Simultaneous luminance and position stabilization for film and video

Anil C. Kokaram[*], Rozenn Dahyot, François Pitié and Hugh Denman

Electronic and Electrical Engineering Department, Trinity College, Dublin 2, IRELAND

ABSTRACT

Temporal and spatial random variation of luminance in images, or "flicker", is a typical degradation observed in archived film and video. The underlying premise in typical flicker reduction algorithms is that each image must be corrected for a spatially varying gain and offset. These parameters are estimated in the stationary region of the image. Hence the performance of that algorithm depends crucially on the identification of stationary image regions. Position fluctuations are also a common artefact resulting in a random "shake" of each film frame. For removing both, the key is first to extract global motion in the presence of unknown local motions. Parameters are then estimated on that part of the image undergoing the global or dominant motion. A new algorithm that simultaneously deals with global motion estimation and flicker is presented. The final process is based on a robust application of weighted Least-squares, in which the weights also classify portions of the image as local or global. The paper presents results on severely degraded sequences showing evidence of both Flicker and random shake.

Keywords: Film restoration, Global Motion, Robust Estimation, Flicker

1. INTRODUCTION

The automated restoration of image sequences is gaining in importance with the continuing rise of digital visual media. Companies such as Philips, Thompson, Sony and Snell and Wilcox, all produce some kind of automated restoration device, some of which operate in real time. Global degradation in archived image sequences are the most visibly displeasing of all the artefacts that can occur. Noise, brightness flicker and image "shake", are all examples of this class of degradation. Brightness flicker could be caused by differing exposure times of each film frame, while shake (position instability) can be caused by unwanted camera motion or by misalignment of film frames in some conversion process. It is only after these degradations are removed that local defects like Dirt and Sparkle, Film Tear, Line Scratches and so on, become more visible. Typically, the restoration operator would seek to remove global defects and then remove the local defects.

In a sense, the removal of global defects is simpler than treating local defects. This is because in treating local defects, it is possible to introduce visible distortion that is more displeasing because the distortion is different from the rest of the picture. Distortion introduced by correcting global defects causes less of a contrast within the frame since the entire frame is generally affected in the same way. Also, in treating global defects, especially shake and brightness flicker, the algorithm can take advantage of information from a large portion of the picture. This is not the case with local defects like Dirt and Sparkle in particular.

Several authors have dealt with the problem of position stabilisation (shake)[1–3] and flicker removal[4,5] in the past. In general, shake reduction is performed by estimating the global motion of each image, and compensating for the unwanted component. Global motion estimation is therefore a key technology in this problem. Flicker removal is performed similarly, by estimating the brightness change between the image frames and then compensating this change in some way. However, flicker reduction algorithms generally assume that it is possible to find correspondences between image frames. Shake removal processes generally assume that the brightness of objects or the scene illumination does not change much with time. In archived footage, flicker can occur in connection with significant scene motion and this makes it difficult to achieve a restoration in severe cases using current technology.

[*] akokaram@tcd.ie

Figure 1. Top: Frames 0,10,20 of the Tunnel sequence. Bottom row: Frames 10, 15, 20 of part of the silent film *Rory O'More*. It is difficult to visualise flicker from these stills, however the flicker in the *Tunnel* sequence is less extreme than for *Rory*.

This paper therefore considers the problems of flicker and shake reduction together. A new formulation of the flicker problem is presented together with a slightly different formulation of a global motion estimation process. The paper considers how the two processes could function together; firstly as a two stage cascade and secondly within a unified approach.

2. GLOBAL MOTION ESTIMATION

Consider a translational image sequence model as follows.

$$I_n(\mathbf{x}) = I_{n-1}(\mathrm{A}_{n,n-1}\mathbf{x} + \mathbf{d}_{n,n-1} + \mathbf{d}'_{n,n-1}(\mathbf{x})) + \epsilon(\mathbf{x}) \tag{1}$$

where $I_n(\mathbf{x})$ is the grey level of the pixel at the location given by position vector $\mathbf{x}$ in the frame n. The motion between the two frames n and $n-1$ at site $\mathbf{x}$ in frame n is given by $\mathrm{A}_{n,n-1}\mathbf{x} + \mathbf{d}_{n,n-1} + \mathbf{d}'_{n,n-1}(\mathbf{x})$. This is made up of a global translational component $\mathbf{d}_{n,n-1}$, a local motion component due to the local motion of objects $\mathbf{d}'_{n,n-1}(\mathbf{x})$ and an affine warping component $\mathrm{A}_{n,n-1}\mathbf{x}$. This affine warp allows the model to cope with the *stretching* or warping of film in severe cases. For the moment, we assume that all of the global motion components are unwanted. Therefore, in this model, the *true* underlying image sequence is obtained by estimating A, $\mathbf{d}$ and shifting each image to compensate.

For ease of notation, define the vector function $F(\mathbf{x}, \Theta) = \mathrm{A}_{n,n-1}\mathbf{x} + \mathbf{d}_{n,n-1}$. with $\Theta = [\mathrm{A}_{n,n-1},\ \mathbf{d}_{n,n-1}]$ is the vector formed by the motion parameters. using a 2×2 matrix of components for $\mathrm{A}_{n,n-1}$, the global motion component can cope with zoom, rotation, stretch, skew and can be written as follows.

$$\begin{aligned} F(\mathbf{x}, \Theta) &= \mathrm{A}_{n,n-1}\mathbf{x} + \mathbf{d}_{n,n-1} \\ &= \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} d_x \\ d_y \end{pmatrix} \\ &= \begin{pmatrix} a_1 x + a_2 y + d_x \\ a_3 x + a_4 y + d_y \end{pmatrix} = \mathrm{B}(\mathbf{x})\ \Theta \end{aligned} \tag{2}$$

where

$$\mathrm{B}(\mathbf{x}) = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{pmatrix}$$

The parameter vector is $\Theta = [a_1, a_2, d_x, a_3, a_4, d_y]^T$ where $\{a_i\}$ are the matrix coefficients of $\mathrm{A}_{n,n-1}$ and $\{d_i\}$ are the vector coefficients of the translational motion component $\mathbf{d}_{n,n-1}$.

