# An optimal blind temporal motion blur deconvolution filter

Yohann Tendero<sup>\*</sup>and Jean Michel Morel<sup>†</sup>

#### Abstract

Frames of a video sequence can be improved by a spatial deconvolution of any motion blur not exceeding two pixels per frame. Yet, this requires an accurate blur estimation and local deconvolution, which is problematic for multiple local motions. We introduce an optimal temporal blur deconvolution filter restoring blindly any nonuniform motion blur with an amplitude below one pixel per frame. The discrete filter has a very low complexity of about 20 operations per pixel. Experiments illustrate the method on simulated data, real movies and on sequences from the *Middlebury* database.

Note to the referees: further examples and C++ implementation are available at http://www.math.ucla.edu/~tendero/blind\_deconvolution/blind\_deconvolution.html.

### 1 Introduction

Most images and movies are degraded by motion blur due to camera or scene motions. Thus, moving scenes require a "small" exposure time, and the number of sensed photons remains limited. The difficulty of motion blur is illustrated by its simplest example, the one dimensional uniform motion blur. The result is a convolution of the image with a one dimensional window shaped kernel. The support of the kernel increases linearly with the exposure time and the velocity of the motion. As soon as the support of the motion kernel exceeds two pixels, the blur is no more invertible, and the restoration process becomes an ill posed problem. Furthermore, the result is highly dependent upon an accurate blur kernel estimate as remarked in [6], who also point out that even "the shift-invariant blur assumption made by most algorithms is often violated." On the other hand, a multi-frame approach requires an accurate estimation of the local motion "which is a very challenging task even with user interactions" [5]. In short, the estimation of motion blur coupled with its deconvolution remains a challenging and not completely solved problem. An exciting circumvention scheme of this classic photography dilemma was proposed in [4] where the authors suggest modifications in the acquisition process to get invertible blur kernels by adding to the camera a *flutter shutter*. If the shutter sequence is well chosen, invertibility is guaranteed for blurs with arbitrary size supports. Nevertheless, this requires designing a new sort of camera and an accurate estimation of the motion to deconvolve the *flutter shutter* blur. In [1] we introduced the notion of *numerical flutter shutter*, which is nothing but a discrete temporal filter. We also proved that for a maximal velocity  $v_{max}$  the optimal filter in terms of PSNR is given by the Fourier coefficients of a zoomed sinc function, only usable with a numerical flutter shutter. These considerations seemed a priori limited to the design of numerical *flutter shutter* cameras. Nevertheless, we develop in the present letter an unforeseen consequence for video post-processing, therefore applicable to a wide range of video data. We show the existence of a universal temporal video filter that deblurs video sequences regardless of their apparent local velocities, provided these velocities do not exceed a limit of one pixel/frame. Thus, this method only requires a rough estimate of the maximal velocity, applied as a safeguard before applying the filter. Indeed experiments on real data, including the benchmark *Middlebury* videos, show acceptable performance even when the apparent motion exceeds this limit.

Section 2 gives the temporal filter formalism taking the noise model into account. Section 3 provides practical implementation details and gives the *explicit* discrete (temporal) filter coefficients. Experiments on the *Middlebury* data base, camera phone movies and simulations are given in section 4.

### 2 The filter formalism

A temporal video filter consists in multiplying the k-th image, for  $k \in 0, ..., L - 1$ , by an  $\alpha_k \in \mathbb{R}$  discrete filter. Then all images are added to get the filtered frame. To formalize the deconvolution we can fix w.l.o.g. an arbitrary direction for

<sup>\*</sup>Y. Tendero is with the Department of Mathematics, University of California, 520 Portola Plaza, Los Angeles, CA-90095-1555, USA. E-mail: tendero@math.ucla.edu.

<sup>&</sup>lt;sup>†</sup>J.M. Morel is with the Centre de Mathématiques et de Leurs Applications (CMLA) at École Normale Supérieure de Cachan, 61 Av du Pdt Wilson, 94235 Cachan, FR.



Table 1: This table summarizes the main formulas for the video filter.

the motion blur. Indeed, applying a temporal filter does not require its knowledge. From the mathematical viewpoint, if the direction of the motion is known, a temporal filter  $\alpha$  boils down to the 1D convolution of the 1D restrictions of the observed landscape in that direction, with the 1D filter  $\frac{1}{v}\alpha(\frac{x}{v})$  where v is the velocity. Since the observation has noise, the expected value at position x of this landscape will be denoted by u(x). In all statements, this ideal landscape u is assumed to be band limited on  $[-\pi, \pi]$ , and with finite energy:  $u \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$ .

