LOCALISED DEFLICKER OF MOVING IMAGES

F. Pitié[1], B. Kent[1], B. Collis[1] and A. C. Kokaram[2]

[1] The Foundry, London UK,
[2] Trinity College Dublin, Ireland
fpitie@mee.tcd.ie, (ben|bill)@thefoundry.co.uk, anil.kokaram@tcd.ie

Keywords: deflicker, *in-scene* flicker, video restoration

Abstract

Brightness fluctuations in sequences are a typical impairment in archive and modern footage. These fluctuations are mainly temporal but they may also present spatial variations across each individual frame. Current deflicker techniques consider that these variations are smooth but this paper shows that the flicker localisation can be very fine in some cases. This paper presents a deflicker algorithm that is able to handle within a same framework both very smooth spatial variations and very localised variations. The core of the method lies in the parametrisation of the flicker model with only one parameter per pixel. The viability of using such a fine parametrisation is made possible by employing a temporal filtering that effectively separates occlusions from the flicker estimation.

1 Introduction

Random fluctuation in the observed brightness of recorded image sequences, also called *flicker*, occur in a variety of situations. The most commonly observed instance of flicker is in archived film and video. It is caused by the degradation of the medium (ageing of the film stock), varying exposure times, or curious effects of poor standards conversion. Varying exposure time is common to hand-cranked footage, but also happens with mechanised cameras in early films or most recently with personal 8mm. Flicker also affects more modern films and video media if the lighting conditions are poor, as in submarines surveillance, or when the transfer from film to video (telecine) is not properly done.

Flicker artefacts remain a problem, even with the use of digital cameras. One frequent source of modern flicker comes from radiometric calibration issues. Consider for instance time-lapse sequences, where each film frame is taken at long interval rates, or sequences made from multiple cameras, like in the "Inbetweening" special effect used in "The Matrix" (1999). These kinds of sequences usually flicker when played back because each frame has been captured under different lighting conditions and calibration settings.

Flicker is considered as a global artefact as it affects the entire frame but frequently the brightness fluctuations also present some spatial variations across each frame. The variations are usually very smooth but may appear as localised structures, like the black diagonal on the right of frame n on figure 8. These structures are a typical manifestation of severe flicker on archive footage. An extreme case of localisation appears in modern footage in the case of *in-scene* flicker. This occurs when fluorescent lights are out of synchronisation with the acquisition rate and that the camera is not is the same place that the lights. If the fluorescent lights are inside a room for instance and the camera is outside the room, then only the door flickers whereas the rest of the scene is non-flickering. An example of this is shown in figure 7, where due to the complexity of the scene, the flicker requires a pixel-wise granularity.

Some work has been done on flicker removal, but only for very smooth spatial variations as it is the case in most archive footage. The aim of this paper is to propose a solution to the problem of localised flicker for both archive and *in-scene* cases. Dealing with the problem of flicker requires some attempt to model or measure the fluctuation between frames (the estimation process), and then to remove this fluctuation in some way (the correction process). After a brief overview of the state of the art deflicker algorithms, the following sections examine how to integrate extreme spatial variations by using a new flicker model and then how to remove the temporal fluctuations by considering a motion compensated filtering of the flicker parameters.

2 Deflicker Framework and Related works

Existing deflicker techniques consist of two stages: 1) the flicker model estimation and 2) the flicker compensation that aligns the brightness level.

Flicker Models. In the general sense, the flicker artefact between two images u and v can be modeled as a mapping t on the grayscale component that depends on the pixel position $\mathbf{x} = (x, y)$:

$$v(\mathbf{x}) = t(u(\mathbf{x}), \mathbf{x}) + \epsilon(\mathbf{x}) \qquad (1)$$

The outlier term $\epsilon(\mathbf{x})$ accounts for the image disparities due for instance to motion or missing data. The mapping at a particular pixel is usually modeled as being linear $t(u(\mathbf{x}),\mathbf{x}) = a(\mathbf{x})u(\mathbf{x}) + b(\mathbf{x})$ in modern footage [7, 4, 2] and non linear in old footage [5, 8, 1]. Ideally the mapping t should be estimated at every pixel site but this would require too much computation. Since the flicker artefact is spatially smooth, the solution adopted in the literature is to interpolate the mapping using 2D polynomial [7, 4], cosine [2] or spline [5] functions. Using splines for instance yields the following interpolation [5]:

$$v(\mathbf{x}) = \sum_i w(\mathbf{x} - \mathbf{x}^{(i)})\ t^{(i)}(u(\mathbf{x})) \tag{2}$$

where $\mathbf{x}^{(i)}$ is the 2D position of the i^{th} control point on the image, $t^{(i)}$ the estimated mapping at this control point and $w(\mathbf{x})$ the interpolating 2D spline.

The problem in estimating the flicker mapping is that not all of each image pair can be matched. The parts that cannot be matched are due to occlusions/uncovering because of motion or simply due to missing data (blotches/dirt/dropout) in the case of degraded film and video material. To cope with this problem, Roosmalen [7] and Yang [9] suggest detecting occluded areas based on spotting large intensity differences that cannot be explained by flicker alone. Parameter estimation is then performed only on the blocks in which there are no outliers detected. Estimates for the “missing blocks” are then generated by some suitable interpolation algorithm. Unfortunately, this method for detecting outliers fails in the presence of heavy flicker degradation.