Consider for the moment, that the local motion component $\mathbf{d}'_{n,n-1}(\mathbf{x})$ is zero. To solve the motion equation with respect to Θ, a typical approach is to minimize a function of the error $\epsilon(\mathbf{x}) = I_n(\mathbf{x}) - I_{n-1}(F(\mathbf{x}, \Theta))$ with respect to Θ. Unfortunately the image function $I_{n-1}(\cdot)$ is not well defined and generally non-linear. Given an existing estimate of Θ, say $\hat{\Theta}_i$, it is possible to generate an update $\delta\Theta$ such that $\Theta = \hat{\Theta}_i + \delta\Theta$, by linearising the residual equation around the current estimate using a Taylor series expansion as follows:

$$\begin{aligned} & I_n(\mathbf{x}) - I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) \\ & = \nabla I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) \cdot \mathrm{B}(\mathbf{x}) \cdot \delta\Theta + \varepsilon(\mathbf{x}) \end{aligned} \tag{3}$$

$\epsilon(\mathbf{x})$ and the higher order terms of the expansion are lumped together in the new residual $\varepsilon(\mathbf{x})$ linear with respect to the update $\delta\Theta$. The ∇ operator is the usual multidimensional gradient operator.

2.1. Maximum likelihood estimation

The maximum likelihood estimation consists in computing the motion parameter that maximizes the likelihood of the residuals across the entire image: $\hat{\Theta} = \arg\max_{\Theta} \{\mathcal{P}(\{\varepsilon(\mathbf{x})\}_{\mathbf{x}}) = \mathcal{P}(\boldsymbol{\varepsilon})\}$. Assuming the distribution of the residual ε is Gaussian, the algorithm is identical to least squares estimation and can be written as:

```
Do
    $\widehat{\delta\Theta} = \arg\min_{\delta\Theta}\{\mathcal{J}(\boldsymbol{\varepsilon}) = \sum_{\mathbf{x}}[\varepsilon(\mathbf{x})]^2\}$
    $\hat{\Theta}^{(i+1)} = \hat{\Theta}^{(i)} + \widehat{\delta\Theta}$
Until convergence at final step $\boldsymbol{i}$ $(\hat{\Theta} = \hat{\Theta}^{(\boldsymbol{i})})$
```
(4)

At each step i, $\widehat{\delta\Theta}$ is estimated by Least Squares:

$$\widehat{\delta\Theta} = [\mathrm{G}^{(i)\ T}\mathrm{G}^{(i)}]^{-1}\mathrm{G}^{(i)\ T}\mathbf{z}^{(i)} \tag{5}$$

where $\mathbf{z}^{(i)}$ is the vector collecting the values $\{I_n(\mathbf{x}) - I_{n-1}(\mathrm{B}(\mathbf{x})\ \hat{\Theta}^{(i)})\}_{\mathbf{x}}$ and $\mathrm{G}^{(i)}$ is the matrix collecting the values $\{\nabla I_{n-1}(\mathrm{B}(\mathbf{x})\ \hat{\Theta}^{(i)}) \cdot \mathrm{B}(\mathbf{x})\}_{\mathbf{x}}$.

2.2. Robust M-estimation

Unfortunately, the local motion component is never zero in most interesting sequences. In other words, there is always something moving around in the foreground or background. The estimator outlined above, will necessarily be biased by these local effects. The task is to find a simple mechanism to remove this local effect. This bias generally manifests as outliers in the residual error surface over the image. In other words, the motion estimation algorithm described above, when applied to the whole image, tends to lock onto the motion that is most prevalent over the largest area of the image, but the regions showing local motion "pull" the estimator away from the global effect. M-estimators[6] are now widely used to solve this problem and perform robust estimation of global motion parameters.[7–9] The underlying assumption is that the probability density function of the residuals is no longer Gaussian, and can be written as:

$$\mathcal{P}(\boldsymbol{\varepsilon}) \propto \exp\left[-\frac{1}{2}\sum_{\mathbf{x}} \rho\left(\frac{\varepsilon(\mathbf{x})}{\sigma_\rho}\right)\right] \tag{6}$$

Several functions ρ, convex or non-convex, have been proposed in the literature.[6,7,9,10] σ_ρ is the *scale parameter* that controls the limit where the influence of the outliers begins to decrease.[6] This parameter is fixed offline[7] but to simplify, it is set to 1 in section 4. The corresponding energy to minimize is non-quadratic and two estimation algorithms are reviewed in section 4.

3. FLICKER PARAMETER ESTIMATION

Consider that there is no motion between successive image frames. An image sequence affected by flicker shows fluctuations in intensity between frames that are spatially varying but of low frequency, and temporally varying at a high frequency. One popular method to model the flicker effect is to consider the following linear relation between two successive images in the sequence[4, 11]:

$$\mathrm{I}_{n+1} = a \cdot \mathrm{I}_n + b \tag{7}$$

with a and b spatially constant for each frame. Then the estimation of parameters a and b can be performed by comparing the maximum, minimum and mean of the gray values in the I_{n+1} and I_n,[11] or by comparing the moments (mean and variance) of their grey-level histograms.[4, 5, 12]

Naranjo[13] considered that the process causing flicker was non-linear. Thus given some non-linear function $f(\cdot)$ the model is

$$f\left(\mathrm{I}_n\right) = \mathrm{I}_{n+1} \tag{8}$$

A restoration was performed by using standard the method of cumulated histogram matching.[13, 14] This results in a non-linear equalisation of histograms using previous frames as references.

