast changes. This is the case of [21, 20], where the authors compensate motion thanks to the algorithm of Black and Anandan [3], in which the usual “brightness constancy constraint” is relaxed by using robust similarity measures. More recently, several approaches have been proposed to precisely estimate motion while remaining invariant or robust to contrast [10, 26, 11], for instance by relying on gradient orientation instead of grey levels. In this paper, we propose to follow another direction. We will show in Section 3.1 that a very simple similarity measure between image patches, built to be robust to contrast changes, permits to take motion directly into account in the flicker stabilization process and to obtain satisfying results, without necessarily estimating this motion very accurately.

1.2. Contributions and outline of the paper. The goal of this paper is to propose a generic method for correcting radiometric unsteadiness in image sequences. The method should apply to all local radiometric degradations that can be described in first approximation by local contrast changes, whatever their physical causes. In Section 2, we give some notations and recall

the definition of the Scale-Time-Equalization introduced in [17]. Next, in Section 3, we present a new stabilization operator for flicker-like effects, called LStab, which acts as a time filtering on image patches. This operator relies on a similarity measure between image patches which is robust to contrast changes, and is able to take motion into account without relying on a sophisticated motion estimation procedure including regularization terms. The corresponding algorithm does not require any pre- or post- treatment, is easy to reproduce and to implement, does not rely on any assumption on the origin of the flicker and does not require “external procedures” like choosing a reference frame. In Section 4, the ability of this operator to deal with different kinds of local contrast degradations is demonstrated on several sequences: sequences with synthetic local flickers, biological sequences and several extracts from old movies with strong real flicker. The whole stabilization scheme is also confronted with state of the art flicker removal procedures on two sequences, and seems to provide similar or better results than the ones presented in [36, 20]. The stability of the procedure in sequences without flicker is also investigated.

2. Scale Time Equalization. The goal of this section is to give a reminder on the mid-way equalization of two images and on its extension to image sequences, namely the Scale-Time Equalization.

2.1. Some notations. We begin here with the notations and vocabulary that will be used throughout this paper. The damaged film is denoted by $u = (u_t(\cdot))$, where u_t is the frame at time t . The image domain, denoted by Ω , is generally a rectangle of \mathbb{R}^2 or of \mathbb{Z}^2 (in the case of discrete images). We only consider “black and white” movies, which means that each image u_t is a function from Ω to \mathbb{R} . The original, non-observed and unknown film, is denoted by u^0 .

As mentioned before, applying a *contrast change* to an image v means changing v into $g(v)$, where g is an increasing function from \mathbb{R} into itself. We will call *affine contrast change* any contrast change of the form $g(\lambda) = a\lambda + b$ with $a, b \in \mathbb{R}$ and $a > 0$.

The *upper and lower level sets* of an image $v : \Omega \mapsto \mathbb{R}$ are respectively defined as the sets

$$\chi_\lambda(v) = \{x \in \Omega; v(x) > \lambda\} \text{ and } \chi^\lambda(v) = \{x \in \Omega; v(x) \leq \lambda\},$$

where $\lambda \in \mathbb{R}$. If g is a contrast change, then $\chi_{g(\lambda)}(g(v)) = \chi_\lambda(v)$, which means that the set of all upper level sets (resp. the set of all lower level sets) remains the same under contrast changes. As a consequence, the *topographic map* of an image [9, 13], *i.e.* the set of connected components of the topological boundaries of its level sets, remains unchanged under contrast changes. This means that the geometrical content of an image is not modified by contrast changes, as we can see on Figure 2.1.



FIG. 2.1. On the left, two images containing the same geometric content. Each image can be obtained by applying a well chosen contrast change to the other. On the right, some level lines of these images. The small differences that can be observed between level lines are due to quantization artifacts.

An image u_t of the film being given, we write $\chi_\lambda(t)$ and $\chi^\lambda(t)$ for the upper and lower level sets of u_t . The histogram of u_t , *i.e.* the distribution of its grey levels, is denoted by h_t . The

corresponding cumulative distribution function (or cumulative histogram) is denoted by H_t . If u_t is defined on a discrete grid Ω with $|\Omega| = N^2$ and takes its values in $\{0, \dots, L\}$, the cumulative histogram $H_t : \mathbb{R} \rightarrow \{0, \dots, N^2\}$ is given by:

$$H_t(\lambda) = |\{x \in \Omega | u_t(x) \leq \lambda\}| = \sum_{\mu=0}^{\lambda} h_t(\mu),$$

where $|\cdot|$ denotes the cardinality of a set.

This definition leads to the following interpretations:

- For any $x \in \Omega$, $H_t(u_t(x))$ is the *rank* of x , when the grey levels of u_t are ordered increasingly.
- Let k be an integer, we define $H_t^{-1}(k) = \min\{\lambda \in \{0, \dots, L\} ; H_t(\lambda) \geq k\}$. If x is a pixel of rank k in u_t , then $H_t^{-1}(k)$ is the grey level of x .
- Let G be an increasing discrete function on $\{0, \dots, L\}$, taking its values in $\{0, \dots, N^2\}$. Let us define $G^{-1}(k) = \min\{\lambda \in \{0, \dots, L\} ; G(\lambda) \geq k\}$. If we assume that H_t is onto, then $v_t = G^{-1} \circ H_t(u_t)$ and u_t share the same geometry, and the cumulative histogram of v_t is given by the function G . In practice, the hypothesis that H_t is onto is almost never satisfied, and the cumulative histogram of $G^{-1} \circ H_t(u_t)$ will only be very close to G . Roughly speaking, this means that we can map the grey level distribution of an image onto the grey level distribution of any other image by an adequate contrast change.

2.2. Midway equalization between two images. In this section, we recall the definition of the midway equalization between two images. Let u_1 and u_2 be two discrete images defined on a grid Ω containing $|\Omega| = N^2$ pixels and taking their values in $\{0, \dots, L\}$. Let h_1 and h_2 denote their respective grey level histograms. In order to average the contrast between u_1 and u_2 (*i.e.* to equalize their respective grey level histograms on a common one), one could just apply a contrast change to each image such that the grey level distribution of both images would fit $(h_1 + h_2)/2$. This solution is generally not satisfactory. Indeed, if u_1 (resp. u_2) has a unimodal grey level histogram, centered at m_1 (resp. $m_2 \neq m_1$), the average histogram $(h_1 + h_2)/2$ contains two modes: one at m_1 and the other at m_2 . It would be much more natural to define a midway histogram between them as a unimodal histogram centered at $(m_1 + m_2)/2$.

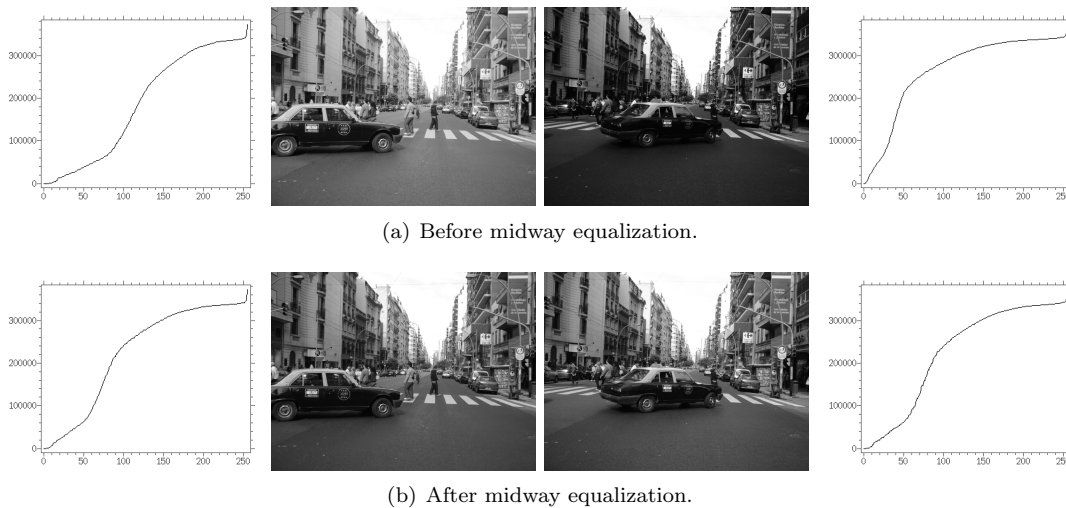


FIG. 2.2. *Top: two images and their respective cumulative histograms. Bottom: the images and their cumulative histograms after the midway equalization.*

As it was shown in [16], the “good” *midway* histogram between h_1 and h_2 can be defined as

the harmonic mean between the cumulative histograms H_1 and H_2 ,

$$H_{1/2} = \left(\frac{H_1^{-1} + H_2^{-1}}{2} \right)^{-1}.$$

The properties of this midway distribution are the following:

- the images $H_{1/2}^{-1} \circ H_1(u_1)$ and $H_{1/2}^{-1} \circ H_2(u_2)$ have the same cumulative histogram $H_{1/2}$.
- if there exists an image u_0 and two contrast changes f and g such that $u_1 = f(u_0)$ and $u_2 = g(u_0)$, then $H_{1/2}$ is the cumulative histogram of $(u_1 + u_2)/2$.
- action on level sets: grey levels having the same rank in both images are averaged together.

Remark 1: In practice, in order to equalize the histograms of two images, one can for instance create two vectors of length N^2 containing respectively the ordered grey levels of u_1 and u_2 , then compute the mean of these two vectors, and finally assign to each pixel of each image the grey level of the element in the averaged vector which has the same rank as the considered pixel.

Remark 2: If we consider h_1 and h_2 as densities of two distributions ν_1 and ν_2 , then in the theory of optimal transportation, the Wasserstein distance between ν_1 and ν_2 is defined as $\int |H_1^{-1} - H_2^{-1}|$ (see [48]). Now, consider the geodesic between ν_1 and ν_2 for the previous distance and define $\nu_{1/2}$ as the middle of this geodesic. Then $H_{1/2}$ is the cumulative distribution function of $\nu_{1/2}$.

2.3. A scale-time correction for global flicker. An extension of the midway equalization to image sequences, called Scale-Time Equalization (STE), can be used to stabilize the flicker globally. In [17], Delon has shown the following proposition, which gives the canonical form of a flicker reduction operator when some simple hypotheses are assumed.

PROPOSITION 2.1. [17] *Let (u_t) be a film and (H_t) the discrete cumulative histograms of the frames. Let STE be an operator acting on films and satisfying the following properties:*

- STE acts like a contrast change on each frame,
- for any λ , the action of STE on the lower level sets $\chi_\lambda(t)$ of the frames does not depend on the action of STE on the upper level sets $\chi^\lambda(t)$,
- STE acts on spatially constant films (i.e. films such that each frame is a constant image, but the constant can vary in time) like a linear scale space (that is like a convolution with a Gaussian kernel, see [28]).

Then, there is a scale parameter σ such that the operator STE fits the histogram of u_t on the “midway” histogram \mathbb{H}_t whose inverse is defined by

$$\mathbb{H}_t^{-1}(k) = \int \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(t-s)^2/2\sigma^2} H_s^{-1}(k) ds. \quad (2.1)$$

The corresponding operator on u can be written

$$\text{STE}[u_t](x) = \frac{1}{\sqrt{2\pi\sigma^2}} \int e^{-(t-s)^2/2\sigma^2} H_s^{-1} \circ \underbrace{H_t(u_t(x))}_{\substack{\text{rank of } x \text{ in } u_t \\ \text{grey level of the pixel of same rank as } x \text{ in } u_s}} ds. \quad (2.2)$$

Observe that this correction does not depend of the origin of the film: the operator commutes with translations and symmetries in time. Let us also mention that the operator STE is very similar to the stabilization procedures that have been proposed independently by Pitie et. al in [35] and by Forbin and Vlachos in [21]. However, the main property here is that the derivation of the STE operator has been obtained following an axiomatic point of view: as it is stated by Proposition 2.1, when some simple axioms are required for a global flicker stabilization operator, then STE becomes the unique possible solution.