The full formalism of the temporal filter and the proof that it works is summarized in table 1. Its first line defines the discrete temporal filter  $\alpha(t)$ , whose goal is to invert the motion blur caused by the motion with velocity v. Because of this filter design is easier in a continuous setting, the second line, the second line considers general continuous temporal filters. Conversely, Theorem 2.1 permits to get back a piecewise constant (temporal) filter from any continuous filter  $\alpha \in L^2(\mathbb{R})$ .

The third line of the table gives the exact formula for the filtered noisy samples under the classic Poisson noise assumption for CCD's.  $X \sim Y$  means that the random variables X and Y have the same law and  $\mathcal{P}(\lambda)$  denotes a Poisson random variable with intensity  $\lambda$ . The temporally filtered digital image at pixel n is the linear combination of weighted acquired images, and each one is a (Poisson) stochastic variable. (Here and elsewhere, \* denotes the classic continuous convolution on  $\mathbb{R}$ .) As is explicit in the fourth line, the expected value of the filtered image is nothing but the result of the 1D convolution of the landscape u moving at velocity v with the temporal filter  $\alpha$ . The fifth line is obtained from the third and gives the expectation of the Fourier transform of the filtered image, when the filtered samples have been interpolated by the Shannon-Whittaker method as a band limited function in  $\mathbb{1}_{[-\pi,\pi]}$ . (Here and elsewhere  $\hat{u}$  denotes the classic continuous Fourier transform on  $\mathbb{R}$ .) The ideal temporal filter follows immediately, since it is the only band limited filter restoring perfectly the ideal landscape u (in expectation).

The six and seventh lines of the table give two general formulas for the MSE and SNR of a movie filtered by a temporal filter  $\alpha$  (they are proved in [1]). As usual, we call signal to noise ratio (SNR) the quotient of the average ideal landscape  $||u||_{L^1} = \int_{\mathbb{R}} u(x)dx$  by its root mean square error (RMSE). It follows easily from this formula (by Jensen's inequality) that for a known velocity v, the filter  $\alpha(t)$  maximizing the SNR is sinc(tv). Assuming now the existence of a maximal velocity  $v_{max}$ , the optimal temporal filter is defined by  $\hat{\alpha}(\xi v_{max}) = 1$  on  $[-\pi, \pi]$ , so that  $\alpha(t) = sinc(tv_{max})$  as indicated in the last line of the table. The good point of this filter is that it keeps a constant Fourier transform on the support of  $\hat{u}$  for any velocity  $|v| \leq |v_{max}|$ . Thus it yields a completely deblurred image for all velocities in the interval  $[-|v_{max}|, |v_{max}|]$ . We gave the above formalism in the realistic case where the image is a Poisson noise. Nevertheless, the optimal deconvolution filter is widely independent of the noise model. If for example the noise is uniform the expectation does not change and the MSE becomes  $\int \frac{||\alpha||_{L^2} \sigma^2}{|\hat{\alpha}(\xi v)|^2} \mathbb{1}_{[-\pi,\pi](\xi)} d\xi$ . As a consequence the optimal filter remains a sinc function. It is easily checked that this also holds for any combination of Poisson and uniform white noise.

All of this time-continuous theory must be discretized to take into account the fact that the movie has only discrete (generally not interpolable) samples. Fortunately, as soon as  $|v_{max}|$  (measured in pixel per frame) is below 1, Thm. 2.1 gives discrete filter coefficients, directly applicable for our sake<sup>1</sup>.

<sup>&</sup>lt;sup>1</sup>The time span  $\Delta t$  represents the time unit used to measure the velocities.



Figure 1: Top: the temporal filter function for the sinc filter (left) with a length L = 52. The Fourier transform (modulus) of the sinc filter (right), approximating the constant Fourier transform of the ideal filter for  $\Delta t = 1$  and  $v_{max} = 1$ . Bottom: the temporal filter function for the sinc filter (left) with a length L = 8 and its Fourier transform on the right. Even the filter with a "small" support has a nearly flat Fourier transform on  $[-\pi, \pi]$  approximating nicely the ideal continuous sinc filter.