In order to deal with the spatially varying nature of flicker, two approaches have been proposed.

Blocks To deal with spatial fluctuations of the flicker, Roosmalen[4] split the image into small overlapping blocks. As the flicker is assumed to be spatially low frequency, it is possible to use equation 7 on each blocks. But the estimation of parameters a and b may be not reliable on some blocks, if the variance of the signal is null for instance or if some events appeared in blocks such as blotches, local motion or outliers. In this case, the parameters are rejected and replaced by an interpolation of more reliable neighbouring blocks. Once parameters a and b have been found for all blocks parameters are interpolated to the pixel scale using bilinear interpolation.

Robust Formulation Roosmalen's estimator is sensitive to outliers due to, for instance, local motion and blotches. Ohuchi[5] improved Roosmalen's method in considering a robust estimation scheme:

$$\mathrm{I}_{n+1}(\mathbf{x}) = a(\mathbf{x}, \mathbf{a}) \cdot \mathrm{I}_n(\mathbf{x}) + b(\mathbf{x}, \mathbf{b}) + \varepsilon(\mathbf{x}) \tag{9}$$

with $a(\mathbf{x}, \mathbf{a})$ and $b(\mathbf{x}, \mathbf{b})$ (second order) polynomial functions with respect to the position vector $\mathbf{x}$. The Flicker parameter vector Θ^f contains the coefficients of the gain function $\mathbf{a}$ and the offset function $\mathbf{b}$. It is estimated in minimizing the following robust energy function:

$$\mathcal{J}(\varepsilon) = \sum_{\mathbf{x}} \rho\left(\frac{\varepsilon(\mathbf{x})}{\sigma_\rho}\right) \tag{10}$$

The maximum likelihood estimation of the flicker parameters is performed by minimizing the energy $\mathcal{J}$. This problem is similar to the one presented in section 2.2. Section 4 presents the algorithms used to compute the estimate.

3.1. The New Robust Flicker Estimation Scheme

Following Ohuchi, we use robust cost functions to reduce the effect of outliers due to local motion or blotches. Cauchy's function ρ is used as the weighting functional: $\rho(\varepsilon(\mathbf{x})) = \frac{1}{2}\log\left(1 + \varepsilon(\mathbf{x})^2\right)$. The scale parameter σ_ρ controls the value over which errors are considered as outliers. It is changed at each step of the estimation, starting from $\sigma_\rho = 30$ (the flicker can cause intensity shifts of more than 20 levels of gray intensity over 256 gray levels), to $\sigma_\rho = 3$ (corresponding to the noise level in our case). The scale parameter is divided by 1.5 at each step.

3.1.1. Choosing the basis

Adopting a Cosine basis instead of polynomial yields a richer set of functions that can be modelled, as well as a lower complexity since the basis is orthogonal (see section 4). The qualities of the estimates using both bases have been assessed using 100 frames from *Rory O'More.* Both estimations had been carried out for a fixed number of iterations and at different orders of the basis (up to order 12). The figure 2 shows the mean of the ratio between error energies using cosine and polynomials. We can see that using a cosine basis gives lower error energy than using a polynomial basis. This does not imply that the underlying model of flicker is cosine rather than polynomial. However, results are visually better with the cosine basis.[15] Considering the possible computational advantages, the algorithm proposed here adopts a cosine basis as follows.

$$\phi_{pq}(m,n) = \alpha_p \alpha_q \cos\left(\frac{\pi p(2m+1)}{2M}\right) \cos\left(\frac{\pi q(2n+1)}{2N}\right)$$

$$\text{where}\alpha_q = \begin{cases} 1/\sqrt{N} & q = 0 \\ \sqrt{2/N} & 1 \le q \le N-1 \end{cases}$$

$$\text{and}\alpha_p = \begin{cases} 1/\sqrt{M} & p = 0 \\ \sqrt{2/M} & 1 \le p \le M-1 \end{cases}$$

Here, $\phi_{pq}(m,n)$ is the p,qth basis over an 'image' of size $M \times N$. Thus the spatial model for the gain causing the flicker is given by $a(\mathbf{x}) = a(i,j) = \sum_{p,q} \alpha_{p,q}\phi_{pq}(i,j)$ where $\alpha p, q$ are the coefficients to be estimated (denoted $\mathbf{a}$ previously. A similar function exists for $b(\mathbf{x})$ given coefficients $\beta_{p,q}$.

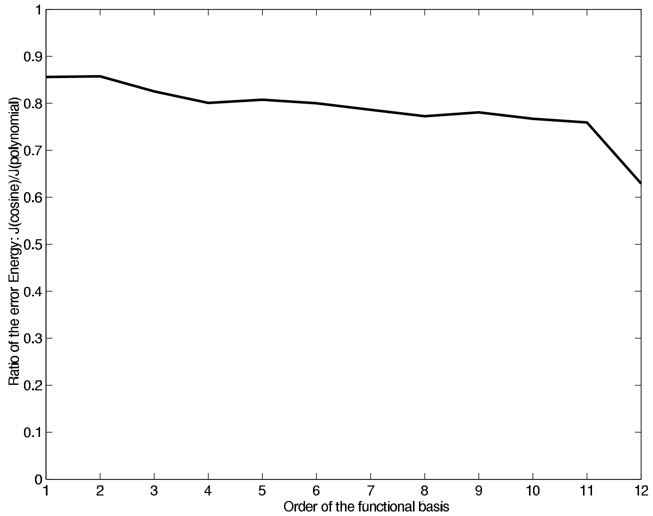

Figure 2. Mean of the ratio of the error energy ($\mathcal{J}(cosine)/\mathcal{J}(poly)$) vs. order of the basis.[15]