3. Motion based flicker reduction without prior estimation. As explained in the introduction, the Scale Time Equalization is easy to perform and gives good results as long as the flicker acts globally on a sequence. However, such a global method cannot handle local flicker

effects as the ones observed in the examples of Figure 1.1. In this section, we define a local, motion-based, flicker stabilization operator, called LStab. This operator relies on a similarity measure between image patches, defined in Section 3.1.

3.1. A similarity measure robust to contrast changes. Usually, when one wants to estimate motion in a movie, one computes the pixel correspondences based on the assumption that grey levels are preserved (this is the usual “grey level constancy constraint” along trajectories). Unfortunately, when the movie is damaged by local contrast changes, the grey level of a given object in the movie can vary in time. We mentioned in the introduction several approaches [3, 10, 26, 11] that have been proposed to estimate motion in such drastic conditions, using robust or invariant similarity measures. In this paper, we also propose to rely on a similarity measure D between patches (small squares of pixels) built to be robust to contrast changes. This measure will at the same time provide an estimation of motion (not necessarily highly accurate) and be used to weight the different terms in the stabilization operator presented in Section 3.2. For these reasons, the measure D of similarity between two image patches I and J should satisfy the following properties :

- (i) Symmetry : $D(I, J) = D(J, I)$,
 - (ii) Robustness to affine and increasing contrast changes: $\forall a > 0$ and $\forall b$, $D(I, aI + b)$ should be equal to 0,
 - (iii) Ability to discriminate: we want $D(I, J)$ to be “large” when J doesn’t look like any $aI + b$.
- In the following, we will consider three different similarity measures satisfying all the first and second properties above, and compare them in relation to the third property. These similarity measures are :

- D_{angle} : the *angular measure* is the L^2 distance between gradient orientations. For an image I , let $x \rightarrow \theta_I(x) = \text{Arg}(\nabla I(x)) \in [0, 2\pi)$ denote the orientation of the gradient of I at any pixel x . These orientations are invariant to any change of contrast (indeed, if $g : \mathbb{R} \rightarrow \mathbb{R}$ is increasing, then $\nabla g(I) = g'(I)\nabla I$, and thus $\theta_{g(I)} = \theta_I$). We then define the “angular measure” between two image patches I and J (defined on the same domain Λ) by

$$D_{\text{angle}}(I, J)^2 = \| e^{i\theta_I} - e^{i\theta_J} \|_2^2 = \frac{1}{|\Lambda|} \sum_{x \in \Lambda} \| e^{i\theta_I(x)} - e^{i\theta_J(x)} \|^2 .$$

Such a distance has been used for instance by Lisani and Morel in [29] to detect changes in satellite images or by Caselles *et al* in [10, 11] for motion estimation. This distance has the property of being invariant under any increasing change of contrast on I or J .

- D_{corr} : the *correlation measure* is the L^2 distance between image patches normalized by their empirical mean and variance. Let $\bar{I} = \sum_{x \in \Lambda} I(x)/|\Lambda|$ (resp. \bar{J}) denote the empirical mean grey level of I (resp. of J) on the domain Λ , $S_I^2 = \sum_{x \in \Lambda} (I(x) - \bar{I})^2/|\Lambda|$ (resp. S_J^2) denote the empirical variance and $\text{Cov}(I, J) = \sum_{x \in \Lambda} (I(x) - \bar{I})(J(x) - \bar{J})/|\Lambda|$ denote the empirical covariance between I and J . Then the correlation measure is given by

$$D_{\text{corr}}(I, J)^2 = \frac{1}{2} \left\| \frac{I - \bar{I}}{S_I} - \frac{J - \bar{J}}{S_J} \right\|_2^2 = 1 - \text{Corr}(I, J) = 1 - \frac{\text{Cov}(I, J)}{S_I S_J} ,$$

- D_{aff} : the “*affine*” *similarity measure* is defined as

$$D_{\text{aff}}(I, J) = \max \left(\min_{a \geq 0, b} \| I - aJ - b \|_2, \min_{a \geq 0, b} \| J - aI - b \|_2 \right) .$$

D_{aff} can be exactly computed by the formula:

$$D_{\text{aff}}(I, J)^2 = \max(S_I^2, S_J^2) \times \min(1, 1 - \text{Corr}(I, J)|\text{Corr}(I, J)|) ,$$

where S_I^2 , S_J^2 and $\text{Corr}(I, J)$ are defined as above in the case of the correlation measure.

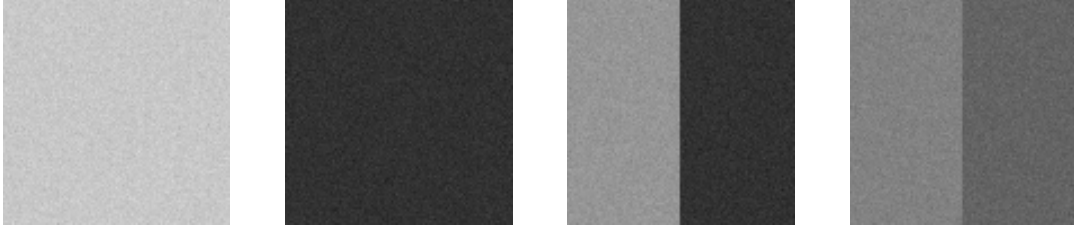


FIG. 3.1. Four patches, from left to right: I_1 , I_2 , J_1 and J_2 . The first two patches I_1 and I_2 both represent a constant zone up to a contrast change and an additive noise. The last two patches J_1 and J_2 both represent the same edge zone up to a contrast change and an additive noise.

Remark 1: We have used here the word “similarity measure” instead of “distance” because the usual properties of the mathematical definition of a distance are not satisfied (mainly the fact that we can have $D(I, J) = 0$ without having $I = J$).

Remark 2: The measure D_{aff} could be generalized to

$$D_{\text{monotone}}(I, J) = \max \left(\min_{g \text{ increasing}} \|I - g(J)\|, \min_{g \text{ increasing}} \|g(I) - J\| \right),$$

which can be easily computed via the *pool adjacent violators* algorithm [2]. For the sake of simplicity, we stick in this paper with the affine formulation.

As explained at the beginning of the section, an “ideal” similarity measure should be at the same time invariant to local contrast changes and discriminant. However, these two goals are somewhat contradictory: the more invariance a similarity measure has, the less important its discrimination power tends to be. To find the optimal measure, a good trade-off should be found between the two criteria. Now, the three measures defined above, D_{angle} , D_{corr} and D_{aff} , are all robust or invariant to affine contrast changes. In the following proposition, we study their discriminative power when comparing two image patches with the same geometry (two constant zones, or two edge zones with a change of contrast), or two patches with dissimilar geometries (a flat zone and an edge zone). Such patches are shown on Figure 3.1.

PROPOSITION 3.1 (Choice of a similarity measure). *Let Λ denote the domain on which the image patches are defined: it is assumed to be a square of $n \times n$ pixels. Let I_1 (resp. I_2) be a random image patch, whose pixel grey levels are i.i.d. random variables with the normal distribution $\mathcal{N}(c_1, \sigma^2)$ (resp. $\mathcal{N}(c_2, \sigma^2)$), where c_1 , c_2 and σ are constants. I_1 and I_2 can be seen as noisy constant patches. Now, let J_1 (resp. J_2) be a random image patch, whose pixel grey levels are independent, and such that for all $1 \leq k \leq n/2$, $J_1(k, l)$ (resp. $J_2(k, l)$) follows the $\mathcal{N}(\alpha_1, \sigma^2)$ (resp. $\mathcal{N}(\alpha_2, \sigma^2)$) distribution, and for all $n/2 < k \leq n$, $J_1(k, l)$ (resp. $J_2(k, l)$) follows the $\mathcal{N}(\beta_1, \sigma^2)$ (resp. $\mathcal{N}(\beta_2, \sigma^2)$) distribution, where α_1 , α_2 , β_1 and β_2 are constants. J_1 and J_2 can be seen as noisy edge patches. Then the asymptotic behavior (in the sense of the a.s. convergence when $n \rightarrow \infty$) of the three similarity measures defined above is summarized in the following table:*

	$D^2(I_1, I_2)$	$D^2(I_1, J_1)$	$D^2(J_1, J_2)$
D_{angle}^2	$2 (\pm \frac{\sqrt{2}}{n})$	$2 (\pm \frac{\sqrt{2}}{n})$	$2 - \frac{2\mu(C_1/\sigma, C_2/\sigma)}{C_1^n C_2}$
D_{corr}^2	1	1	$1 - \frac{C_1^n C_2}{\sqrt{C_1^2 + 4\sigma^2} \sqrt{C_2^2 + 4\sigma^2}}$
D_{aff}^2	σ^2	$\frac{C_1^2}{4} + \sigma^2$	$\left(\sigma^2 + \max\left(\frac{C_1^2}{4}, \frac{C_2^2}{4}\right) \right) \cdot \min\left(1, 1 - \frac{C_1 C_2 C_1 C_2 }{(4\sigma^2 + C_1^2)(4\sigma^2 + C_2^2)}\right)$

where $C_1 = \beta_1 - \alpha_1$ (resp. $C_2 = \beta_2 - \alpha_2$) is the contrast of the edge in J_1 (resp. J_2), and $\mu(C_1/\sigma, C_2/\sigma)$ is a function of C_1/σ and C_2/σ which takes its values between -1 and 1 .

Proof.

• We first start with the angular measure D_{angle} . We will assume here that the gradient of any discrete image I is computed by simple differences on a four pixels neighborhood, that is:

$$\nabla I(k, l) = \frac{1}{2} \begin{pmatrix} I(k+1, l) + I(k+1, l+1) - I(k, l) - I(k, l+1) \\ I(k, l+1) + I(k+1, l+1) - I(k, l) - I(k+1, l) \end{pmatrix}.$$

Using complex numbers notations, we write

$$2\nabla I(k, l) = (1 + i)(I(k + 1, l + 1) - I(k, l) + i(I(k, l + 1) - I(k + 1, l))).$$

Now, observe that if X_1, X_2, X_3 and X_4 are four i.i.d. $\mathcal{N}(0, 1)$ random variables, then $X_4 - X_1 + i(X_3 - X_2) = Re^{i\Theta}$, where Θ is uniform on $[0, 2\pi)$ and R has the probability density $r \rightarrow \frac{1}{2}r \exp(-r^2/4)$ on \mathbb{R}_+ . Thus, if I is a noisy constant image patch (like I_1 or I_2), the orientations $\theta_I(k, l)$ are uniformly distributed on $[0, 2\pi)$. If I is a noisy edge patch (like J_1 or J_2), the orientations $\theta_I(k, l)$ are uniform on $[0, 2\pi)$ when the point (k, l) does not belong to the edge (i.e. when $k \neq n/2$). When the pixel (k, l) belongs to the edge (i.e. when $k = n/2$), then $\theta_I(k, l) = \text{Arg}(C + i\sigma Re^{i\Theta}) = \text{Arg}(\frac{C}{\sigma} + iRe^{i\Theta})$, where C is the contrast of the edge and where R and Θ follow the laws given above.