Flicker Compensation. The second step is to find the mapping that needs to be applied on each frame to compensate for the fluctuations and thus align the brightness levels. This can be done by estimating the flicker between the current frame u_n and the last restored frame u^R_{n-1} and apply the mapping onto u_n. The brightness levels are then *locked* to match the levels of the first frame. To avoid error accumulations, Roosmalen [7] relaxes the brightness stabilisation by constructing the restored frame as a mixture of the locked frame $u^L_n(x, y)$ and the observed image $u_n(x, y)$:

$$u^R_n(x, y) = k\ u^L_n(x, y) + (1 - k)\ u_n(x, y) \tag{3}$$

where k is a forgetting factor usually set between 0.85 and 0.9. There is thus a trade-off between the amount of deflicker that can be handled and the propagation of errors in the restored sequence. The key solution to flicker compensation is actually to consider the problem as a filtering problem [2, 5]. Instead of locking the brightness to the previous frame, it should be sought to average the brightness levels between the current frame and its past and future neighbouring frames. The filtering idea [5] can be simplified as follows:

$$u^R_n(x, y) = \frac{1}{2T + 1} \sum_{i=-T}^{T} t_{n,n+i}(u_n(\mathbf{x}), \mathbf{x}) \tag{4}$$

where $t_{n,n+i}$ is the estimated flicker mapping that aligns the brightness levels of the current frame u_n to match the neighboring frame u_{n+i}. The number of neighbouring frames T can be up to 7 frames forwards and backwards. This compensation method results in more stable brightness alignment and is also less dependent on a perfect flicker estimation since the compensation depends on several estimations.

3 Flicker Model

The existing deflicker algorithms specifically target smooth instances of flicker and due to their limitations they cannot be scaled to a higher resolution. For non-parametric models, the limitation arises from the manipulation of local histograms [6, 5] which are not designed to be used at pixel resolution. For parametric methods, the limitation comes from the amount of reliable data required for the mapping estimations. Because the parametric models involve more than one parameter per mapping, the problem becomes under-determined on flat areas and intractable at pixel resolution. The generic solution proposed in [7] is to use the smoothness assumption and interpolate the mapping on these flat areas from more reliable neighbouring mappings. Sadly the complexity of *in-scene* flicker makes it impossible to establish a smoothness prior that would give enough constraint on the estimation. Thus instead of looking for a complicated smoothness prior, it would be beneficial to establish a flicker model with only one parameter per pixel. This is already the case in the purely linear case $v(\mathbf{x}) = a(\mathbf{x})\ u(\mathbf{x})$, but not for other more complex models. The idea is thus to propose an original flicker model that requires only one parameter per pixel but that is still able to handle non-linear flicker distortions.

3.1 Mixing Model For *in-scene* Flicker

A first solution is to consider that a pixel is only affected by a percentage of the original flicker source (see figure 2):

$$v(\mathbf{x}) = \alpha(\mathbf{x})\,(t_0(u(\mathbf{x})) - u(\mathbf{x})) + u(\mathbf{x}) \tag{5}$$

This model can be understood by assuming that in *in-scene* flicker, the lamp which is the source of flicker has a global impact of $t_0(u(\mathbf{x}))$, but that due to the scene geometry, a particular point can only receive a percentage $\alpha(\mathbf{x})$ of this light. The complexity of the problem is then dramatically reduced because the mapping estimation is done globally, whilst the local variations are modeled with only one parameter α. Thus, provided that the flicker can be derived from only one source, the problem comes down to the estimation of one parameter per pixel, which yields a fully determined problem.

3.2 Symmetric Mixing Model

Although intuitive, the mixing parameter α as it is presented needs a re-parametrisation because it describes a non-symmetrical problem. Consider for instance the case of figure 3 where frame n is corrected with respect to frame $n-1$ with a non-linear mapping $t_0 : u \mapsto u^2$. The inverse mapping, that corresponds to the flicker from $n+1$ to n, is $t_0^{-1} : u \mapsto \sqrt{u}$. The problem is that $u \mapsto \sqrt{u}$ cannot be derived as linear proportion of $u \mapsto u^2$. This means that restoring frame n from $n-1$ requires a different model than restoring frame $n-1$ from n. This asymmetry artificially increases the model complexity and is also a potential issue in the parameter filtering stage.

The key to the re-parametrisation is to realise that for any mapping t, both t and t^{-1} are reflected across the identity axis $u = v$. Consider then the following new mixing parameter γ that is measured perpendicularly to the identity axis $u = v$. This is illustrated on figure 4. Since the mixing is done perpendicularly to the identity axis, the way to express γ is to rotate the joint intensity space (u, v) by $\pi/4$. Denote as (q, r) the new intensity representations:

$$\left\{ \begin{array}{l} q = (u+v)/\sqrt{2} \\ r = (u-v)/\sqrt{2} \end{array} \right. \text{ and resp. } \left\{ \begin{array}{l} u = (q+r)/\sqrt{2} \\ v = (q-r)/\sqrt{2} \end{array} \right. \tag{6}$$

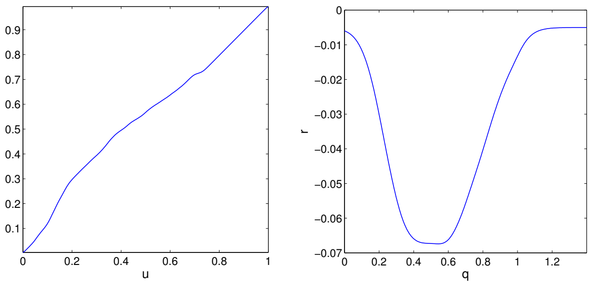
With this new intensity representations, the mapping $u \mapsto v = t(u)$ becomes $q \mapsto r = s(q)$ with $t(u) \equiv s(q)$. The parametrisation is these QR coordinates is then straightforward:

$$r(\mathbf{x}) = \gamma(\mathbf{x})\ s_0(q(\mathbf{x})) \tag{7}$$

where s_0 corresponds to the flicker characteristic function t_0 in the QR coordinates. The mixing parameter γ controls the flicker strength in a symmetric way. For $\gamma = 0$, there is no flicker, and $q \mapsto -\gamma s_0(q)$ corresponds to the inverse mapping of $q \mapsto \gamma s_0(q)$.