**Theorem 2.1** [1] Let  $\beta \in L^2(\mathbb{R})$  be a band-limited temporal convolution kernel satisfying  $\hat{\beta}(v\xi) \neq 0$  for  $\xi \in [-\pi, \pi]$ . If  $|v|\Delta t \leq 1$ , there exists an invertible piecewise constant (discrete) filter  $\alpha(t) = \sum_{k \in \mathbb{Z}} \alpha_k \mathbb{1}_{[k\Delta t, (k+1)\Delta t]}(t)$  with  $(\alpha_k)_{k \in \mathbb{Z}} \in l^2(\mathbb{Z})$ , such that  $\hat{\alpha}(v\xi) = \hat{\beta}(v\xi)$  on  $[-\pi, \pi]$ . The coefficients  $\alpha_k$  of the discrete filter are explicitly given by  $\alpha_k = \frac{1}{2\pi} \int_{-\pi|v|\Delta t}^{\pi|v|\Delta t} \frac{\hat{\beta}(\frac{\xi}{\Delta t})e^{i\frac{\xi}{2}}}{sinc(\frac{\xi}{2\pi})}e^{ik\xi}d\xi$ .

#### 3 Blind motion blur deconvolution

As we have just seen for a fixed velocity  $v_{max}$  the best temporal filter is  $sinc(v_{max}t)$ . This filter permits to guarantee a sharp image for all velocities v in  $[-|v_{max}|, |v_{max}|]$ . Indeed, this ideal filter convolves the observed landscape by a zoomed sinc function whose cutoff is beyond the cutoff of observed landscape u (see figure 1). This means that the frequency content of the observed landscape remains unchanged, despite the motion. Thus, no velocity estimation and/or local deconvolution is needed: the observed image does not change if v = 0 or if  $|v| = |v_{max}|$ . In other words, the ideal filter is motion invariant, and so is the discrete filter, thanks to Thm. 2.1. Indeed, the image obtained by filtering is, in expectation, equal to the landscape convolved by  $sinc(tv_{max})$ . When the frequency cutoff of the sinc function is larger than the frequency cutoff of the landscape, the above theory implies that the resulting image is, in expectation, equal to the landscape. This remains valid for all velocities not exceeding  $|v_{max}|$ , where  $|v_{max}| = 1$  if we measure the velocity in pixel(s)/frame, the time unit between two consecutive frames being normalized to 1. Furthermore, the filter works regardless of the local motion directions, that need not be estimated. In short, the deconvolution is blind but mathematically well posed. The only required test is whether the overall motion magnitude does not exceed one pixel.

From a practical point of view, implementing the sinc requires the use of Thm. 2.1 to produce a discrete filter that allows movies post processing for instance. Thus, dropping the constant multiplicative factor and the time translation term the discrete filter coefficients are  $\alpha_k = \int_0^{\pi |v| \Delta t} \frac{\mathbb{1}_{[-|v_{max}|, |v_{max}|]}(\frac{\xi}{\Delta t}) \cos(k\xi)}{\sin((\frac{\xi}{2\pi}))} d\xi$ . We shall call this algorithm FSM (filtering by sinc for motion). Given L movie frames  $(I_k)_k \ k \in 0, ..., L-1$  the

We shall call this algorithm FSM (filtering by sinc for motion). Given L movie frames  $(I_k)_k \ k \in 0, ..., L-1$  the FSM image is  $\sum_{k+0}^{L-1} \alpha_k I_k$ . The observed sharp image is obtained from a linear combination of blurry images as the filter deconvolves itself, a fact illustrated in figure 4. Notice that discrete sinc filters are the only filter family that have the self-deconvolving property. The filter coefficients normalized so that  $\int \alpha(t) dt = 1$  are (0.0002, -0.0002, 0.0002, -0.0003, 0.0003, -0.0003, 0.0003, -0.0005, 0.0005, -0.0006, 0.0006, -0.0007, 0.0008, -0.0009, 0.0010, -0.0013, 0.0016, -0.0020, 0.0024, -0.0031, 0.0043, -0.0062, 0.0096, -0.0164, 0.0345, -0.1070, 1.1663, -0.1070, 0.0345, -0.0164, 0.0096, -0.0006, 0.0005, -0.0003, 0.0003, -0.0002) for a discrete filter of length  $L = 52, \ \Delta t = 1, \ v_{max} = 1 \ \text{and} \ (-0.0165, 0.0345, -0.1072, 1.1688, -0.1072, 0.0345, -0.0165, 0.0096) \$  when  $L = 8, \ \Delta t = 1, \ v_{max} = 1 \$ 

The quantitative results concerning the FSM method are summarized in tables 2 and 3.



Figure 2: This figure shows the result of the proposed sinc filter on two *Middlebury* sequences, *evergreen* (top-left) and *schefflera* (bottom-left) and two self made camera phone sequences (right). Each pair shows one frame of the sequence followed by the filtered image. Details are shown in figure 3.