Let us now compute the angular measure D_{angle} between two generic image patches I and J : $D_{\text{angle}}^2(I, J) = \frac{1}{n^2} \sum_{k,l} |e^{i\theta_I(k,l)} - e^{i\theta_J(k,l)}|^2 = 2 - \frac{2}{n^2} \sum_{k,l} \cos(\theta_I(k, l) - \theta_J(k, l))$. If Θ is a random variable uniformly distributed on $[0, 2\pi)$, then $\mathbb{E}(\cos \Theta) = \frac{1}{2\pi} \int_0^{2\pi} \cos \theta d\theta = 0$ and $\text{Var}(\cos \Theta) = \frac{1}{2\pi} \int_0^{2\pi} \cos^2 \theta d\theta = \frac{1}{2}$. Now, if I_1 and I_2 are two noisy constant image patches, $\theta_{I_1}(k, l) - \theta_{I_2}(k, l)$ is uniformly distributed on $[0, 2\pi)$. It follows that the expected value of their squared angular distance is $\mathbb{E}(D_{\text{angle}}^2(I_1, I_2)) = 2$, and its variance is of the order of $\frac{4}{n^4} n^2 \text{Var}(\cos \Theta) = 2/n^2$. If I_1 is a noisy constant patch and J_1 is a noisy edge patch, then $\theta_{I_1}(k, l) - \theta_{J_1}(k, l)$ is also uniformly distributed on $[0, 2\pi)$, thanks to the fact that θ_{I_1} is uniformly distributed on $[0, 2\pi)$ and independent of θ_{J_1} . Consequently, their angular similarity measure has exactly the same property as $D_{\text{angle}}(I_1, I_2)$. Finally, assume that J_1 and J_2 are two noisy edge patches, with respective contrasts C_1 and C_2 across their edges. Consider the two random variables $\Phi_1 = \text{Arg}(C_1/\sigma + iR_1 e^{i\Theta_1})$ and $\Phi_2 = \text{Arg}(C_2/\sigma + iR_2 e^{i\Theta_2})$, where R_1, R_2, Θ_1 and Θ_2 are independent random variables following the laws defined above. Let us denote $\mu(C_1/\sigma, C_2/\sigma) = \mathbb{E}(\cos(\Phi_1 - \Phi_2))$. The computation of this function μ is not straightforward: it involves special Bessel type functions. However, notice that when the normalized contrasts C_1/σ and C_2/σ increase, the laws of Φ_1 and Φ_2 get more concentrated around 0 and μ gets closer to 1. Finally, to compute the expected angular similarity measure between J_1 and J_2 , we just write

$$\mathbb{E}(D_{\text{angle}}^2(J_1, J_2)) = 2 - \frac{2}{n} \mu(C_1/\sigma, C_2/\sigma)$$

- The computation of the two other similarity measures D_{corr} and D_{aff} is simpler. In the following, the $\rightarrow_{a.s.}$ sign denotes the almost sure convergence when $n \rightarrow \infty$. Observe that the image patch I_1 can be written $I_1(k, l) = c_1 + \sigma X_{k,l}$, where the $X_{k,l}$'s are i.i.d. with the $\mathcal{N}(0, 1)$ distribution. In the same way, I_2 can be written $I_2(k, l) = c_2 + \sigma Y_{k,l}$, where the $Y_{k,l}$'s are i.i.d. with the $\mathcal{N}(0, 1)$ distribution. It follows that

$$\bar{I}_1 = \frac{1}{n^2} \sum_{k,l} (c_1 + \sigma X_{k,l}) = c_1 + \sigma \bar{X} \rightarrow_{a.s.} c_1, \quad S_{I_1}^2 = \sigma^2 S_X^2 \rightarrow_{a.s.} \sigma^2,$$

$$\text{and } \text{Cov}(I_1, I_2) = \frac{\sigma^2}{n^2} \sum_{k,l} (X_{k,l} - \bar{X})(Y_{k,l} - \bar{Y}) \rightarrow_{a.s.} 0.$$

Thus, when $n \rightarrow \infty$, $D_{\text{corr}}^2(I_1, I_2) \rightarrow_{a.s.} 1$ and $D_{\text{aff}}^2(I_1, I_2) \rightarrow_{a.s.} \sigma^2$.

For the patch J_1 , it can be written $J_1(k, l) = \alpha_1 + \sigma Z_{k,l}$ for $1 \leq k \leq n/2$ and $J_1(k, l) = \beta_1 + \sigma Z_{k,l}$ for $n/2 < k \leq n$, where the $Z_{k,l}$'s are i.i.d. with the $\mathcal{N}(0, 1)$ distribution. As a consequence,

$$\bar{J}_1 = \frac{1}{n^2} \sum_{(k,l); k \leq n/2} (\alpha_1 + \sigma Z_{k,l}) + \frac{1}{n^2} \sum_{(k,l); k > n/2} (\beta_1 + \sigma Z_{k,l}) = \frac{\alpha_1 + \beta_1}{2} + \sigma \bar{Z} \rightarrow_{a.s.} \frac{\alpha_1 + \beta_1}{2}, \text{ and}$$

$$\begin{aligned}
S_{J_1}^2 &= \frac{1}{n^2} \left(\sum_{(k,l); k \leq n/2} \left(\frac{\alpha_1 - \beta_1}{2} + \sigma(Z_{k,l} - \bar{Z}) \right)^2 + \sum_{(k,l); k > n/2} \left(\frac{\beta_1 - \alpha_1}{2} + \sigma(Z_{k,l} - \bar{Z}) \right)^2 \right) \\
&= \frac{C_1^2}{4} + \sigma^2 - C_1 \frac{\sigma}{n^2} \left(\sum_{(k,l); k \leq n/2} Z_{k,l} - \sum_{(k,l); k > n/2} Z_{k,l} \right) \xrightarrow{a.s.} \frac{C_1^2}{4} + \sigma^2.
\end{aligned}$$

In the same way, it can be shown that

$$\begin{aligned}
\text{Cov}(I_1, J_1) &= -\frac{C_1}{2} \frac{\sigma}{n^2} \left(\sum_{(k,l); k \leq n/2} X_{k,l} - \sum_{(k,l); k > n/2} X_{k,l} \right) + \frac{\sigma^2}{n^2} \sum_{k,l} (X_{k,l} - \bar{X})(Z_{k,l} - \bar{Z}) \xrightarrow{a.s.} 0 \\
\text{and } \text{Cov}(J_1, J_2) &\xrightarrow{a.s.} \frac{C_1 C_2}{4}.
\end{aligned}$$

□

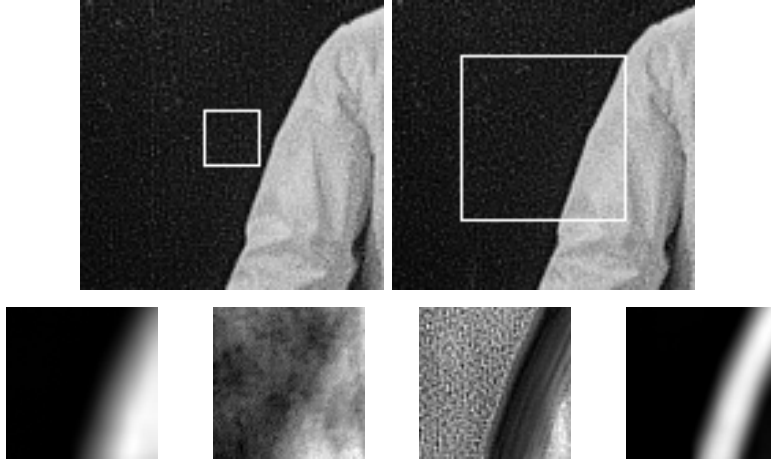


FIG. 3.2. *Similarity map for a constant patch. Top: two consecutive images of a movie. Top left: patch I delimited by the white frame. Top right: neighborhood to which belongs the center of the patch J . Bottom, from left to right: similarity map ($J \rightarrow D(I, J)$) for the similarity measures L^2 , D_{angle} , D_{corr} and D_{aff} . For each map, the white grey level corresponds to the maximal value obtained on the neighborhood and the black grey level corresponds to the minimal one. We notice here that the similarity measure D_{aff} is the only one which is at the same time discriminant (the “distance” between the constant patch and an edge patch is large) and robust to affine contrast changes.*

If we look at the first two columns in the table of Proposition 3.1, we notice that both the angular similarity measure and the correlation measure are unable to discriminate between a constant image patch and an edge zone. Whereas for the affine similarity measure, we statistically have that $D_{\text{aff}}^2(I_1, J_1) > D_{\text{aff}}^2(I_1, I_2)$, which means that it is able to discriminate between constant zones and edge zones. Thus, the conclusion is the following:

COROLLARY 3.1. *The affine similarity measure D_{aff} is the only one of the three considered similarity measures which is able to discriminate between an edge patch and a constant patch.*

An experimental way to check this result is to compute similarity maps on real images. This is illustrated by Figures 3.2 and 3.3. For a given patch, we compute the similarity measures to all neighboring patches for four different measures: L^2 , D_{angle} , D_{corr} and D_{aff} . The similarity maps thus obtained are an experimental check of Proposition 3.1 on more realistic patches. In particular, we notice on the figures that the measures D_{corr} and D_{angle} are not very discriminant: patches in flat zones are similar to almost all other patches, whatever their geometry. On the other hand, the similarity measure D_{aff} is able to select only the patches which have the same geometry as the considered patch I .

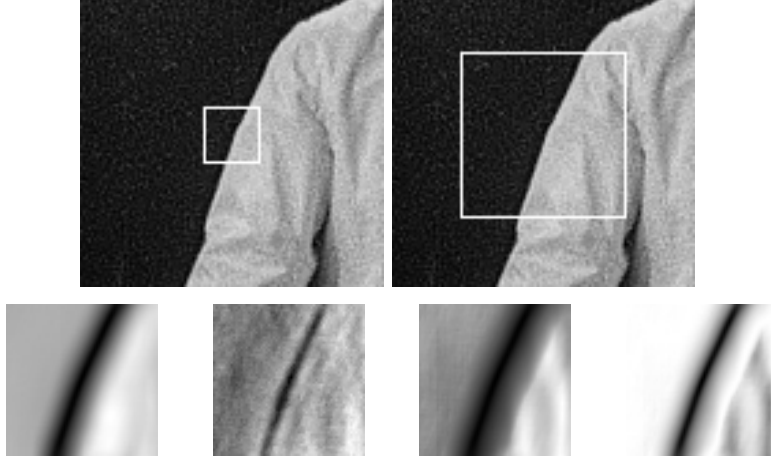


FIG. 3.3. *Similarity map for an edge patch. Top: two consecutive images of a movie. Top left: patch I delimited by the white frame. Top right: neighborhood to which belongs the center of the patch J . Bottom, from left to right: similarity map ($J \rightarrow D(I, J)$) for the similarity measures L^2 , D_{angle} , D_{corr} and D_{aff} . For each map, the white grey level corresponds to the maximal value obtained on the neighborhood and the black grey level corresponds to the minimal one. Notice again that the similarity measure D_{aff} is the only one which is both discriminant and robust to affine contrast changes.*

The conclusions of both the theoretical analysis and the experimental study are the same: among the considered similarity measures, the “optimal” one is D_{aff} . Thus, in all the following, we will use $D = D_{\text{aff}}$ to measure the similarity between image patches.

3.2. Motion based flicker reduction. In this section, we present an operator for local contrast stabilization, called LStab (for Local Stabilization), that relies on the similarity measure D_{aff} .