Thus within this new model, to estimate the flicker strength between two images u and v at a particular point $\mathbf{x}$, it is sufficient to convert $(u(\mathbf{x}), v(\mathbf{x}))$ into $(q(\mathbf{x}), r(\mathbf{x}))$ and then derives $\gamma(\mathbf{x})$ by

$$\gamma(\mathbf{x}) = \frac{r(\mathbf{x})}{s_0(q(\mathbf{x}))} \tag{8}$$



t_0 s_0

Figure 1: Example of real flicker characteristic map $t_0(u)$ and the corresponding $s_0(q)$. The mapping has been estimated on a image area by histogram matching over two frames.

3.3 Choice of the Flicker Mapping s_0

The shape of s_0 corresponds to the characteristic shape of the flicker mapping and can be derived from the literature models. In this paper, the flicker in modern footage is modeled by a purely linear mapping:

$$s_0(q) = a\,q \quad \text{which corresponds to} \quad t_0(u) = \frac{1-a}{1+a}\,u \tag{9}$$

For archive footage, the adopted solution is to ask the post-production artist to manually select on one frame an area where the flicker is spatially constant. The underlying mapping t_0 is then obtained by applying on this block the histogram transfer techniques presented in [6, 3, 5]. As displayed for a real flicker example on Figure 1, the characteristic flicker mapping s_0 is then derived from t_0 by using the QR transform. Because the nature of flicker does not vary across a shot, the same mapping s_0 can be used throughout the image sequence without any other adjustment.

3.4 Applying the Flicker Compensation

Once the flicker strength $\gamma(\mathbf{x})$ has been estimated for each pixel $\mathbf{x}$, the problem is to find the corresponding mapping $u \mapsto t_\gamma(u)$. In the QR system this mapping is simply given by $q \mapsto \gamma s_0(q)$, thus it remains to express this mapping back in the UV coordinates to obtain t_γ. This can be done by numerically inverting the QR transformation for γs_0. The inversion would however involve the linear interpolation of the whole mapping. Since this has to be done for every pixel, this is potentially time consuming. To avoid the complete evaluation of the mapping, an iterative solution is proposed. As observed on figure 5, for a particular pixel $\mathbf{x}$ with observed intensity u_0, the target value $v_0 = t_\gamma(u_0)$ corresponds to the intersection of the axis $u = u_0$ and the flicker mapping curve $q \mapsto \gamma s_0(q)$. Since the mapping t_γ is monotonic (see appendix), the

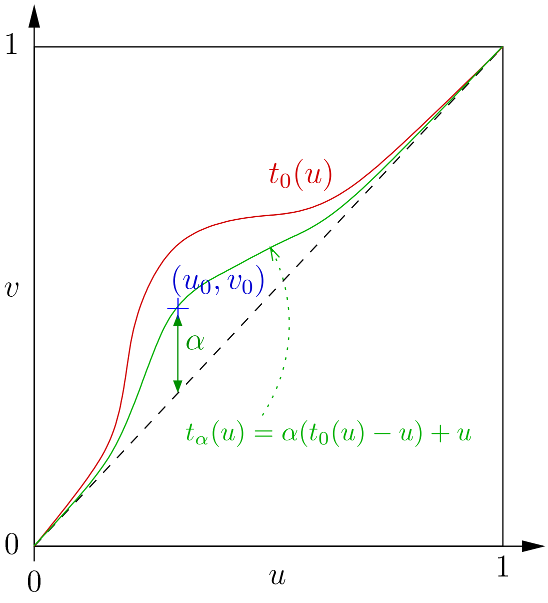


Figure 2: Mixing Model. At a pixel $\mathbf{x}$, the flicker mapping t_α (in green) is a linear combination of the identity (*i.e.* no flicker) and the flicker model s_0 (in red).

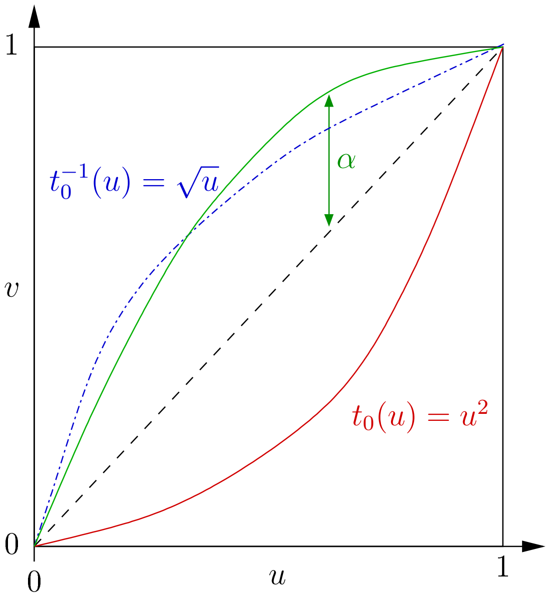


Figure 3: Mixing Model Asymmetry: t^{-1} cannot be obtained from t. This means that the flicker estimation from I_{n-1} to I_n requires a different model that the one from I_n to I_{n-1}.