Velocity	v = 0	v = 0.5	v = 1	v = 1.5	v = 2
House	-0.7	-0.5	+1.1	+2	+1.4
Boat	-0.7	-0.4	+1.5	+3.2	+1.1

Table 2: This table provides the PSNR evolution in dBs when the filter length is L = 8. The method permits to gain up to 3.2dB.

#### 4 Experiments

This section compares the proposed temporal filter and the snapshot acquisition (without the filter) both quantitatively and qualitatively. The simulations use a peer reviewed simulator [2], comparative RMSE are shown in figure 4 and tables 2 and 3 contain PSNR values. Real experiments using the *Middlebury* database are shown in figures 2. The proposed filter can be cumulated with some further enhancement, like a PSF based resharpening. Indeed, our filter compensates for motion blurs only and does not alter the OTF, as can be seen in figure 5.

# 5 Conclusion

We have proposed a new filter to process video sequences acquired by classic cameras. By applying the filter for a given maximal velocity, video sequences can be deblurred exactly and automatically by a fast fixed temporal filter requiring only the knowledge of the maximal observed velocity. This filter has been tested on synthetic results and permits an increase of up to 20% in RMSE (ie 3.2dB). The filter can be expected to work locally in all regions of the image where the motion is below the 1 pixel per frame. Nevertheless, a blind application seems to give good visual results in a wider range of velocities. This can be explained by the fast decay of the frequency content of images. (Beyond 2 pixels per frame the deconvolution becomes ill-posed anyway.) An (exact) knowledge of the velocity vector would also permit a direct spatial (not temporal) deconvolution, but the temporal filter works without requiring this accurate estimation.

These results mean that 1) it is possible to turn any camera into a camera that ensures a sharp image for a broader range of velocities than a standard camera 2) it is possible to perform an automatic blind movie deconvolution. In the second case, the proposed apparatus remains extremely simple as it creates the "always" sharp image as a linear combination of blurry images. Therefore, the computational cost is quasi null. Furthermore, assuming that the camera has provided a burst satisfying the maximal velocity condition, the proposed filter can be considered an useful addition,

Velocity	v = 0	v = 0.5	v = 1	v = 1.5	v = 2
House	-0.7	-0.4	+1.1	+2.1	+1.4
Boat	-0.7	-0.4	+1.6	+1.5	+0.7

Table 3: This table provides the PSNR in dBs when the filter length is L = 52. The method permits to gain up to 2dB.



Figure 3: This figure shows the result of the proposed sinc filter on two *Middlebury* sequences and two self made camera phone sequences. On the left side: one frame of the sequence, on the right side: crops of the filtered image. From top to bottom: the *evergreen* sequence, the *schefflera* sequence.



Figure 4: This figure provides the RMSE for the snapshot (green curves) and for the filtered image (red curves) in function of the velocity v for a normalized exposure time  $\Delta t = 1$ . On the left side: for the L = 8 filter, on the right side: for the L = 52 filter. At the top: for the House image, at the bottom: for the Boat image. For small velocities the filtered image is slightly noisier than the snapshot, the noise remains white nonetheless. This is due to the finitely supported piecewise constant approximation of the ideal sinc filter. For larger velocities the filter shows a significant gain in terms of RMSE. As predicted by the theory, the RMSE of the filtered image remains almost constant for all velocities  $|v| \leq 1$  pixel/frame.



Figure 5: This figure shows the result of a Gaussian sharpening both before (1st image) and after the filtering (2nd image). The 4th and 5th images are crops of the first resp. the second. The filtered image remains sharper and no artifact is introduced by the filter. This experiment was done using [3] and can be checked on line.

not impeding other image improvement operations such as denoising by accumulation or spatial PSF deconvolution.

# References

- Y. Tendero, J-M Morel, and B. Rougé. The Flutter Shutter paradox. Accepted to SIAM Journal on Imaging Sciences, pages 1–33, 2012.
- [2] Y. Tendero. The Flutter Shutter Camera Simulator. Image Processing On Line, 2012, doi:10.5201/ipol.2012.t-fscs.
- [3] P. Getreuer Total Variation Deconvolution using Split Bregman. Image Processing On Line, 2012, doi:10.5201/ipol.2012.g-tvdc.
- [4] A. Agrawal and R. Raskar. Resolving objects at higher resolution from a single motion-blurred image. In *Proceedings* of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8, 2007.
- [5] J.F. Cai, H. Ji, C. Liu, and Z. Shen. Blind motion deblurring from a single image using sparse approximation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 104–111, 2009.
- [6] A. Levin, Y. Weiss, F. Durand, and W.T. Freeman. Understanding and evaluating blind deconvolution algorithms; In Computer Vision and Pattern Recognition (CVPR), pages 1964–1971, 2009.