3.1.2. Regression limit

In previous flicker reduction algorithms, there is the problem that as the restoration process proceeds, the flicker estimation becomes progressively worse. This can be explained by considering that the estimation of the flicker parameter is expressed as a regression problem. Considering the linear global model (cf. equation 7), two formulations could be considered using the following errors:

$$\begin{cases} \varepsilon_1(\mathbf{x}) & = \mathrm{I}_n(\mathbf{x}) - a_1 \cdot \mathrm{I}_{n+1}(\mathbf{x}) - b_1 \\ \varepsilon_2(\mathbf{x}) & = \mathrm{I}_{n+1}(\mathbf{x}) - a_2 \cdot \mathrm{I}_n(\mathbf{x}) - b_2 \end{cases} \tag{11}$$

They lead to two different solutions unless the correlation between the random variables I_n and I_{n+1} is 1.[16,17] This implies that the estimation will be biased. In particular, the gain a will be usually be less than 1. To fix this problem, another possible formulation is to estimate the parameters using the Principal components of the cloud of points $(\mathrm{I}_n(\mathbf{x}), \mathrm{I}_{n+1}(\mathbf{x}))$.[18] But to use the robust PCA,[19] to avoid the effect of outliers in their estimation, would increase the computation time.

As an alternative, we propose to change the model in order to make the problem symmetric. Two symmetric estimators can then be written as follows:

$$\begin{cases} \varepsilon_1(\mathbf{x}) = (\mathrm{I}_{n+1}(\mathbf{x}) - \mathrm{I}_n(\mathbf{x})) - u_1 \cdot \frac{1}{2}(\mathrm{I}_{n+1}(\mathbf{x}) + \mathrm{I}_n(\mathbf{x})) - b_1 \\ \varepsilon_2(\mathbf{x}) = (\mathrm{I}_n(\mathbf{x}) - \mathrm{I}_{n+1}(\mathbf{x})) - u_2 \cdot \frac{1}{2}(\mathrm{I}_{n+1}(\mathbf{x}) + \mathrm{I}_n(\mathbf{x})) - b_2 \end{cases} \tag{12}$$

with $u_1 = a_1 - 1$, $u_2 = a_2 - 1$. With such a model and assuming that $u \approx 0$, both estimations give effectively equivalent estimates. A simple experiment was conducted on 100 frames of *Rory O'More*, using no spatial variability in gain or offset using both formulations. With the classical model the correlation coefficient[20] was not 1 but rather $\sqrt{\hat{a}_1 \cdot \hat{a}_2} = 0.99$. The bias observed is quite important since the gain of each estimation is underestimated by about 0.01, which means that usually each successive frame will look darker of about $0.01 * 100 = 1$ gray level. By adding the symmetry aspect, the correlation coefficient was $\sqrt{\hat{a}_1 \cdot \hat{a}_2} = 0.99999$ (for 100 frames), which is visually not perceptible.

3.1.3. A new Estimator

Experience shows that using the linear model above is insufficient in extreme cases. Residual fluctuations appear in the restored output because the real underlying model is better described by some non-linear process. It becomes useful to incorporate non-linearity into the model in effect by cascading the idea of Naranjo with the previous model as follows.

$$(\mathrm{I}_{n+1}(\mathbf{x}) - f(\mathrm{I}_n(\mathbf{x}))) = \frac{1}{2}(\mathrm{I}_{n+1}(\mathbf{x}) + f(\mathrm{I}_n(\mathbf{x}))) \cdot u(\mathbf{x}) + b(\mathbf{x}) + \varepsilon(\mathbf{x}) \tag{13}$$

The non-linearity is expressed by the function f which operates over the entire image as a global process, while the spatially varying nature of the flicker is handled by the spatially varying gain a ($a = 1 - u$) and offset b. The offset function b was less relevant for the restoration of *Rory O'More*.

For computational simplicity, it is possible to treat f and u separately. The non-linearity is first removed as a pre-process using Naranjo's estimator.[13] Then the linear model is used to compensate for the spatially varying flicker i.e by solving:

$$\left(\mathrm{I}^p_{n+1}(\mathbf{x}) - \mathrm{I}^p_n(\mathbf{x})\right) = \frac{1}{2}\left(\mathrm{I}^p_n(\mathbf{x}) + \mathrm{I}^p_{n+1}(\mathbf{x})\right) \cdot u(\mathbf{x}) + b(\mathbf{x})$$

Where $\mathrm{I}_{n+1}(\mathbf{x})$ and $\mathrm{I}^p_n(\mathbf{x})$ are the frames obtained after the removal of the non-linearity.

However, none of the flicker models used here are valid in practice unless there is no motion between the frames. This is often not the case, and various techniques have been used to deal with local motion. The next section discusses the single unifying idea that links Flicker and Global Motion estimation.

4. ALGORITHMS FOR ROBUST M-ESTIMATION

In both the problems above, the effect of local motion is to cause outliers in the residual error which bias the parameter estimation process. To deal with this problem, the effect of these outliers must be reduced. This section presents two algorithms that can be used to perform the estimation of the global motion parameters and the flicker parameters (cf. sections 3.1 and 2.2). For simplicity, the parameters to estimate are noted Θ (instead of $\delta\Theta$ in section 2.2, or $\mathbf{a}$ in 3.1) and the error $\varepsilon(\mathbf{x})$ depends linearly on Θ.

The first algorithm is widely used and is called *The location step with modified weights*[6] in the robust statistic framework, or more commonly the *Iterative Reweighted Least Squares*,[21] or as ARTUR in the Half Quadratic (HQ) formulation.[22] The IRLS algorithm is reviewed in paragraph 4.1. The second algorithm, little known in computer vision literature, has first been called *The location step with modified residuals* in robust statistics[6] and as LEGEND in the HQ framework.[22] This algorithm is referred as Iterative Modified Residuals (IMR) and is explained in paragraph 4.2 for global motion parameter estimation.