Let u be a discrete movie damaged by some flicker (local or not). Let Λ be a square of pixels centered at 0. For $x \in \Omega$, we consider the patch $u_t(x + \Lambda)$ centered in x . Let Λ_x denote the set $\{x + \Lambda\}$. The similarity measure D_{aff} introduced in Section 3.1 is robust to contrast changes, and thus can be used to roughly estimate the motion in the presence of flicker. Observe that we do not look for an highly accurate motion estimation, but only for an indication of motion. For this reason, the measure D_{aff} will be used without any regularity constraint. For each pixel x in u_t , its matching point $\varphi_{t,s}(x)$ in u_s is estimated by comparing $u_t(x + \Lambda)$ and $u_s(y + \Lambda)$ for all y in a suitable neighborhood:

$$\varphi_{t,s}(x) = \text{Argmin}_{y \in W_{t,s}(x)} D_{\text{aff}}(u_t(x + \Lambda), u_s(y + \Lambda)), \quad (3.1)$$

where $W_{t,s}(x)$ is a search window in u_s . In other words, the best corresponding patch is obtained by block-matching using the distance D_{aff} between patches. Obviously, we get in particular $\varphi_{t,t}(x) = x$. In practice, the choice of $W_{t,s}(x)$ is crucial and will be discussed in details in Section 3.5 (in particular, we will see how the size of this window should be related to the size of the patches). Let us mention however that this window can either be static and written $x + W$, with W a neighborhood of 0, or be updated at each frame and written $W_{t,s}(x) = \varphi_{t,s-1}(x) + W$ if $s > t$ and $W_{t,s}(x) = \varphi_{t,s+1}(x) + W$ if $s < t$. In this case, the estimation of $\varphi_{t,s}(x)$ depends on the estimation of $\varphi_{t,s-1}(x)$ (or $\varphi_{t,s+1}(x)$ if $s < t$).

Relying on this estimation, our weighted motion-based local contrast stabilization operator LStab is defined as:

$$\text{LStab}[u_t](x) = \frac{1}{\tilde{Z}_{t,x}} \int_s e^{-(t-s)^2/2\sigma^2} w_{t,x}(s, \varphi_{t,s}(x)) \underbrace{H_{s, \Lambda_{\varphi_{t,s}(x)}}^{-1} \left(\underbrace{H_{t, \Lambda_x}(u_t(x))}_{\text{rank of } x \text{ in the patch } u_t(\Lambda_x)} \right)}_{\text{grey level of the pixel of same rank as } x \text{ in } u_s(\Lambda_{\varphi_{t,s}(x)})} ds, \quad (3.2)$$

where H_{t,Λ_x} denotes the cumulative histogram of the patch $u_t(\Lambda_x)$ and where $w_{t,x}$ is a weight function given by $w_{t,x}(s, y) = \rho(D_{\text{aff}}^2(u_t(x + \Lambda), u_s(y + \Lambda)))$, with $\rho : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ decreasing, and $\tilde{Z}_{t,x}$ a normalizing constant which ensures that the total sum of all weights is equal to 1, *i.e.*

$$\tilde{Z}_{t,x} = \int_s e^{-(t-s)^2/2\sigma^2} w_{t,x}(s, \varphi_{t,s}(x)) ds.$$

It may happen that the Argmin operator in (3.1) is ambiguous and returns several points. In this case, one can simply use all of them in Equation (3.2).

The interest of balancing the different values by the weights $w_{t,x}(s, y)$ seems pretty clear if we think of a situation where the sequence changes a lot between the current frame t and the frame s : in this case, any motion estimation algorithm will still find a best correspondence to the patch $u_t(x + \Lambda)$ in the frame u_s , but this best correspondence will not make a lot of sense. When we stabilize brightness fluctuations along motion trajectories, using these weights allows to discard patches which are too different from the current patch.

In practice, we used in our experiments the decreasing function $\rho(x) = e^{-\frac{x^2}{h^2}}$, where h is a parameter that has to be tuned by the user. This parameter plays the same role as the value of the threshold in soft thresholding denoising methods: when h is “large”, all weights become almost equal, and thus the mean will mix everything together. On the other hand when h is “small”, only points which look very similar to x will have a non negligible weight in the sum. We will again discuss the way to set the value of h in the experimental section at the end of the paper. Observe that other decreasing functions could be used for g , such as a hard thresholding function for example.

One could argue at this point that the motion estimation performed by Equation (3.1) is not always accurate. Indeed, because of the redundancy of images, and because the similarity measure we use is robust to affine contrast changes, a pixel can have several “good” matches, and it is then uneasy to find the right one among them. It is particularly true in textured sequences such as the ones presented in Figures 4.5, 4.6, 4.7. The naive motion estimation provided by Equation (3.1) could thus be strengthened by regularity constraints, such as the ones used in classical motion estimation algorithms. However, we will see in the experimental section at the end of the paper that this flicker stabilization procedure provides very satisfying results, without needing a more complex and more precise motion estimation.

3.3. Multi-patch based flicker stabilization. Following the previous remark, we propose another stabilization operator, called SLStab (for Semi-Local Stabilization), which takes into account all the possible correspondences of a patch in order to stabilize the contrast, and not only the best correspondence. This operator is inspired by the NL-means method [7] introduced by Buades, Coll and Morel for movie denoising and which is known for the good results it provides. In SLStab, the grey level distribution of a patch $u_t(x + \Lambda)$ is replaced by a weighted mean (in the sense of Equation (2.1)) of the distributions of all patches $u_s(y + \Lambda)$. The mean is taken only for $y \in W_{t,s}(x)$ where $W_{t,s}(x)$ is defined as above, and weighted by the weights $w_{t,x}(s, y)$, in order to favour patches which are similar to $u_t(x + \Lambda)$.

More precisely, the weighted multi-patch stabilization operator SLStab can be written

$$\text{SLStab}[u_t](x) = \frac{1}{Z_{t,x}} \int_s C_{t,x}(s) \int_{y \in W_{t,s}(x)} w_{t,x}(s, y) H_{s,\Lambda_y}^{-1}(H_{t,\Lambda_x}(u_t(x))) dy ds, \quad (3.3)$$

where

- $C_{t,x}(s)$ is a normalizing constant which controls the total weight of u_s . In practice, as it was the case in Equation (2.1), we just set

$$C_{t,x}(s) = \frac{e^{-(t-s)^2/2\sigma^2}}{\int_y w_{t,x}(s, y) dy}.$$

Another possibility is to normalize each frame in such a way that its total weight is $\max_y w_{t,x}(s, y)$: this can be useful whenever there is a change of sequence in the movie;

- $Z_{t,x}$ is the final renormalization constant: it ensures that the total sum of all weights is 1,

$$Z_{t,x} = \int_s C_{t,x}(s) \int_y w_{t,x}(s,y) dy ds.$$

One of the main interests of this method relying on weighted patches is that it does not rely as heavily on motion estimation as the weighted motion one. If the search window $W_{t,s}(x)$ is fixed, the method does not rely on any motion estimation at all. If this can seem meaningless for “classical” motion sequences, it makes perfectly sense for biological or textured image sequences, such as the ones shown in Figures 4.5, 4.6, 4.7. Indeed, these sequences show non-rigid objects, which undergo non-rigid distortions. For these kind of movies, the concept of motion trajectories is not completely relevant anymore.

3.4. Links between LStab and SLStab. The operator LStab can be seen as the limit of SLStab when we introduce a real parameter p that we let go to infinity. More precisely, if we define the operator $\text{SLStab}^{(p)}$ by

$$\text{SLStab}^{(p)}[u_t](x) = \frac{1}{Z_{t,x}^{(p)}} \int_s \frac{e^{-(t-s)^2/2\sigma^2}}{\int_y w_{t,x}^p(s,y) dy} \int_{y \in \{W_{t,s}(x)\}} w_{t,x}^{p+1}(s,y) H_{s,\Lambda_y}^{-1}(H_{t,\Lambda_x}(u_t(x))) dy ds,$$

where

$$Z_{t,x}^{(p)} = \int_s \frac{e^{-(t-s)^2/2\sigma^2}}{\int_y w_{t,x}^p(s,y) dy} \int_{y \in \{W_{t,s}(x)\}} w_{t,x}^{p+1}(s,y) dy ds,$$

then $\text{SLStab}^{(0)} = \text{LStab}$ and $\text{SLStab}^{(p)}[u_t](x)$ goes to $\text{LStab}[u_t](x)$ as p goes to infinity. This amounts to change the parameter h^2 used in the weight function w in two different ways depending on the considered sum: it is changed into $h^2/(p+1)$ in the weighted sum of grey levels and it is changed into h^2/p in the normalization per frame. This result is analogous to the fact that in some sense “the L^∞ norm is the limit of the L^p norm as p goes to infinity”.

3.5. Setting the parameters. The different parameters (the parameter h in the decreasing function $\rho(x) = e^{-\frac{x^2}{h^2}}$, the size of Λ and the size of W) used in the previous stabilization procedures are important and should be chosen carefully.

On the size of the patches $|\Lambda|$. If $|\Lambda|$ is too large, the patches can contain different moving objects and the estimation of the similarity measure $D_{\text{aff}}(u_t(x+\Lambda), u_s(y+\Lambda))$ can be unreliable. However, if $|\Lambda|$ is too small, the information in the patches can be poor and the measure $D_{\text{aff}}(u_t(x+\Lambda), u_s(y+\Lambda))$ can not be trusted either. A good choice for the size $|\Lambda|$ is clearly related to the resolution of the frames and the size of the objects moving in the sequences. In all the experiments presented in this paper, Λ was set as a 21×21 square window.

On the choice of the parameter h . The parameter h is related to the values taken by the similarity measure D_{aff} on the image. Since the weights are given by $\exp(-D_{\text{aff}}(\cdot, \cdot)^2/h^2)$, h can be seen as a soft threshold. It can also be seen as a regularization scale for the flicker. If h is large, then all weights in (3.3) will be close to 1. In this case, the flicker is efficiently stabilized, but some artefacts can appear since patches which can have very dissimilar geometries are mixed together. Conversely, when h is small, only patches which are very similar to the considered frame will have non negligible weights in the sum. But then, flicker is not as efficiently stabilized. In practice, we observed that values of h between 10 and 30 (the distance D_{aff} between patches being normalized by the size of the patches) generally yield good results on most sequences.

On the size of the search window $|W|$. The size of the search window W should be large enough to follow a pixel in the presence of a large motion. On the other hand, if W is too large, there is a risk that patches corresponding to different objects, with the same geometry but different contrast, will be mixed together. Now, a very simple and elementary requirement for the choice of W (and for the whole approach in general) is that if a movie u does not contain any motion nor any flicker (which means that all frames are the same - let v denote one of these frames), then

it should be invariant under the operators LStab and SLStab. To satisfy such a requirement, we need to ensure that for any pixel x , a patch $v(y + \Lambda)$ with its center y in the window $x + W$ and which is similar (according to the similarity measure D_{aff}) to the patch $v(x + \Lambda)$ will not create “new” grey level values at x when equalizing the grey level histograms. In other words, if we consider the operator \mathcal{T} defined on a single image v by

$$\mathcal{T}v(x) = \frac{\int_{x+W} w(x, y) H_{\Lambda_y}^{-1}(H_{\Lambda_x}(v(x))) dy}{\int_{x+W} w(x, y) dy}, \quad (3.4)$$

where $w(x, y) = \exp(-D_{\text{aff}}^2(v(x + V), v(y + V))/h^2)$, then we should have $\mathcal{T}v \simeq v$.

PROPOSITION 3.2. *In order to avoid any “ringing” effect in the neighborhood of edges, we have to set*

$$W \subset 2\Lambda.$$

Proof. Let v be the image of a noisy edge, as defined in Proposition 3.1. We assume that the image domain is $\{-N, \dots, -1, 0, 1, \dots, N\}^2$ and that v is given by: $v(x) = v(x_1, x_2) = \alpha + \sigma n(x)$ if $x_1 < 0$, and $v(x) = \beta + \sigma n(x)$ if $x_1 \geq 0$, where α and β are two grey levels, σ is the standard deviation of the noise, and the $n(x)$ are i.i.d. $\mathcal{N}(0, 1)$.