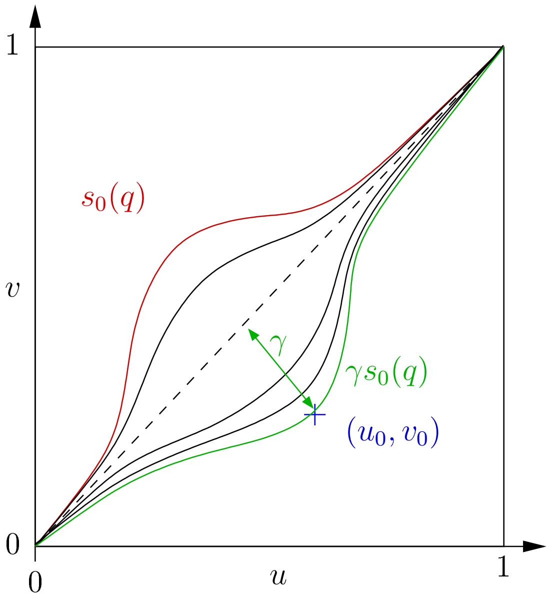


Figure 4: Symmetric Mixing Model. The flicker mapping is locally controlled by the flicker strength parameter γ.

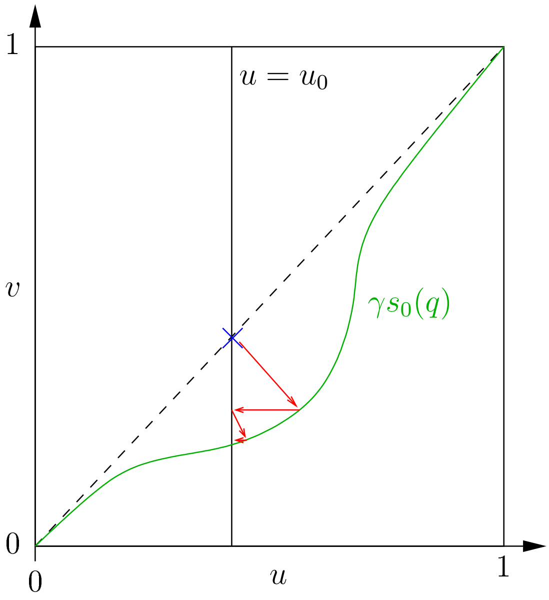
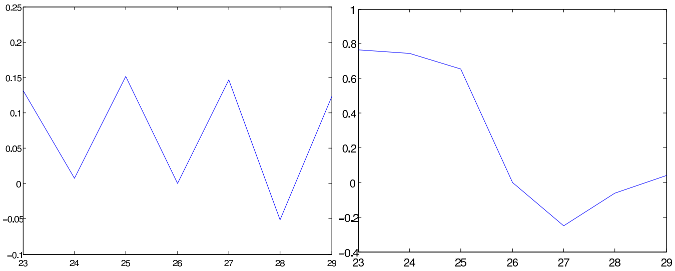


Figure 5: Flicker Compensation. Given u_0 and γ, the flicker mapping $t_\gamma(u_0)$ is found by iterative projection onto $u = u0$ and $q \mapsto \gamma s(q)$.



Flicker Instance

Occlusion Instance

Figure 6: Example of flicker parameter evolution for two different positions in the scene. On the left the pixel position is affected by severe flicker. On the right, an object is moving across the pixel. The flicker appears as a high frequency signal whereas the occlusion appears as a low frequency signal.

intersection can then be found by iteratively projecting on the flicker curve $q \mapsto s_q(q)$ and $u = u_0$. The procedure is as follows:

1. $(u_0, u_0) \mapsto (q, r)$
2. projection on $(q \mapsto \gamma s_0)$: $(q, r) := (q, \gamma r)$
3. projection on $(u = u_0)$: $(q, r) := (\sqrt{2}u_0 + q - r, \sqrt{2}u_0 - q + r)/2$
4. go to 2

4 Flicker Compensation

Drawing from previous works [2, 5], the flicker compensation is done here by filtering the flicker parameters. Relying on more than one estimation is actually critical in this case. The problem is that at a fine resolution, the flicker can also explain occlusions due to local motion. It is then very difficult to dissociate between flicker and occlusions without having to resort to complex estimation methods. This would also require the use of motion vectors, which are always delicate to manipulate, especially in this case because the motion estimation is deteriorated by the presence of flicker. But by using a filtering approach, the dissociation between flicker and occlusions can be handled by the temporal filter. From a temporal point of view, the flicker artefact is an impulsive noise whereas an occlusion has a causal behaviour (before and after occlusion). This is apparent on Figure 6 where the variations of γ have been plotted for two pixel locations. Hence using a low pass filter yields an implicit flicker removal.

The compensation scheme is thus as follows. For the current frame, the first step is to estimate the flicker parameters that match the neighbouring frames levels. This is done using equation (8), without any outlier treatment. The flicker resolution is then reached by smoothing the parameter fields by the apropriate 2D spatial Gaussian filter. The flicker parameters are then filtered temporally using a low-pass temporal filter. The resulting mapping is applied to the current frame in order to align its levels with the neighbouring frames. The resulting sequence is thus deflickered. The filtering process can handle most of the motion and the results are often acceptable for low resolution flicker. In case of pel-resolution flicker, the results can be blurred on areas where fast motion occurs.