The origin of both algorithms can be explained in different ways,[6,10,22] but for simplicity we choose here the Half-Quadratic framework. To minimize the energy $\mathcal{J}$ (cf. equation 10), HQ theory defines an augmented energy $\mathcal{J}^*$ with the same global minimum (for simplicity, the scale parameter σ_ρ is set to one):

$$\mathcal{J}(\boldsymbol{\varepsilon}) = \min_{\mathbf{b}} \left\{ \mathcal{J}^*(\boldsymbol{\varepsilon}, \mathbf{b}) = \sum_{\mathbf{x}} \rho^*(\varepsilon(\mathbf{x}), b(\mathbf{x})) \right\} \tag{14}$$

$\mathcal{J}^*$ is minimized iteratively in Θ and $\mathbf{b} = \{b(\mathbf{x})\}_{\mathbf{x}}$:

```
Do
    $\Theta^{(j)} = \arg\min_\Theta \{\mathcal{J}^*(\boldsymbol{\varepsilon}^{(j)}, \mathbf{b}^{(j)})\}$
    $\mathbf{b}^{(j+1)} = \arg\min_{\mathbf{b}} \{\mathcal{J}^*(\boldsymbol{\varepsilon}^{(j+1)}, \mathbf{b}^{(j)})\}$
Until convergence at final step $\boldsymbol{j}$ ($\widehat{\Theta} = \Theta^{(j)}$)          (15)
```

$\mathbf{b}$, the auxiliary variable, corresponds to weights on the residuals in the IRLS algorithm and hence is noted $\mathbf{w}$ in section 4.1. In the following sections, the matrix G corresponds to the Jacobian matrix of the error.

4.1. Iterative Reweighted Least Squares (IRLS)

The first proposed augmented energy can be written as:

$$\mathcal{J}^*\,(\boldsymbol{\varepsilon}, \mathbf{w}) = \sum_{\mathbf{x}} w(\mathbf{x})\ [\varepsilon(\mathbf{x})]^2 + \Psi(w(\mathbf{x})) \tag{16}$$

When the auxiliary variable $\mathbf{w} = \{w(\mathbf{x})\}_{\mathbf{x}}$ is fixed, the update is estimated by weighted Least Squares:

$$\Theta^{(j)} = [\mathrm{G}^T \mathrm{W}^{(j)} \mathrm{G}]^{-1}\ (\mathrm{G})^T \mathrm{W}^{(j)} \mathbf{z} \tag{17}$$

The matrix W is diagonal and collects all weights $w(\mathbf{x})$ defined by:

$$w(\mathbf{x}) = \frac{\rho'(\varepsilon(\mathbf{x}))}{2 \cdot \varepsilon(\mathbf{x})} \tag{18}$$

where ρ' is the differential $\partial\rho(\epsilon)/\partial\epsilon$.

4.2. Iterative Modified Residuals (IMR)

The second augmented energy can be written as:

$$\mathcal{J}^*\,(\boldsymbol{\varepsilon}, \mathbf{b}) = \sum_{\mathbf{x}} [\varepsilon(\mathbf{x}) - b(\mathbf{x})]^2 + \xi(b(\mathbf{x})) \tag{19}$$

When the auxiliary variable $\mathbf{b} = \{b(\mathbf{x})\}_{\mathbf{x}}$ is fixed, the update is computed by :

$$\Theta^{(j)} = [\mathrm{G}^T\ \mathrm{G}]^{-1}\ \mathrm{G}^T\ (\mathbf{z} - \mathbf{b}^{(j)}) \tag{20}$$

The minimization in $\mathbf{b}$ leads to:

$$b^{(j)}(\mathbf{x}) = \varepsilon(\mathbf{x})\ \left(1 - \frac{\rho'\,(\varepsilon(\mathbf{x}))}{2\ \varepsilon(\mathbf{x})}\right) \tag{21}$$

4.3. Remarks

It has been shown that the IRLS algorithm converges in less steps $\boldsymbol{j}$ to the estimate $\widehat{\Theta}$ than the IMR.[6,10] But in comparing equations (20) and (17), we see that the IMR algorithm involves less computation (product of matrixes [G^T G]) at each j step than the IRLS ([$\mathrm{G}^T\mathrm{W}^{(j)}\mathrm{G}$]). Most of all, in the specific case where [G is orthogonal, the equation 20 is simplified by[10]: $\Theta^{(j)} = \mathrm{G}^T\ (\mathbf{z} - \mathbf{b}^{(j)})$

5. PRACTICAL FLICKER AND GLOBAL MOTION ESTIMATION

Since flicker and global motion (shake or otherwise) can occur in the same sequence, it becomes useful to write both the flicker and global motion estimation problem in the same model expression as follows.

$$\begin{aligned} & I_n(\mathbf{x}) = a(\mathbf{x}) I_{n-1}(\mathbf{A}\mathbf{x} + \mathbf{d}_{n,n-1} + \mathbf{d}'_{n,n-1}(\mathbf{x})) + b(\mathbf{x}) \\ \text{Linearising w.r.t global motion} \Rightarrow \; & I_n(\mathbf{x}) = a(\mathbf{x}) \left[I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) + \nabla I_{n-1}(\mathrm{B}(\mathbf{x})\ \Theta) \cdot \cdot \delta\Theta \right] + \varepsilon(\mathbf{x}) \end{aligned} \tag{22}$$

The equation is made linear in the motion parameters in the usual way. As is shown in the expression above, this yields a non-linear equation in the gain and motion parameters i.e. a term $a(\mathbf{x})\delta\Theta$. To solve this system the flicker and motion unknowns are treated separably: in effect using a kind of multidimensional Newton-Rhapson approach.