We then consider a patch $v(x + \Lambda)$ where $\Lambda = \{-K, \dots, -1, 0, 1, \dots, K\}^2$. By symmetry, we will just compute what happens on the right side of the edge (*i.e.* in the domain $x_1 \geq 0$). Let $r(x) = H_{\Lambda_x}(v(x))$. By definition, it is the rank of $v(x)$ in the set of all $v(x + z)$ with $z \in \Lambda$. If $x = (x_1, x_2)$ with $x_1 \geq K$, then the patch $v(x + \Lambda)$ doesn't intersect the edge. It is thus an almost constant patch (up to the additive noise), and then $r(x)$ is uniformly distributed between 1 and $|\Lambda| = (2K + 1)^2$.

The affine similarity measure between $v(x + \Lambda)$ and another patch $v(y + \Lambda)$ can be easily computed (in a way similar to what we did in the proof of Proposition 3.1), and we obtain that: $D_{\text{aff}}^2(v(x + \Lambda), v(y + \Lambda)) \simeq \sigma^2$ when $v(y + \Lambda)$ is another constant patch (that is when $y_1 \geq K$ or $y_1 < -K$). And $D_{\text{aff}}^2(v(x + \Lambda), v(y + \Lambda)) \simeq p_y(1 - p_y)(b - a)^2 + \sigma^2$ when $v(y + \Lambda)$ is a patch containing the edge and where $p_y = (K - y_1)/(2K + 1)$ is the proportion of $\simeq \alpha$ values in the patch.

On the other hand, we can also compute the expected values for $H_{\Lambda_y}^{-1}(H_{\Lambda_x}(v(x)))$ and get:

$$H_{\Lambda_y}^{-1}(H_{\Lambda_x}(v(x))) = H_{\Lambda_y}^{-1}(r(x)) = \begin{cases} \alpha & \text{if } y_1 < -K, \\ \beta & \text{if } y_1 \geq K, \\ \alpha p_y + \beta(1 - p_y) & \text{otherwise.} \end{cases}$$

Let now W be the neighborhood used in the definition of the operator \mathcal{T} in Equation (3.4). We will denote it by $W = \{-m, \dots, -1, 0, 1, \dots, m\}^2$.

If $m > 2K$, then there will exist points x such that $K \leq x_1 < m - K$. For all these points, since $x_1 - m < -K$ their neighborhood $x + W$ will contain points y such that $y_1 < -K$. Thus, since the similarity measure between $v(x + \Lambda)$ and all other patches $v(y + \Lambda)$ such that $y_1 < -K$ or $y_1 \geq K$ is the same and smaller than the similarity measure to edge patches, the grey level $\mathcal{T}v(x)$ will be a weighted mean of α and β values, and will not remain close to $v(x) \simeq \beta$. This effect can clearly be observed on Figure 3.5. Thus, in order to avoid creating such artefacts, we have to set $m \leq 2K$, in other words $W \subset 2\Lambda$.

On the other hand, points x which are such that the patch $v(x + \Lambda)$ contains the edge will have an almost unchanged grey level because there are no patches in the image which will have the same geometry but with very different grey level values.

The typical edge profiles that we obtain after applying Equation (3.4) is shown on Figure 3.4.

□

4. Experiments and discussion. This section presents the results of the proposed stabilization procedures on several sequences, containing real or synthetic flicker. For the sake of completeness, most of these sequences and their corresponding results are available at http://www.tsi.enst.fr/~delon/Demos/Flicker_stabilization/.

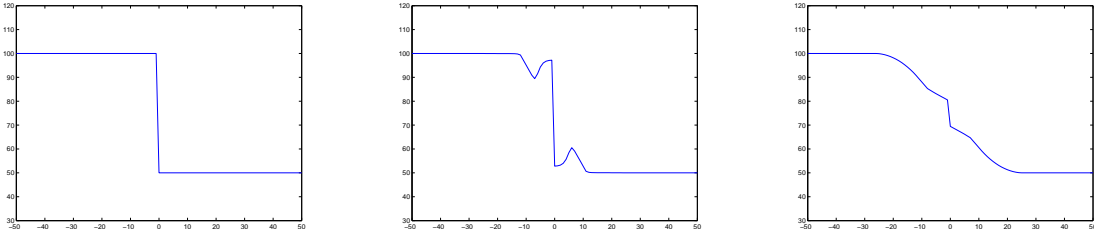


FIG. 3.4. Result of Equation (3.4) when W is larger than Λ . In this experiment, the size of Λ is 15×15 pixels and the size of the search window W is 37×37 . Left: profile of a pure edge v . Middle: profile of $\mathcal{T}v$ when $h = 10$. This profile shows some “ringing phenomenon”. Right: profile of $\mathcal{T}v$ when $h = 100$. In this case, since h is large, all patches have an almost equal weight, whatever their geometry.

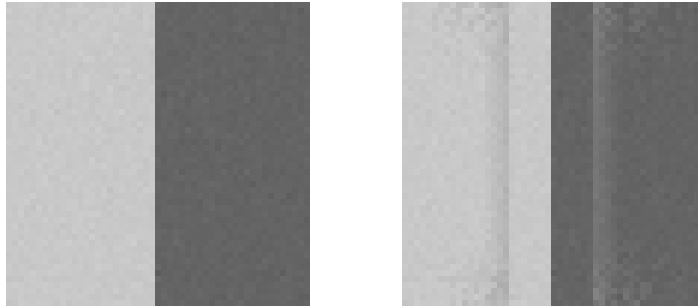


FIG. 3.5. Left: image v of a noisy edge (size 67×67 pixels). Right: image $\mathcal{T}v$ when the size of Λ is 15×15 pixels and the size of the search window W is 37×37 . The search window is too large compared to the size of the patches, and some artefacts appear: in particular, we can notice some kind of “ringing phenomenon” around the edge.

4.1. Implementation. Equations (3.2) and (3.3) can be implemented independently on each pixel x of u_t . However, the computation of the weighted motion for the neighborhood Λ_x of each pixel x in Ω can take time. In order to accelerate the process, the stabilization procedure can be implemented by blocks. The domain Ω is divided into a fixed grid of overlapping blocks Λ_i , $i = 1, \dots, N_B$. Equation (3.2) (or (3.3)) is then used on each of these blocks to stabilize the flicker of all pixels contained in the block (and not only the central pixel of the block). If a pixel x belongs to several blocks $\Lambda_{i_1}, \dots, \Lambda_{i_x}$, the restored grey levels obtained at x for each Λ_{i_j} are averaged to obtain the final restored value at x . This averaging regularization is a way to avoid block effects.

The following experiments are obtained using this block-based implementation. In these experiments, the block (or patch) Λ is always chosen as a square of size 21×21 . The overlap between these blocks is chosen as half the size of a block, *i.e.* as 10 here. The search window W is also chosen as 21×21 in most of the experiments, which is large enough for the motion observed in the sequences presented here and which permits to satisfy the condition of Proposition 3.2. The temporal standard deviation σ is set to 5, which in practice means that the temporal neighborhood taken into account in the restoration of u_t is approximately $[t - 2\sigma, t + 2\sigma] = [t - 10, t + 10]$ (10 images before t and 10 images after t).

4.2. Effects on flicker-free sequences. A crucial property of a flicker stabilization procedure is its ability to leave intact sequences which do not contain any flicker. Figure 4.1 illustrates this property. On the left, one can see a frame extracted from a sequence containing several artifacts but no visible flicker. After applying the operator LStab (Equation (3.2)) to the whole sequence with $h = 10$, the same frame in the modified sequence is shown in the center of the figure. The difference between both frames is shown on the right, rescaled from its range $[0, 15]$ to $[0, 255]$ for better visualization. The two images are almost identical and the mean grey level difference between both images is 0.79. If the sequence is stabilized with $h = 30$, the mean grey level difference becomes 0.91, which is still quite small. Although this mean difference remains moderate, a slight difference can be perceived at some very specific locations. These zones correspond to small



FIG. 4.1. *Left: a frame of a sequence containing no visible flicker. Center: same frame, after applying the operator $LStab$ (Equation (3.2)) with $h = 10$ to the whole sequence. Right: difference between both frames, rescaled from $[0, 15]$ to $[0, 255]$ for better visualization.*

blotches, caused by dirt on the film support. These blotches are present in this frame but do not correspond to any information in other frames of the movie. They tend to disappear under the stabilization procedure, since the grey level distribution of their neighborhood is mixed with neighborhoods in other frames which do not contain blotches. As a consequence, these blotches appear clearly when we compute the difference between the original sequence and the regularized one.

4.3. Synthetic flicker. One of the worst difficulties of old film restoration is that most of the films suffer from several defects (blotches, scratches, flicker) at the same time. As it can be noticed on the example of Figure 4.1, the restoration of a single defect is not completely independent of the presence of the others. For this reason, we begin our experiments with two sequences containing no such defects. The first sequence, called *flag* is build from a single 247×188 frame, to which we added independent samples of a white Gaussian noise random image. This sequence, on purpose, does not contain any motion. The second sequence is the classical sequence *taxi*¹, generally used in motion estimation benchmarks. In this sequence, the size of the frames is 256×191 .

To each sequence, we added two kinds of synthetic flicker. The first one consists in a random multiplicative and local flicker, placed on random positions along the sequence. More precisely, each frame u_t of size $N_c \times N_r$ becomes

$$\forall(k, l) \in \Omega, \quad \tilde{u}_t(k, l) = u_t(k, l) \times \left(1 + S_t e^{-\frac{(k-X_t)^2 + (l-Y_t)^2}{r^2}} \right), \quad (4.1)$$

where S_t , X_t and Y_t are independent and $S_t \sim \mathcal{U}[-0.3, 0.3]$, $X_t \sim \mathcal{U}[0, N_c]$, $Y_t \sim \mathcal{U}[0, N_r]$; and where the radius r is fixed to 100. This flicker has no motion coherence (see Figure 4.2 (a) and 4.3 (a)). The frames become randomly locally darker or lighter. The second kind of flicker consists in a multiplicative dark transparent strip moving from the top left to the down right of the frames as the movie is played. More precisely, a frame u_t becomes

$$\forall(k, l) \in \Omega, \quad \tilde{u}_t(k, l) = u_t(k, l) \times \left(0.6 + 0.4 e^{-\frac{w}{0.5+k+l-(t+1)\gamma}} \right), \quad (4.2)$$

where w is the width of the strip and γ is the speed of the flicker motion. The width w is set to 60 for both sequences and the speed s is set to 40 for the *flag* sequence and to 15 for the *taxi* sequence. This flicker is motion-coherent (see Figures 4.2 (c) and 4.3 (c)).

The four resulting sequences and the results of the $LStab$ operator (Equation (3.2)) using the implementation described in Section 4.1 can be seen at the address http://www.tsi.enst.fr/~delon/Demos/Flicker_stabilization/. All the parameters are set as described in Section 4.1

¹This sequence is available at http://i21www.ira.uka.de/image_sequences/.