The optional second stage of the compensation scheme starts with performing a motion estimation on this deflickered sequence. In that way, the motion vectors are not degraded by the presence of flicker and only marginally degraded by the blurring. These vectors are then used to register the frames before estimating again the flicker strength γ and filtering the registered parameters. Deflickering the original sequence, but this time by using registered frames, prevents from blurring the images.

5 Results

***in-scene* Sequence.** The overall method has been tested on 2 challenging sequences. The first one is a modern footage with *in-scene* flicker (see Figure 7). The restoration is performed using a purely linear model as described in equation 9, at a 3×3 pixel resolution and using motion registration as described in the compensation paragraph.

A common evaluation of deflicker results is to trace the evolution of the global mean intensity. The intensity curve is spiky in presence of flicker, thus a smoothness of the curve gives an indication on the deflicker improvements. This is however only realistic for relatively clean sequences and not for archive sequences. The average mean is traced in Figure 9 for the *in-scene* sequence. The original sequence corresponds to the blue plot, the proposed results to the green one and Roosmalen's results to the red dashed plot. It is visible that Roosmalen's compensation tends to drift after a while, whereas the filtering approach offers a better lock. To evaluate the restoration quality at a local scale, the intensity is measured on a small area of the images and plotted on Figure 10. The graph confirms that in contrast with Roosmalen's method, the proposed method is able to stabilise the brightness levels even on small areas.

It is however well acknowledged that this assessment method cannot a replace visual control. The compact representation proposed on Figure 11 allows such an evalution. The i^{th} column of this time-compressed representation corresponds to the one dimensional projection of frame i on the y axis. This representation has the advantage of displaying both the temporal intensity evolution and some degree of spatial granularity. On the figure, the original sequence is on the left. The presence of flicker is visible on areas where the brightness changes from column to column. The deflickered sequence in the middle displays as expected visibly smoother temporal variations. The corresponding flicker strength γ is represented on the right.

Archive Sequence. The second sequence is a severely degraded archive sequence from *Rory O'More* (1911). This archive suffers from very fine localised flicker (see Figure 8). Although it has been shown in [5] that this sequence can most of the time be effectively deflickered using histogram matching techniques, some of the shots display localised flicker structures that cannot be removed by existing methods. The restoration is done here by considering a non-linear model as described in paragraph 3.3. The flicker resolution is of 30×30 pixels, and no motion compensation has been considered.

The visual quality of the improvements can be assessed on the time-compressed representations on Figure 12 and as expected the sequence is effectively deflickered. To better analyse the deflicker restoration, Figure 13 shows some results on three consecutive frames affected by a very localised flicker. The diagonal black band, visible on the original frames on the left, has been properly removed in the result frames in the middle. The diagonal structure is actually clearly visible in the flicker strength γ field on the right. The method offers then a substantial improvement to existing methods like [5] since, as seen on figure 13, the diagonal structure cannot be removed when using this method [5]. To better appreciate the difference, an enhanced view of the difference between both restored versions is shown on the right.

6 Conclusion

This paper presents a new model for flicker. Because the model requires only one parameter per pixel, it is possible to consider, within the same framework, both spatially smooth as well as very fine variations. The framework can even consider a pixel-wise granularity in the flicker modeling. The viability of the deflicker algorithm is given by the filtering stage which separates the flicker signal from occlusions due to local motion. The overall method can handle severe cases of flicker localisation that no existing method could restore.

It should also be noted that the proposed method offers a generic framework, which would allow other flicker models [7, 8, 5] to be incorporated.

Acknowledgements

This project has been funded by the European Community's Sixth Framework Programme under the Marie Curie Action MTKI-CT-2005-029564

A Proof of Convergence for the Compensation Iterations.

Recall that the procedure is to iteratively find the intersection with the flicker mapping t_γ and then project on $u = u_0$. At iteration n, denote as (u_0, v_n) the projection on $u = u_0$. The first step is to project this point parallely to the second diagonal onto the flicker mapping t_γ. Denote as $(u'_n, v'_n = t_\gamma(u'_n))$ this point. The point is then projected onto $u = u_0$ thus $v_{n+1} = v'_n$. Consider the case where $t_\gamma(u_0) < u_0$. The intersection occurs then for $u'_n > u_0$ and it holds also that $v'_n = v_{n+1} < v_n$. Because $t'_\gamma > 1$, it follows that $t_\gamma(u_0) < t_\gamma(u'_n) = v_{n+1} < v_n$. Thus the sequence $\{v_n\}_n$ converges. The same results holds if $t_\gamma(u_0) > u_0$.

References

[1] G. Forbin, T. Vlachos, and S. Tredwell. Flicker Compensation for Archived Film Using a Spatially-Adaptive Non-Linear Model. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, may 2006.

[2] A. C. Kokaram, R. Dahyot, F. Pitié, and H. Denman. Simultaneous luminance and position stabilization for film and video. In *Visual Communications and Image Processing (VCIP)*, San Jose, California USA, January 2003.

[3] V. Naranjo and A. Albiol. Flicker reduction in old films. In *Proceedings of the 2000 International Conference on Image Processing (ICIP'00)*, volume 2, pages 657–659, September 2000.

[4] T. Ohuchi, T. Seto, T. Komatsu, and T. Saito. A robust method of image flicker correction for heavily-corrupted old film sequences. In *Proceedings of the 2000 International Conference on Image Processing (ICIP'00)*, September 2000.