A straightforward implementation of this idea is to iterate between the de-flicker and global motion estimation steps as previously described until a low enough error energy $\sum_{\mathbf{x}} \epsilon(\mathbf{x})$ is had. A more practical approach is to use the global motion estimation module as a pre-process for de-flicker, and then the final global motion estimates are generated after de-flickering. This latter idea would rely on the fact that image gradient is less affected by low frequency flicker and the motion estimation used here is gradient based. Thus one could consider that a reasonable (but not good enough for de-shaking) estimate of the global motion components can be had without de- flickering. This estimate would be good enough to increase the amount of useful data that the flicker estimation process uses from the image.

It transpires that this latter approach is the more practical, and made even more computationally feasible by using the idea of Integral Projections[23] to generate an *estimate* from the pre-processing Global Motion estimation module. Using a straightforward implementation of Integral Projections requires that a horizontal $I_n^h(i)$ and vertical $I_n^v(i)$ image section is created by summing columns and rows respectively as follows.

$$I_n^h(i) = \sum_j I_n(i,j) \qquad I_n^v(j) = \sum_i I_n(i,j) \tag{23}$$

Such projections are created for I_{n-1} as well and corresponding projections are matched over a range of possible motion components to yield estimates for the vertical and horizontal components of global motion. This assumes that the effect of local motion is reduced by the integration operation. Of course using Integral Projections in this manner restricts the pre- processing module to translational global motion only.

Three systems are therefore examined as follows.

F: Flicker estimation without any treatment of global motion

GM-F: An initial pre-process with Global Motion estimation, followed by Flicker estimation.

iGMF: Incorporating the integral projection method for global motion estimation with the robust flicker estimation process by iterating between updates for motion and flicker parameters.

To allow de-shaking, these systems are then followed by the 6 parameter robust global motion estimation process as outlined earlier.

To test the behaviour of these systems in a controlled environment, the first frame from the mobile and calendar sequence was duplicated to create a 100 frame sequence. Flicker (using order 6 cosine basis functions for gain and offset), random translational shake, artificial blotches, lines and noise were then added. Three frames from this sequence are shown in Figure 3. Results of parameter estimation on this sequence are shown in Figure 4. The MSE plot on the right measures the performance of the three systems with respect to the error in the estimated gain and offset function for each frame. All the systems perform well much of the time (note the low MSE), but the F system, without global motion compensation performs quite badly at several points. The GM-F system performs better but shows bad behaviour around frame 65 where the flicker was probably too much for the GM pre-process to cope. The iGMF system performs most reliably. The displacement plot on the left of the figure follows the same pattern. It is to be noted that both the integral projection and the 2D solution for Global motion perform identically in this case.

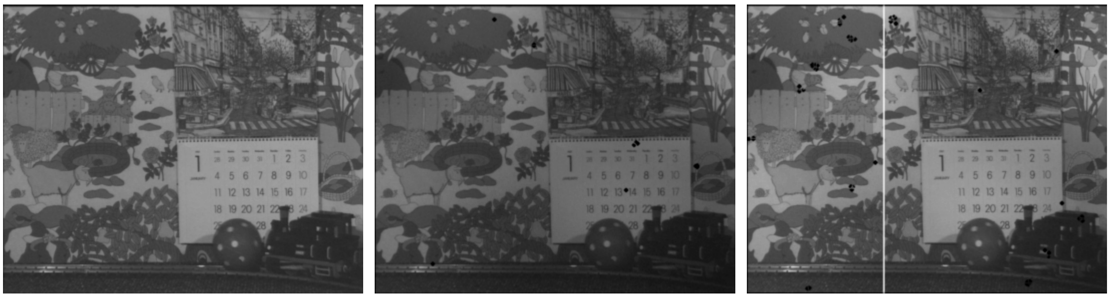

Figure 3. Frames 0,10,20 from the heavily artificially degraded Mobile sequence. The degradation due to flicker is easily seen in the stills. The shake is less visible.

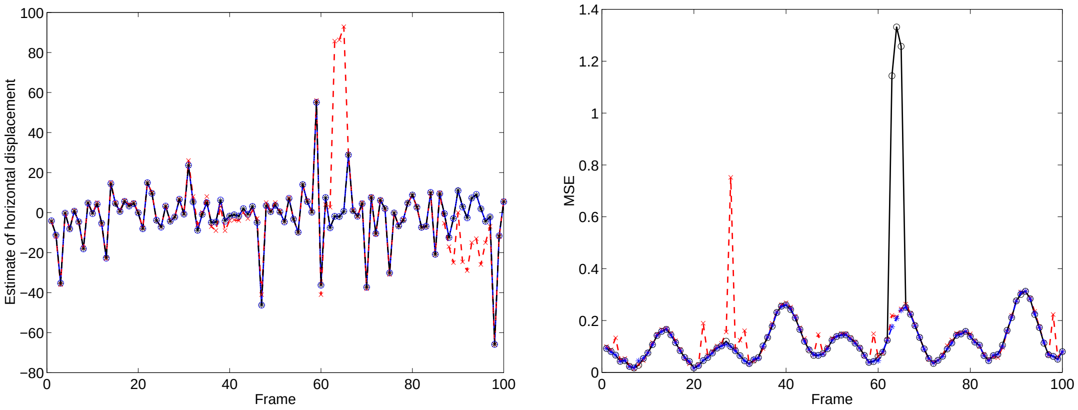

Figure 4. Results of parameter estimation using the degraded mobile sequence. Left: Estimated global horizontal displacement. Actual displacement ∘−, estimated with GM-F × − −, estimated with iGMF ∗−. Right: MSE between actual gain and offset components and that estimated with F × − −, GM-F ∘−, iGMF ∗−.