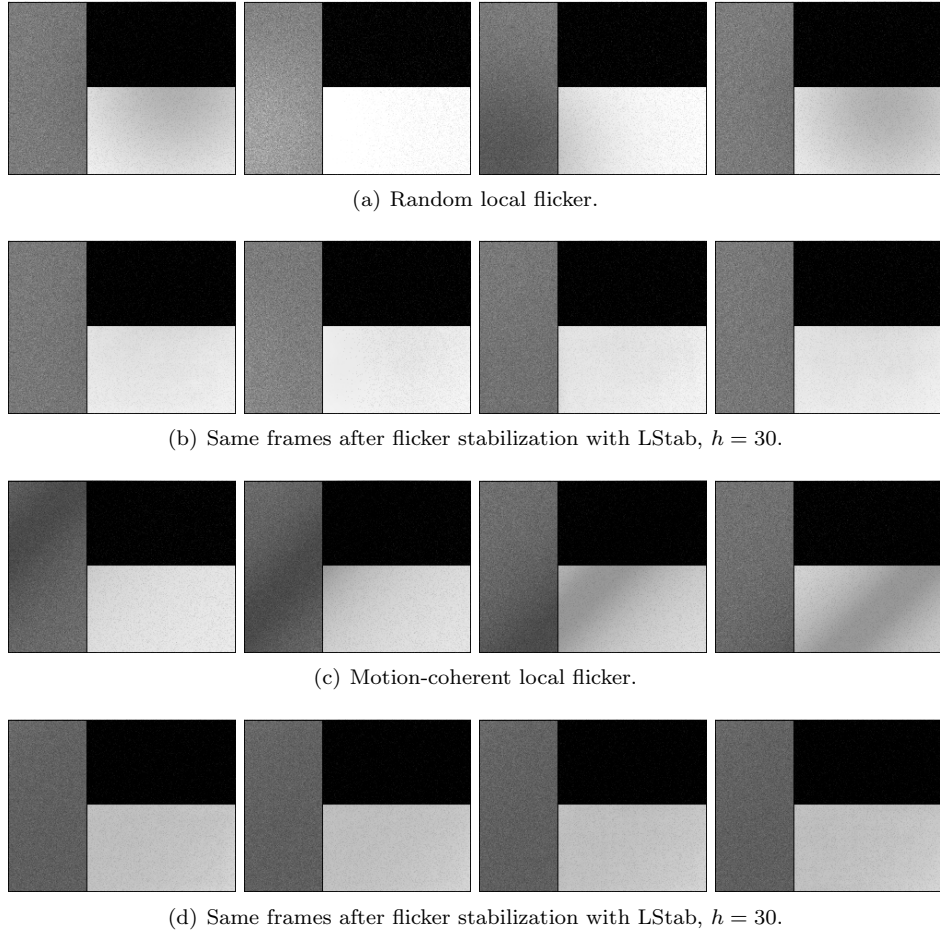


FIG. 4.2. (a) Four frames of the flag sequence, affected by the random flicker given by Equation (4.1). (b) Same frames after using the operator $LStab$ (Equation (3.2)) to remove the flicker. (c) Four frames of the flag sequence, affected by a motion-coherent flicker given by Equation (4.2) (a dark strip that goes through the images). (d) Same frames after using the operator $LStab$ to remove the flicker. One can notice that in both cases, since we only perform changes of contrast, there is no spatial smoothing, and thus the noisy texture of the movie remains intact.

and h is set to 30. Some frames extracted from the different sequences and the corresponding frames in their restored versions can also be seen on Figures 4.2 and 4.3. Observe that the restored sequences are not identical and depend strongly on the flicker that we try to eliminate. For instance, the contrast of the frames in Figure 4.2 (b) (resp. Figure 4.3 (b)) is larger than the contrast of the frames in 4.2 (d) (resp. Figure 4.3 (d)). Since the flicker has destroyed some contrast information, we cannot hope to recover completely the contrast of the original sequence, but only to filter it enough to make it barely noticeable. Observe also on the *flag* sequence how the flicker disappears while the noisy texture remains intact. This is due to the fact that only changes of contrast are performed, without spatial smoothing.

It is generally not easy to evaluate the performance of a flicker stabilization method because in most cases there is no available ground truth. However, in the case of a synthetic flicker, we can compare a frame of the original sequence with the same frame in the flickering sequence and with the same frame after flicker stabilization. In order to measure the ability of the stabilization procedure to recover the geometrical information of the original sequence, we compare the topographic map of an original frame with the one of the corresponding restored frame. This is illustrated by Figure 4.4. On the center, we notice that the local flicker has modified some level lines. In particular, it has created level lines which are parallel to the dark transparent strip (see



(a) Random local flicker.



(b) Same frames after flicker stabilization with LStab, $h = 30$.



(c) Motion-coherent local flicker.



(d) Same frames after flicker stabilization with LStab, $h = 30$.

FIG. 4.3. (a) Four frames of the taxi sequence with an additional random local flicker, given by Equation (4.1). (b) Same frames after applying the operator $LStab$ to remove the flicker. (c) Four frames of the taxi sequence with an additional local flicker, looking like a dark transparent strip moving across the frames, given by Equation (4.2). (d) Same frames after applying the operator $LStab$ to remove the flicker.

below the white car, on the right of the black car). We also see that the most contrasted level lines, which correspond to the salient objects in the image are not modified. On the right of the figure, we show the level lines after flicker stabilization by $LStab$. The level lines created by the flicker have disappeared in this restored frame.

4.4. Biological sequences. In this section, we test the $LStab$ and $SLStab$ operators on three biological sequences. The first sequence, called *Drosophila*, represents a *Drosophila* wing imaginal disc (kindly provided to us by F. Graner, F. Lense and Y. Bellaïche, from Institut Curie, Paris). It has been acquired thanks to a spinning disk microscope, camera Roper HQ2. The second sequence, called *Cells*, provided by Marion Ghibaudo and Benoît Ladoux (MSC Lab., Paris Diderot University) is a movie of an array of fluorescent micropillars, on which cells exert forces. The motion of the pillars is estimated by biologists [18] to measure the different forces exerted by the cells. The last sequence, called *Physarum*, was provided by Marc Durand (MSC Lab., Paris Diderot University) and shows a *physarum polycephalum*, which is moving in quest of food. The three movies contain naturally strong flicker-like effects, which cannot be entirely explained by their acquisition protocols. We show in Figures 4.5, 4.6 and 4.7 the results on these sequences of the combination $STE+SLStab$ (equation (2.2) followed by equation (3.3), with $h = 10$). The same kind of results could be obtained by using the operator $LStab$ instead of $SLStab$. In all cases, the flicker effects are completely stabilized, which permits a better visual estimate of the fine motions

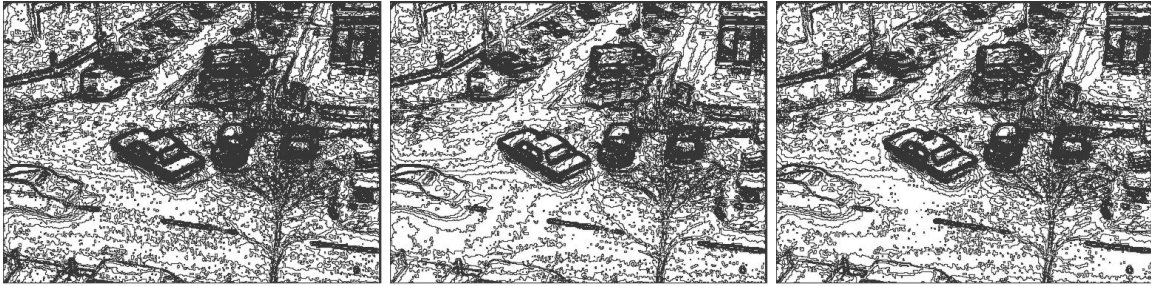
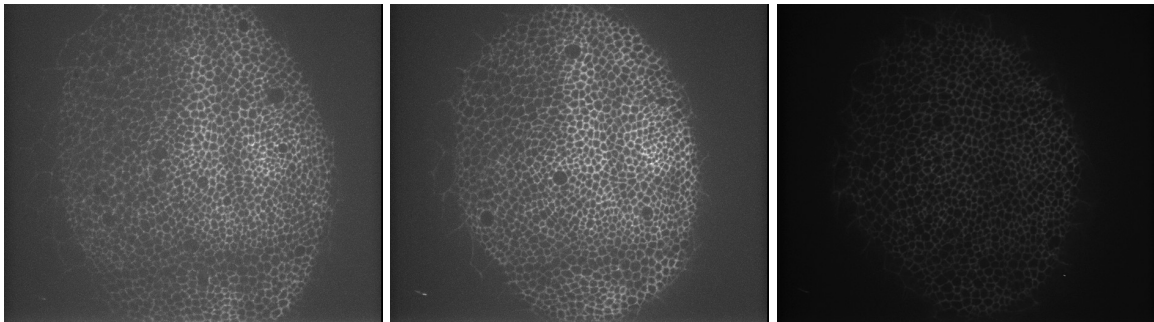
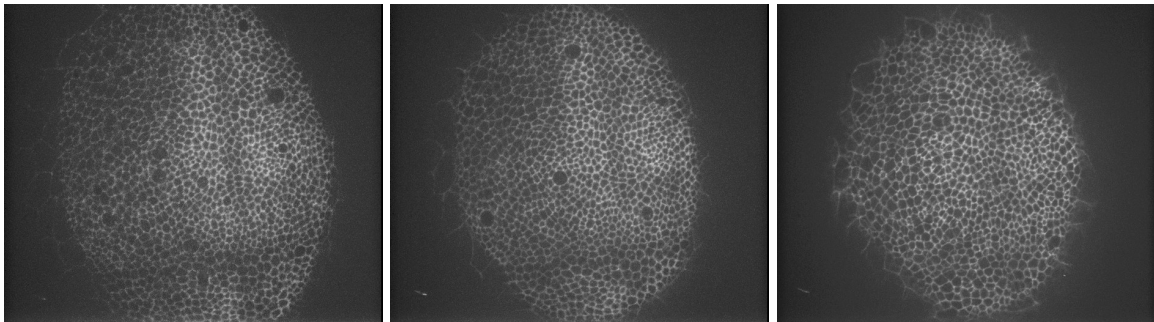


FIG. 4.4. *Left: level lines of a frame of the original taxi sequence. The figure shows all level lines at a sample rate of 5 grey levels. Center: level lines of the same frame in the sequence containing the motion-coherent flicker. We notice that the local flicker has modified some level lines. In particular, it has created level lines which are parallel to the dark transparent strip (see below the white car, on the right of the black car). We also see that the most contrasted level lines, which correspond to the salient objects in the image are not modified. Right: level lines of the same frame in the sequence restored thanks to the operator LStab (Equation (3.2)). The level lines created by the flicker have disappeared in this restored frame.*

and distortions in the sequences. Observe that in these sequences, the sole application of the global correction STE removes most of the visible flicker effects, but is not adequate to remove all the fine local contrast changes. This is particularly obvious on the *Physarum* sequence shown in motion (see http://www.tsi.enst.fr/~delon/Demos/Flicker_stabilization/).



(a) Three frames from the original *Drosophila* sequence.

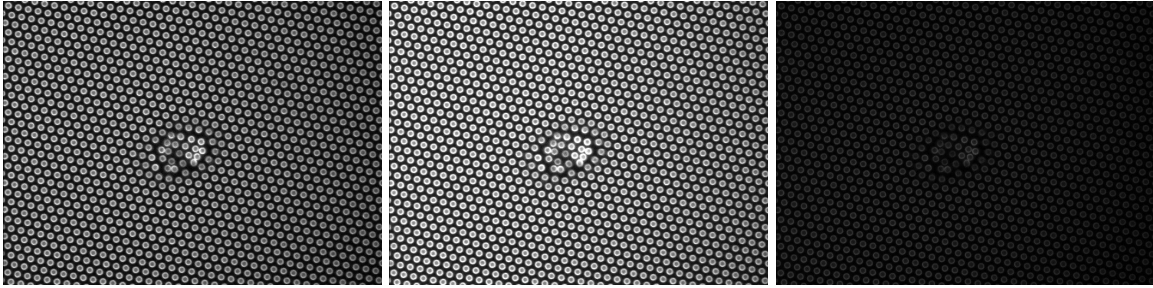


(b) Same frames in the sequence stabilized with STE + SLStab, $h = 10$.