[5] F. Pitié, R. Dahyot, F. Kelly, and A. C. Kokaram. A new robust technique for stabilizing brightness

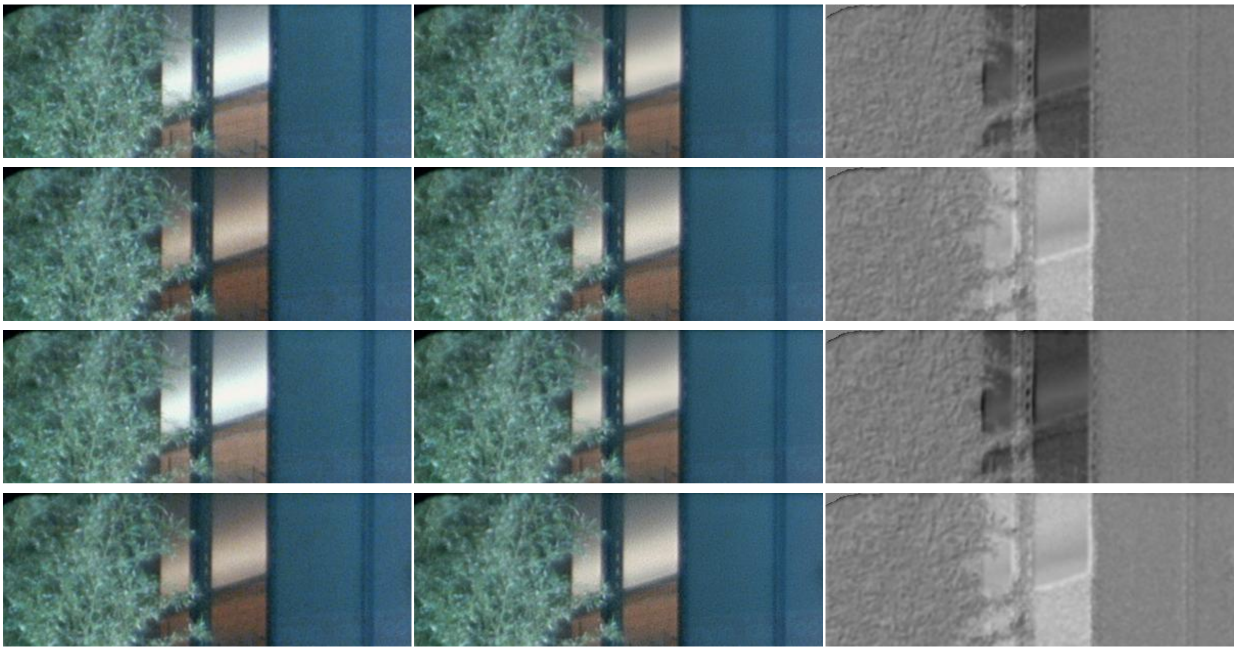


Figure 7: Example of *in-scene* flicker localisation due to out of sync fluorescent lighting. On the left, the original sequence, on the middle, the deflickered sequence and on the right the corresponding flicker map (see section 3.2). A gray value for γ corresponds to $\gamma = 0$, *i.e.* no flicker. The scene geometry is too complex to consider the flicker as having smooth variations.

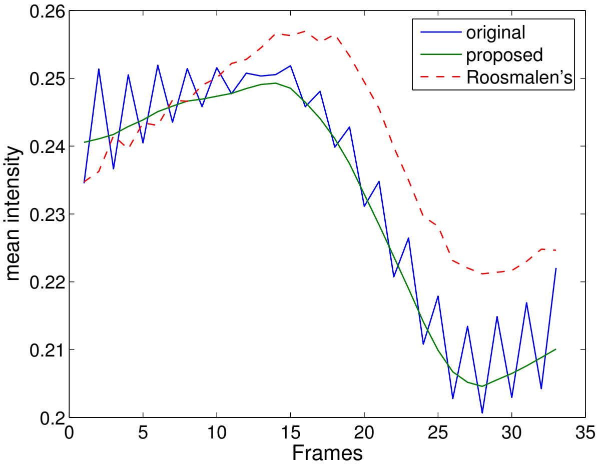


Figure 9: Global Mean Intensity Comparison for *inscene* deflicker Results. The plot of the global mean gives an indication of the flicker level. It is visible that Roosmalen's method (in dashed red) tends to drift after a while. In comparison the proposed method has a stable behaviour.

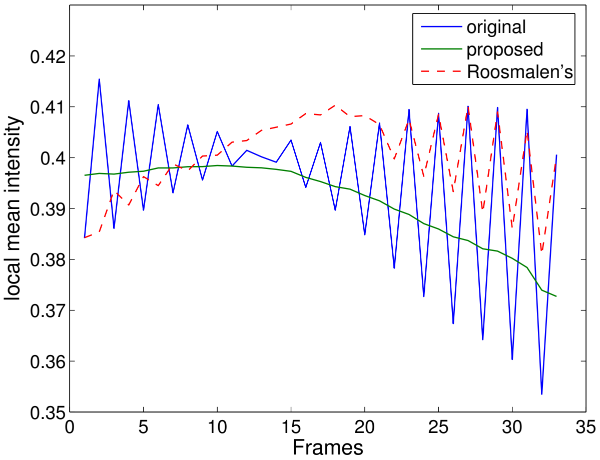


Figure 10: Local Mean Intensity Comparison for *inscene* deflicker Results. The mean intensity is done on a small area of the frames. Roosmalen's method (in dashed red) cannot cope with the extreme localisation. In comparison the proposed deflicker method results in smooth temporal variations, which is a sign of a better deflicker.