6. RESTORATION SCHEMES

Previous sections have shown ways to estimate the flicker and affine shake parameters between frames. These estimates consist of *both* deliberate effects (e.g. fade, pan, zoom) and unwanted effects. The difference between the types of effects is usually indicated by the fact that deliberate effects are gradual transitions while defects are high frequency. Given the observed parameter estimate corresponding to frames $n, n-1$ as $\Theta_{n,n-1}$ the signal can be written as $\Theta_{n,n-1} = \theta_{n,n-1} + \eta_{n,n-1}$ where $\theta_{n,n-1}$ is the true underlying signal component and $\eta_{n,n-1}$ represents noisy artefacts. The problem is to estimate $\eta_{n,n-1}$ and hence compensate the image for these global artefacts.

To estimate $\eta_{n,n-1}$ a high pass filter is used. To generate the actual measurements $\eta'_{n,n-1}$ that are used for compensation, two slightly different schemes are used for handling motion and flicker.

In the case of the motion parameters, the cumulative motion signal is smoothed with a non-linear high-pass filter H using $\hat{\theta}_n = H * \Theta^c_n$ where Θ^c_n is the cumulative motion signal and $\hat{\theta}_n$ is the smoothed output. If the motion estimate was purely translational then Θ^c_n would be given by $\Theta^c_n = \sum_{k=0}^{n} \mathbf{d}_{k-1,k}$. But as the parameters can be affine, the cumulation of the motion between frames involves the product of affine transformations $\mathbf{A}_{k-1,k}$ as well as the simple sum of translational components. It is necessary to use this signal since compensation requires that each frame be shifted relative to each of the previous frames. The high pass filter consists of a 3 tap median filter followed by a 30 tap FIR filter using the Hamming function as the coefficients. This non-linear filter is used because impulsive defects i.e. a large single frame shake are often observed. The idea is to reject all outliers immediately as artefacts, and smooth the remaining signal. The signal used for compensation is then

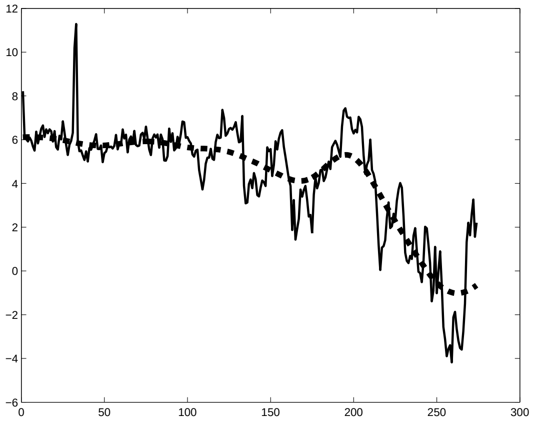

Figure 5. Smoothing the integrated motion estimates in *Rory O' More*

$\eta'_n = \Theta^c_n - \hat{\theta}_n$. An example of this process is shown in Figure 5. Those are results from using system GM-F on the real sequence, *Rory O'More*. Note how impulsive the raw estimates are.

In the case of the flicker parameters, the situation is slightly different. In real archived footage, it is still unclear that any of the proposed models are accurate. Cumulatively correcting for flicker through a recursive process can still yield artefacts if errors occur. Roosmalen[4] offered a simple mechanism for dealing with this by mixing the current deflickered output with the original degraded signal at each step. Here, an overlapped window method is used. Given a reference frame n, the flicker parameters between that frame and each degraded frame $n - K \dots n - 1$; $n + 1 \dots n + K$ is generated. Those parameters are smoothed using the filter described above and the value η'_n is used to compensate frame n only. This process is then repeated for frame $n + 1$, but the previously compensated frame is not re-used. This is a computationally intensive process, but it is able to cope with extreme fluctuations in flicker without propagation of errors.

7. REAL SEQUENCES

Assessing the performance of the systems on real degraded sequences is difficult because of the lack of any useful criteria for assessing the visibility of artefacts. However, it is feasible to expect that a good de-flicker process would reduce the fluctuations in the mean of image intensities from frame to frame. The Tunnel sequence as used in[4] and *Rory* were tested. The visual results can be seen at `www.mee.tcd.ie/sigmedia/˜research` in the form of .avi movies. A fourth system (AGM-F), using the 6 parameter affine global motion estimation process before de-flicker was also tested. The various combination systems perform about the same on the Tunnel sequence, presumably because there is little drastic Global Motion and the Flicker is not intense. For *Rory* the situation is different, and visibly better results are obtained when Global Motion and Flicker estimation are conducted together. The AGM-F and iGMF systems outperform F alone, but both AGM-F and iGMF yield comparable results.

Stills from the restored sequence do not show the improvement clearly so instead the average brightness and the variance of the pixel intensities for each frame are measured and plotted over the entire sequence. Figure 6 shows the mean brightness of each frame before (red) and after processing. All the processes clearly reduce the flicker, but it is harder to see from these plots which of the systems performs best. Since the visible problem with flicker is the variation in brightness between frames it is sensible to measure the sum of the absolute differences between mean brightness in consecutive frames and use that as a measure of temporal restoration smoothness. With this measure, the original Rory curve has a sum 'temporal gradient' of 328, while the restored data had sum gradients of 29 for F, and 25.5 for both iGMF and AGM-F. For Tunnel the situation is the same with the corresponding measurements being 182, 8.5 and 7.5 respectively. This implies that the combination systems are indeed yielding better temporal smoothness than using a de-flicker process alone without taking global motion into account.