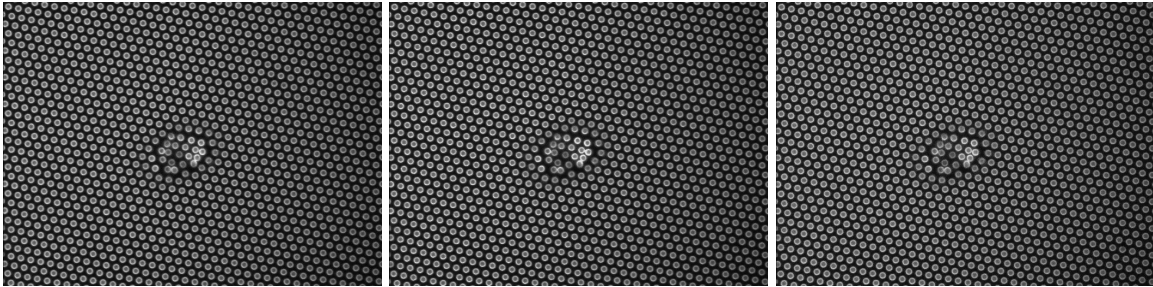
FIG. 4.5. (a) Three images of the *Drosophila* sequence showing important flicker-like effects. (b) Same frames after the successive applications of the operators STE and SLStab (Equation (3.3)), with $h = 10$ and a search window of size 11×11 . Other parameters are set as described in Section 4.1.

4.5. Sequences extracted from old movies. We finally show the results of the LStab operator on four sequences extracted from old movies. Some frames extracted from the original and stabilized sequences are shown on Figures 4.8, 4.9, 4.10 and 4.11.

The first two sequences, called *Boat* and *GreatWall*, are extracts of the movie *Our Shrinking*



(a) Three frames from the original *Cells* sequence.



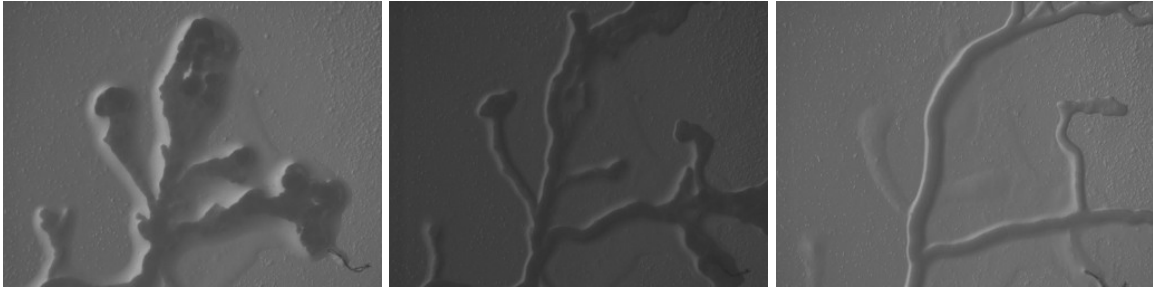
(b) Same frames in the sequence stabilized with STE + SLStab, $h = 10$.

FIG. 4.6. (a) Three images from the *Cells* sequence [18] showing important flicker-like effects. (b) Same frames after the successive applications of the operators STE and SLStab (Equation (3.3)), with $h = 10$. Other parameters are set as described in Section 4.1.

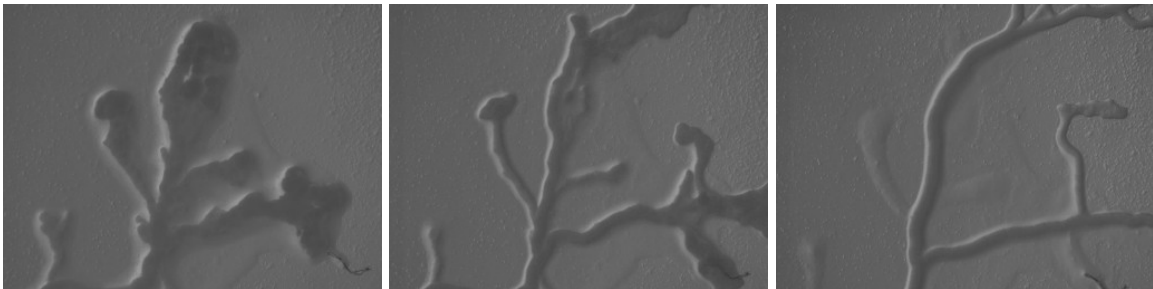
world (1946), available on the website www.archive.org. The first sequence is 93 frames long and contains small movements in the foreground, with a static, although unsteady, background. The second sequence is 150 frames long and consists in a tilting of the camera on the Great Wall. They both present challenging local flicker degradations, which can not be removed by a global correction. We chose these two sequences to confront our stabilization scheme with state of the art flicker removal procedures. Indeed, according to the results kindly provided by G.Forbin at <http://personal.ee.surrey.ac.uk/Personal/G.Forbin/results/>, these degradations are not entirely stabilized by recent methods as the ones of [36, 20]. Figures 4.8 and 4.9 illustrate the results of the STE + LStab combination on these sequences, with $h = 10$. For a complete and fair comparison, the results of our stabilization scheme on the whole sequences are available at http://www.tsi.enst.fr/~delon/Demos/Flicker_stabilization/. Observe that on these two sequences, the scheme introduced in this paper show similar or better results than the ones presented in [36, 20]. For instance, with these last two approaches, some important contrast changes could still be observed on the pyramid, on the right part of the *Boat* sequence, while these contrast changes have completely disappeared with our approach. Figure 4.8(c) shows the absolute differences between the frames 9 and 67 of the sequence, before and after the stabilization. Notice how the differences are distributed over the whole image domain before the stabilization, and only restricted to motion edges after.

In the third sequence (see Figure 4.10), which is 80 frames long, the flicker is a mix of global contrast changes and of a dark transparent strip going through the images from right to left when the movie is being played (the strip is placed on the face of the man on the middle frame in the figure). This sequence can thus be compared to the synthetic sequences created in the previous section with the motion-coherent flicker. After applying the LStab operator, with $h = 30$, the other parameters being set as described in Section 4.1, the flicker is not visible anymore.

The last sequence is a short extract of 101 frames from the movie *Les Aventures des Pieds Nickelés*, Emile Cohl/Eclair (1917-1918, copyright: Marc Sandberg). In this sequence, shown on Figure 4.11 (a), the flicker has a strong global component, although local variations can also



(a) Frames 1, 42 and 170 from the *Physarum* sequence.



(b) Frames 1, 42 and 170 from the *Physarum* sequence, after stabilization with STE + SLStab, $h = 10$.

FIG. 4.7. (a) Three frames from the *Physarum* sequence. (b) Same frames after the successive applications of the operators STE and SLStab (Equation (3.3)), with $h = 10$. Other parameters are set as described in Section 4.1.

be perceived along the sequence. As a consequence, we applied first the global operator STE (Equation (2.2)) to this sequence, with a large value of σ to get rid of these strong global contrast changes along the sequence. After this global stabilization, local and fast fluctuations remained visible, taking the shape of large bright spots at different locations on the frames. This is illustrated by Figure 4.12 (a), which shows the absolute difference between two successive frames of the sequence after the global correction. The visible differences are not only due to motion (or film shaking in this case) but appear also as large bright spots at different places. As shown in Figures 4.11 (b) and 4.12 (b), applying LStab with $h = 30$ to this sequence enabled us to remove these spots and the remaining flicker.

5. Conclusion. In this paper, a new procedure to stabilize flicker-like effects in image sequences has been proposed. The method relies on a similarity measure between image patches that has been demonstrated to be at the same time robust to affine contrast changes and discriminant. This similarity measure permits to roughly estimate the motion in the movie despite the presence of flicker. The main stabilization operator introduced in this paper, called LStab, makes use of this similarity measure to find the correspondences of a given patch and replaces the grey level distribution of the patch by a weighted mean of the grey level distributions of all these corresponding patches, the weights being chosen as a decreasing function of the similarity measures between patches. The ability of this operator to deal with strong flicker has been demonstrated on several synthetic and real examples. In particular, since this operator only performs contrast changes on image patches, and does not involve spatial smoothing, it is able to preserve fine spatial features such as noise or textures in the frames.

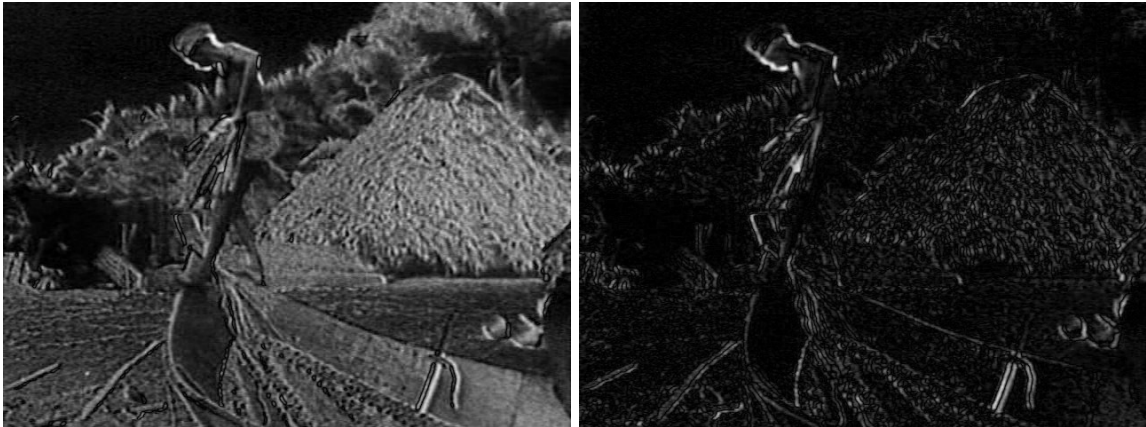
The perspectives of this work are twofold. First, we plan to study in more detail the theoretical aspects of the method. In particular, the different operators introduced (LStab and SLStab for movies, and \mathcal{T} for a single image) should certainly find statistical interpretations, in the same vein as what has been done for the NL-means operator [7]. Another perspective lies in the adaptation of this work to color sequences. Among the questions that color sequences raise, let us mention the choice of a measure of similarity between color patches equivalent to D_{aff} and the generalization of the scale time equalization (STE) to 3D color histograms. This perspective could certainly benefit



(a) Frames 9, 67 and 93 from the original sequence.



(b) Same frames in the sequence stabilized with STE + LStab, $h = 10$.



(c) Absolute differences between the frames 9 and 67, before and after stabilization.

FIG. 4.8. (a) Three images of the sequence Boat, suffering from a strong local flicker. (b) Same frames after flicker stabilization by Operator LStab (Equation (3.2)), with $h = 10$. Other parameters are set as described in Section 4.1. (c) On the left, absolute difference between the first and second frames of line (a). Because of flicker-like effects, large differences can be observed everywhere. On the right, absolute difference between the first and second frames of line (b). The only differences are located around the edges and are due to motion.

from recent developments in optimal transportation theory and will require future investigation.

REFERENCES

- [1] E. Bennett and L. McMillan. Video enhancement using per-pixel virtual exposures. In *ACM SIGGRAPH 2005 Papers*, page 852. ACM, 2005.
- [2] L. Birgé. The Grenander estimator: A nonasymptotic approach. *The Annals of Statistics*, 17(4):1532–1549, 1989.
- [3] M. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow-fields. *Computer Vision and Image Understanding*, 63(1):75–104, January 1996.
- [4] L. Bogoni. Extending dynamic range of monochrome and color images through fusion. In *International Conference on Pattern Recognition*, volume 15, pages 7–12, 2000.



(a) Frames from the original sequence.



(b) Same frames in the sequence stabilized with STE + LStab, $h = 10$.

FIG. 4.9. (a) Three images of the sequence Greatwall suffering from a strong local flicker. (b) Same frames after flicker stabilization by Equation (3.2), with $h = 10$. Other parameters are set as described in Section 4.1.