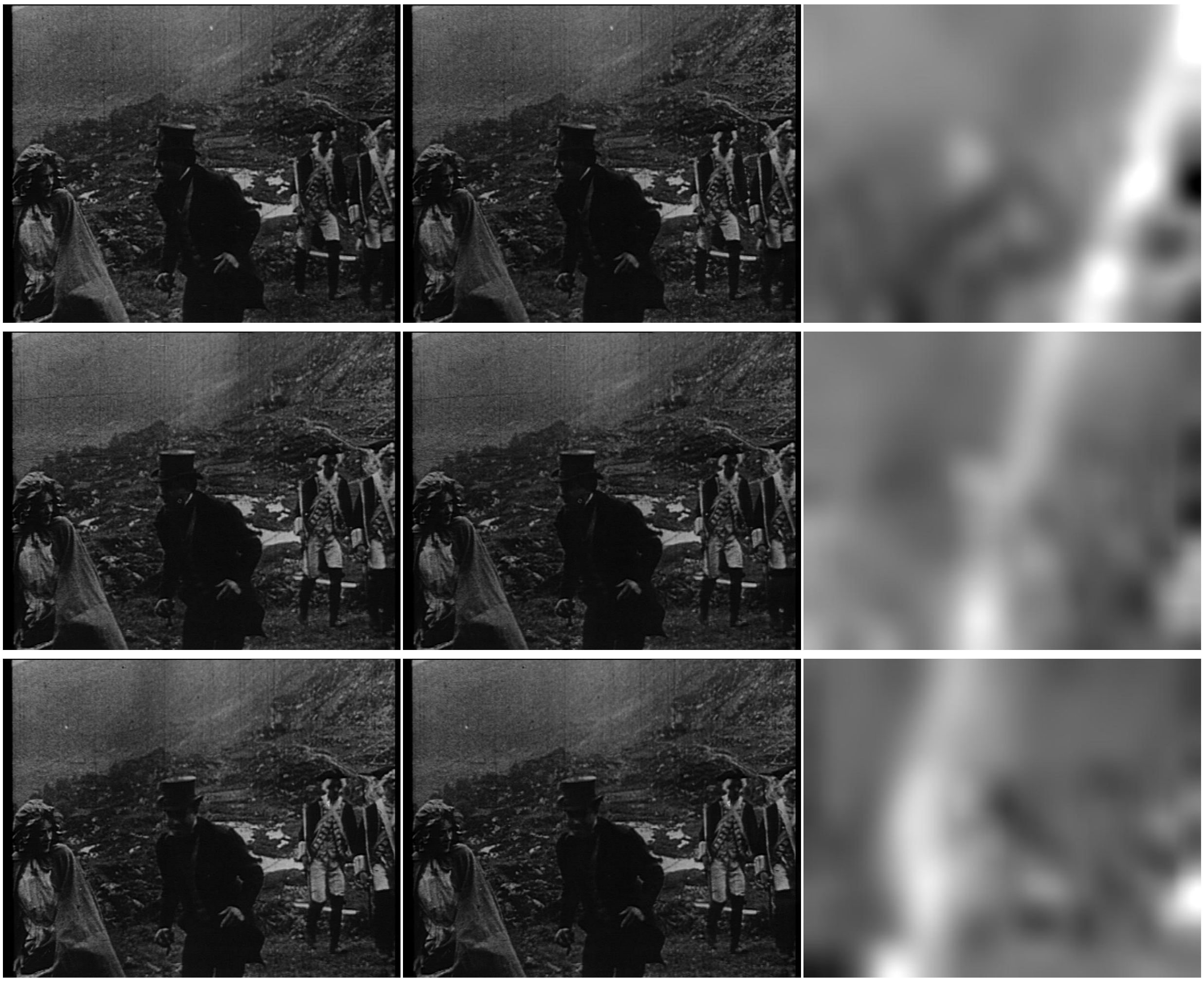


Figure 8: Example of strongly localised archive flicker. On the left, the original sequence, on the middle, the deflickered sequence and on the right the corresponding flicker map (see section 3.2).

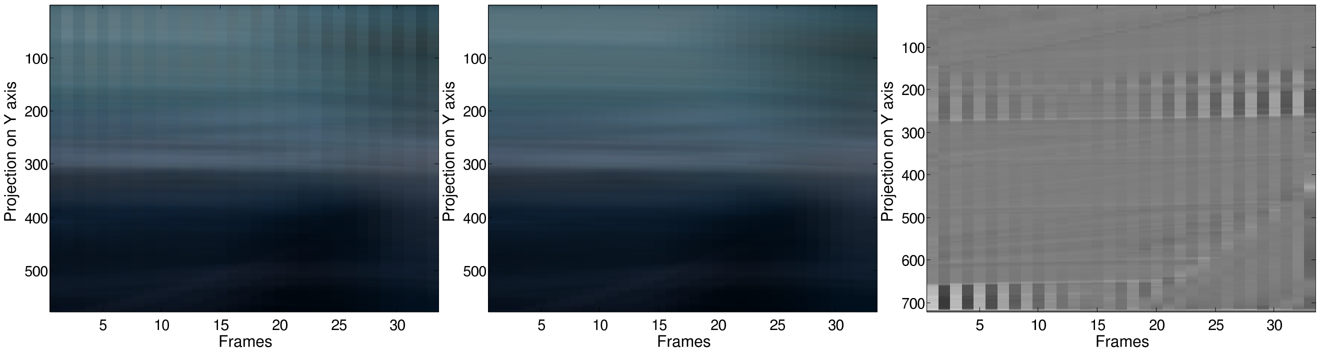
fluctuations in image sequences. In *2nd Workshop on Statistical Methods in Video Processing In conjunction with ECCV 2004 Prague*, Prague, Czech Republic, may 2004.