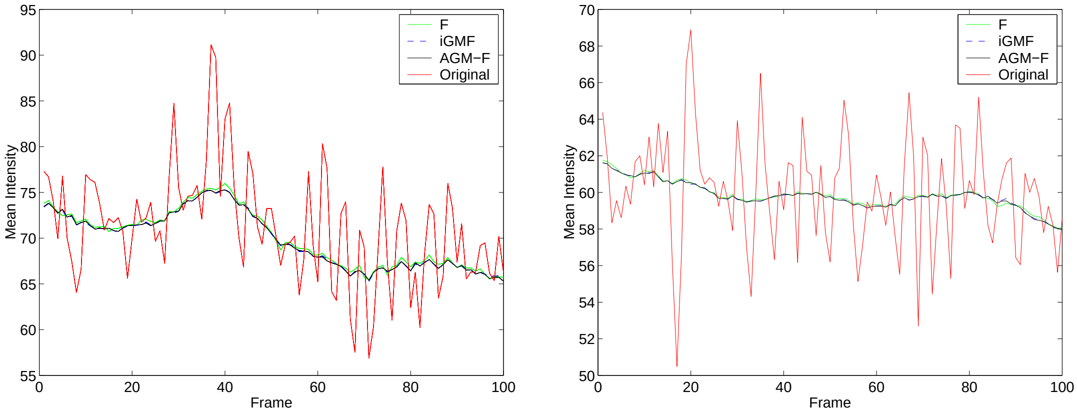

Figure 6. Mean brightness after various processes to remove flicker and shake from *Rory* (left) and *Tunnel* (right). Original, degraded brightness (red), F (green), iGMF (- -), AGM-F (black).

8. FINAL COMMENTS

The Flicker and Global Motion estimation problems have distinct similarities in the sense that both processes are biased by local motion or local artefacts in the image e.g. blotches, lines. By using a robust estimation framework, both problems can be treated in the same way by weighting out the effect of the local outliers. This paper has investigated the connections between the two problems, and illustrated that improved results can be had when the two defects are considered together. The use of Integral Projection was shown to be effective as a computationally low cost method for accommodating some measure of global motion while flicker estimation takes place. The visual results on *Rory* in particular, illustrate that when the two problems occur together it is difficult to achieve usable results without using a joint process.

ACKNOWLEDGMENTS

We would like to thank Ben Cloney, Cassandra O'Connel and those at the Irish Film Centre for collaborating with us to work on the film *Rory O'More*. This work was made possible through support from the EU IST Project BRAVA and the RTN project MOUMIR `www.moumir.org`.

REFERENCES

1. J. Tucker and A. de Sam Lazaro, "Image stabilization for a camera on a moving platform," in *Proc. of the IEEE Pacific Rim Conf. on Communications, Computers and Signal Processing*, **2**, pp. 734–737, May 1993.
2. K. Ratakonda, "Real-time digital video stabilization for multimedia applications," in *Proceedings International Symposium on Circuits and Systems*, **4**, pp. 69–72, IEEE, (Monterey, CA, USA), May 1998.
3. K. Uomori, A. Morimura, H. Ishii, T. Sakaguchi, and Y. Kitamura, "Automatic image stabilizing system by full-digital signal processing," *IEEE Transactions on Consumer Electronics* **36**, pp. 510–519, August 1990.
4. P. van Roosmalen, *Restoration of archived film and video.* PhD thesis, Delft University of Technology, October 1999.
5. T. Ohuchi, T. Seto, T. Komatsu, and T. Saito, "A robust method of image flicker correction for heavily-corrupted old film sequences," in *Proc. of the 2000 International Conference on Image Processing (ICIP-2000)*, September 2000.
6. P. Huber, *Robust Statistics*, John Wiley and Sons, 1981.
7. P. Bouthémy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization," *IEEE Transactions on Circuits and Systems for Video Technology* **9**, pp. 1030–1044, 1999.
8. A. Smolic and J.-R. Ohm, "Robust global motion estimation using a simplified m-estimator approach," in *IEEE International Conference on Image Processing*, (Vancouver, Canada), September 2000.

9. W. Qi and Y. Zhong, “New robust global motion estimation approach used in mpeg-4,” *Journal of Tsinghua University Science and Technology* , 2001.
10. R. Dahyot, *Appearance based road scene video analysis for the management of the road network.* PhD thesis, University of Strasbourg I, France, November 2001.
11. E. Decencière, *Restauration automatique de films anciens.* PhD thesis, Ecole Nationale Supérieure des Mines de Paris, December 1997.
12. X. Yang and N. Chong, “Enhanced approach to film flicker removal,” *Society of Photo-Instrumentation Engineers* , 2000.
13. V. Naranjo and A. Albiol, “Flicker reduction in old films,” in *Proc. of the 2000 International Conference on Image Processing (ICIP-2000)*, September 2000.
14. R. Gonzalez and P. Wintz, *Digital Image Processing*, 2nd ed., 1987.
15. F. Pitié, “Removing flicker from old movies,” Master’s thesis, University of Nice-Sophia Antipolis, France, September 2002.
16. S. Stigler, *The History of Statistics*, Cambridge: Belknap Press of Harvard Press, 1986.
17. F. Galton, “Regression towards mediocrity in hereditary stature,” *Journal of the Anthropological Institute* , pp. 246–263, 1886.
18. A. Lehman and J. Sall, “Why is it called regression?.” http://www.jmp.com/news/jmpercable/summer98/-regression.html.
19. F. D. la Torre and M. J. Black, “Robust principal component analysis for computer vision,” in *Proceedings of the International Conference on Computer Vision*, (Vancouver, Canada), juillet 2001.
20. E. Weisstein, “The world of mathematics.” Available at http://mathworld.wolfram.com/.
21. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C - The Art of Scientific Computing*, Cambridge University Press, 1995.
22. P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, “Two deterministic half-quadratic regularization algorithms for computed imaging,” in *Proceedings of the International Conference on Image Processing ICIP-94*, **2**, pp. 168–172, 1994.
23. C. Tu, T. D. Tran, J. L. Prince, and P. N. Topiwala, “Projection-based block motion estimation for video coding,” in *Proc. SPIE Applications of Digital Image Processing XXIII*, pp. 374–384, August 2000.