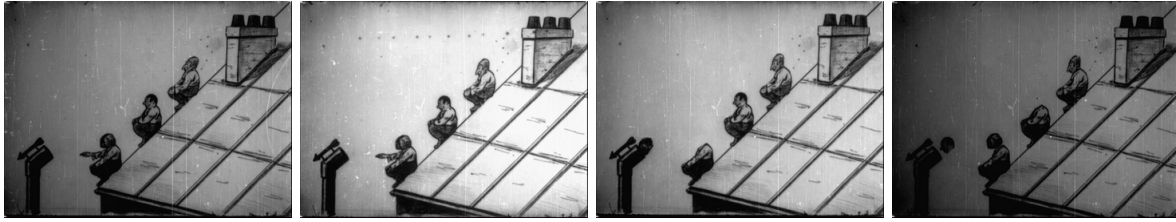
(a) Frames from the original sequence



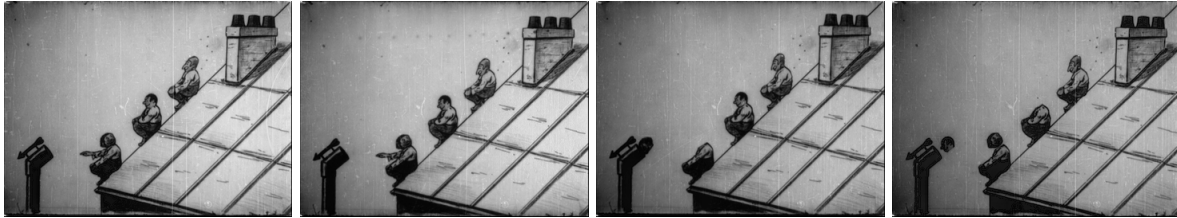
(b) Same frames in the sequence stabilized with LStab, $h = 30$

FIG. 4.10. (a) Three images of an old movie suffering from a strong local flicker. This flicker is visible as a dark transparent strip going through the images from right to left when the movie is being played. The strip is placed on the face of the man on the middle frame. (b) Same frames after flicker stabilization by Equation (3.2), with $h = 30$. Other parameters are set as described in Section 4.1.

- [5] N. Bonnier and E. Simoncelli. Locally adaptive multiscale contrast optimization. In *International Conference on Image Processing 2005*, pages I: 949–952, 2005.
- [6] J. Boulanger, J.-B. Sibarita, C. Kervrann, and P. Bouthemy. Non-parametric regression for patch-based fluorescence microscopy image sequence denoising. In *Proc. IEEE Int. Symp. on Biomedical Imaging: from nano to macro (ISBI'08)*, pages 748–751, Paris, France, May 2008.
- [7] A. Buades, B. Coll, and J.-M. Morel. Nonlocal image and movie denoising. *Int. J. Comput. Vision*, 76(2):123–139, 2008.
- [8] G. Buonaccorsi, C. Roberts, S. Cheung, Y. Watson, J. O'Connor, K. Davies, A. Jackson, G. Jayson, and G. Parker. Comparison of the performance of tracer kinetic model-driven registration for dynamic contrast enhanced MRI using different models of contrast enhancement. *Academic Radiology*, 13(9):1112–1123, 2006.
- [9] V. Caselles, B. Coll, and J.-M. Morel. Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33(1):5–27, 1999.

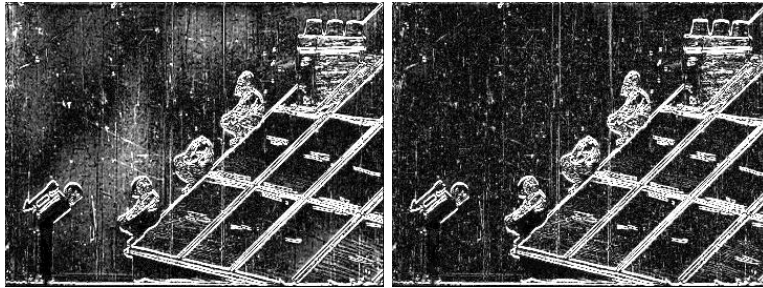


(a) Frames from the original sequence.



(b) Same frames in the sequence stabilized with STE + LStab, $h = 30$.

FIG. 4.11. (a) Four images of an old movie *Les Aventures des Pieds Nickelés*, *Emile Cohl/Eclair* (1917-1918, copyright: Marc Sandberg) suffering from local flicker. (b) Same frames after a global correction by STE, followed by a local stabilization by LStab, with $h = 30$. Other parameters are set as described in Section 4.1.



(a) After STE.

(b) After STE + LStab.

FIG. 4.12. Left: difference between two successive frames of *Les Aventures des Pieds Nickelés* after a global flicker stabilization thanks to the operator STE (Equation (2.2)). Right: difference between the same frames after an additional local stabilization by LStab, with $h = 30$. In both differences, a ratio of 10% pixels with maximal values has been put to the maximal value for better visualization.

- [10] V. Caselles, L. Garrido, and L. Igual. A contrast invariant approach to motion estimation. In *Scale Space*, pages 7–9. Springer, 2005.
- [11] V. Caselles, L. Garrido, and L. Igual. A Contrast Invariant Approach to Motion Estimation: Validation and Application to Motion Estimation Improvement. *MATHEMATICS IN INDUSTRY*, 12:863, 2008.
- [12] V. Caselles, J.-L. Lisani, J.-M. Morel, and G. Sapiro. Shape preserving local histogram modification. *IEEE Transactions on Image Processing*, 8(2):220–230, 1999.
- [13] V. Caselles and P. Monasse. *Geometric Description of Topographic Maps and Applications to Image Processing*. Lecture Notes in Mathematics. Springer, 2009. To Appear.
- [14] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIG-GRAPH 2008 classes*, page 31. ACM, 2008.
- [15] E. Decencière. *Restauration automatique de films anciens*. PhD thesis, École Nationale Supérieure des Mines de Paris, 1997.
- [16] J. Delon. Midway image equalization. *Journal of Mathematical Imaging and Vision*, 21(2):119–134, 2004.
- [17] J. Delon. Movie and video scale-time equalization. *IEEE Transactions on Image Processing*, 15(1):241–248, 2006.
- [18] O. Du Roure, A. Saez, A. Buguin, R. Austin, P. Chavrier, P. Siberzan, and B. Ladoux. Force mapping in epithelial cell migration. *Proceedings of the National Academy of Sciences*, 102(7):2390, 2005.
- [19] E. Eisemann and F. Durand. Flash photography enhancement via intrinsic relighting. *ACM Transactions on Graphics (TOG)*, 23(3):673–678, 2004.
- [20] G. Forbin and T. Vlachos. Flicker compensation for archived film sequences using a segmentation-based

- nonlinear model. *EURASIP Journal on Advances in Signal Processing*, 2008, 2008.
- [21] G. Forbin and T. Vlachos. Nonlinear Flicker Compensation for Archived Film Sequences Using Motion-Compensated Graylevel Tracing. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(6):803–816, 2008.
- [22] P. Hayton, M. Brady, L. Tarassenko, and N. Moore. Analysis of dynamic MR breast images using a model of contrast enhancement. *Medical Image Analysis*, 1(3):207–224, 1997.
- [23] R. Hummel. Image enhancement by histogram transformation. *Computer Graphics Image Processing*, 6:184–195, 1975.
- [24] Y. Jin, L. M. Fayad, and A. F. Laine. Contrast enhancement by multiscale adaptive histogram equalization. In A. F. Laine, M. A. Unser, & A. Aldroubi, editor, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 4478, pages 206–213, 2001.
- [25] S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High dynamic range video. *ACM Transactions on Graphics*, 22(3):319–325, 2003.
- [26] Y. Kim, A. Martínez, and A. Kak. Robust motion estimation under varying illumination. *Image and Vision Computing*, 23(4):365–375, 2005.
- [27] A. Levin, A. Zomet, S. Peleg, and Y. Weiss. Seamless image stitching in the gradient domain. *Lecture Notes in Computer Science*, pages 377–389, 2004.
- [28] T. Lindeberg. On the axiomatic foundations of linear scale-space: Combining semi-group structure with causality vs. scale invariance. In K. A. Publisher, editor, *Gaussian Scale-Space Theory*, pages 75–98, 1997.
- [29] J.-L. Lisani and J.-M. Morel. Detection of major changes in satellite images. In *International Conference on Image Processing (ICIP)*, pages 941–944, 2003.
- [30] J. Lu, H. D. M., and W. J. B. Contrast enhancement of medical images using multiscale edge representation. *Optical engineering*, 33(7):2151–2161, 1994.
- [31] H. Malm, M. Oskarsson, E. Warrant, P. Clarberg, J. Hasselgren, and C. Lejdfors. Adaptive enhancement and noise reduction in very low light-level video. In *IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007*, pages 1–8, 2007.
- [32] S. Mann, C. Manders, and J. Fung. Painting with looks: Photographic images from video using quantimetric processing. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 117–126. ACM New York, NY, USA, 2002.
- [33] V. Naranjo and A. Albiol. Flicker reduction in old films. In *International Conference on Image Processing (ICIP)*, volume 2, 2000.
- [34] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics (TOG)*, 23(3):664–672, 2004.
- [35] F. Pitié, R. Dahyot, F. Kelly, and A. Kokaram. A new robust technique for stabilizing brightness fluctuations in image sequences. In *2nd Workshop on Statistical Methods in Video Processing In conjunction with ECCV 2004*. Springer, 2004.
- [36] F. Pitié, B. Kent, B. Collis, and A. Kokaram. Localised deflicker of moving images. In *Proceedings of the 3rd IEE European Conference on Visual Media Production (CVMP'06)*, pages 134–143, 2006.
- [37] S. Pizer. Intensity mappings for the display of medical images. *Functional Mapping of Organ Systems and Other Computer Topics*, 1981.
- [38] S. Pizer, E. Amburn, J. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. Romeny, J. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.
- [39] P. Richardson and D. Suter. Restoration of historic film for digital compression: A case study. In *International Conference on Image Processing (ICIP)*, volume 2, 1995.
- [40] T. Saito, T. Komatsu, T. Ohuchi, and T. Seto. Image processing for restoration of heavily-corrupted old film sequences. In *15th International Conference on Pattern Recognition*, volume 3, pages 13–16 vol.3, 2000.
- [41] P. Schallauer, A. Pinz, and W. Haas. Automatic restoration algorithms for 35mm film. *J. Computer Vision Res*, 1(3):59–85, 1999.
- [42] J. Starck, F. Murtagh, E. Candes, and D. Donoho. Gray and color image contrast enhancement by the curvelet transform. *IEEE Transactions on Image Processing*, 12(6):706–717, June 2003.
- [43] J. Stark. Adaptive image contrast enhancement using generalizations of histogram equalization. *IEEE Transactions on Image Processing*, 9(5):889–896, May 2000.
- [44] D. Suter and P. Richardson. Historical film restoration and video coding. In *Proceedings of the Picture Coding Symposium*, pages 389–394, 1996.
- [45] R. Szeliski and H. Shum. Creating full view panoramic image mosaics and environment maps. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 251–258. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA, 1997.
- [46] M. Uyttendaele, A. Eden, and R. Szeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 2001.
- [47] P. van Roosmalen, R. Lagendijk, and J. Biemond. Correction of intensity flicker in old film sequences. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(7):1013–1019, 1999.
- [48] C. Villani. *Topics in optimal transportation*. American Mathematical Society, 2003.
- [49] T. Vlachos. Flicker correction for archived film sequences using a nonlinear model. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(4):508–516, 2004.
- [50] K. Wong, A. Das, and M. Chong. Improved flicker removal through motion vectors compensation. In *Third*

- International Conference on Image and Graphics*, pages 552–555, 2004.
- [51] C. Xiaohua, M. Brady, J. Lo, and N. Moore. Simultaneous segmentation and registration of contrast-enhanced breast MRI. In *Information Processing in Medical Imaging*, pages 126–137, 2005.