[6] P. Schallauer, A. Pinz, and W. Haas. Automatic Restoration Algorithms for 35mm Film. *Videre: Journal of Computer Vision Research*, 1(3):60–85, summer 1999.

[7] P.M.B. van Roosmalen, R.L. 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.

[8] T. Vlachos. Flicker Correction for Archived Film Sequences. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(4):508–516, april 2004.

[9] X. Yang and N. Chong. Enhanced approach to film flicker removal. *Proceedings of SPIE Applications of Digital Image Processing XXIII*, 4115:39–47, 2000.

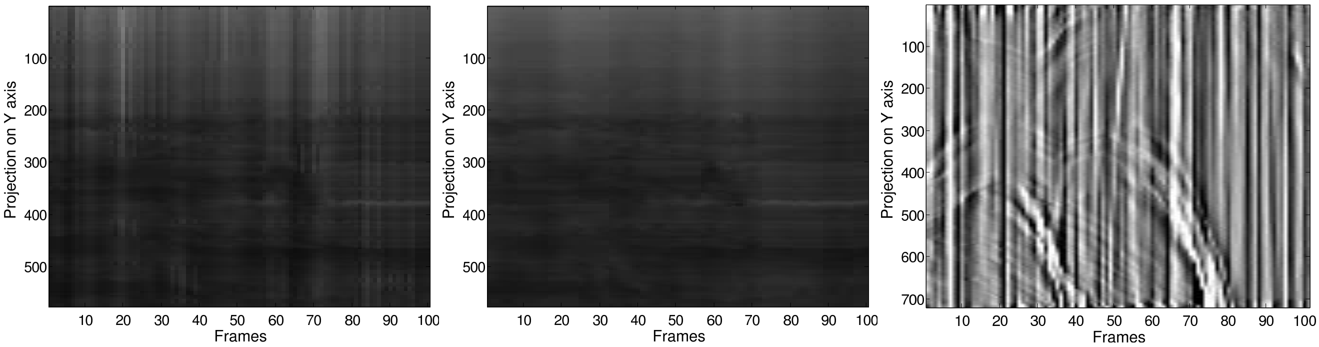


Original

Deflickered

Flicker Strength γ

Figure 11: Flicker Removal Results for *inscene* flicker. Each frame is represented by its horizontal projection on the vertical axis. The results on the middle clearly show that the brightness has been stabilised. The flicker map γ is displayed on the right.

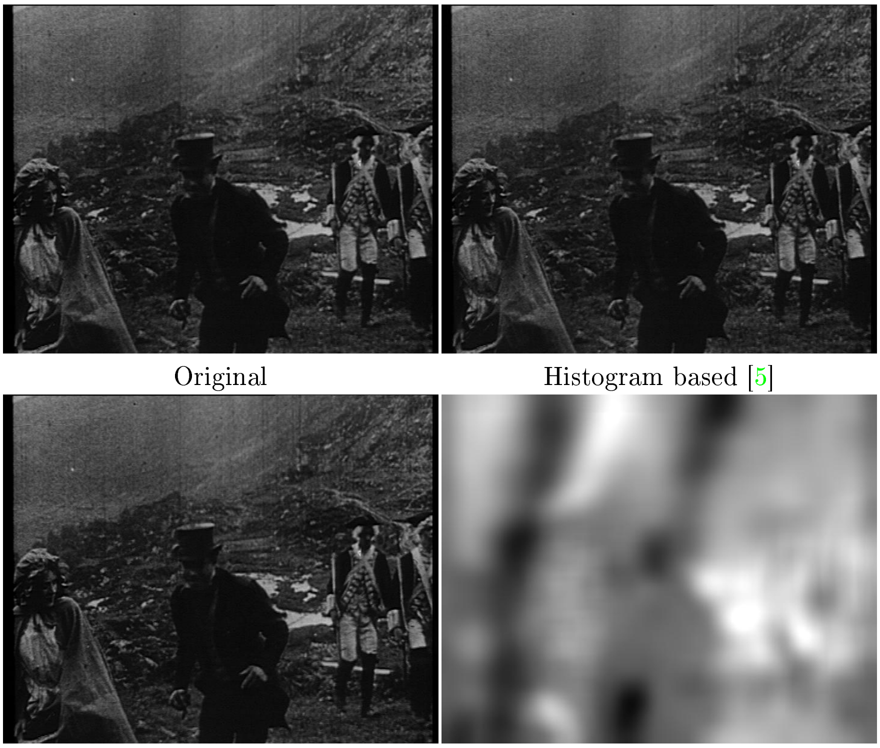


Original

Deflickered

Flicker Strength γ

Figure 12: Flicker Removal Results for archive flicker. Each frame is represented by its horizontal projection on the vertical axis. The results on the middle clearly show that the brightness has been stabilised. The flicker map γ is displayed on the right.



Original

Histogram based [5]

proposed method

restoration difference

Figure 13: Flicker Removal Comparison with the method described in [5]. On the top left the original frame. On the top right the frame restored by the histogram based technique of [5] at order 6. On the bottom right the restored frame using this method. The difference between both restorations is displayed on the bottom right (the difference has been enhanced to help the reader).