

USE OF MOTION INFORMATION IN SUPER-RESOLUTION MOSAICING

P. Krämer^a, O. Hadar^b, J. Benois-Pineau^a, J.-P. Domenger^a

^a LaBRI, CNRS UMR 5800/University of Bordeaux 1, 351, cours de la Libération, F-33405 Talence

^b Ben-Gurion University of the Negev, Dept. of Comm. Sys. Eng., P.O. Box 653, IL-84105 Beer-Sheva

ABSTRACT

In this paper, we present a super-resolution (SR) method based on iterative backprojections. Motion information is used for the synthesis of the restoration filter in the SR method. Both, the blur estimation and the choice of the degradation model, are based on estimated global motion. The method is applied to highly under-sampled images such as DC images of MPEG compressed video and medical magnetic resonance (MR) images. Results of the comparison of our blur model with some common blur models are encouraging.

Index Terms— Super-resolution, blur estimation

1. INTRODUCTION

The problem of recovering high quality stills from low resolution (LR) video has been addressed in various research [1, 2, 3, 4] due to its promising industrial applications. The goal of these SR methods is to combine a set of LR images of the same scene in order to obtain a single image of higher resolution while also compensating for blurring. Various blur estimation and restoration techniques for this task have been proposed [5, 6].

In our previous work, we developed a method for mosaicing with increased resolution from MPEG compressed streams using only DC low resolution frames and motion vectors. In [7], we already thought to use motion information not only in the mosaicing process, but in the restoration as well. We proposed a method for the estimation of motion blur in the direction of global motion in the image plane. In this paper we go further and introduce the knowledge of motion information into the design of the restoration filter. We apply our method to two classes of video: broad-casted main stream video and medical magnetic resonance imaging (MRI) sequences. Despite the very different nature of the imaging system, in both cases the image degradation by motion blur appears. Motion blur in broad-casted video arises due to the camera motion and in medical MRI sequences due to organ motion during breathing and blood supply.

The paper is organised as follows: In section 2, we review the reference SR method from Irani and Peleg [4] that we apply to mosaics for restoration and increasing the image resolution. In section 3, we present the blur model in motion

direction, its parameter estimation and the restoration method. In section 4, we compare our blur model to common 2D blur models and present some results. Section 5 concludes our work and shows perspectives for future research.

2. REFERENCE SR METHOD

We chose the iterative backprojection (IBP) algorithm from [4] to increase the resolution of mosaics because of its simplicity and rapidity. The algorithm starts with an initial guess of the SR mosaic m :

$$m^{(0)} = \beta_K \sum_{k=1}^K T_k(g_k \uparrow s) \quad (1)$$

where $\uparrow s$ is the upsampling operator, T_k is the geometric transformation from the LR image g_k to the SR mosaic m , K is the number of LR images g_k in the sequence, and $\beta_K(p) = \frac{1}{|p|}$ with $|p|$ as the number of available pixels at position p .

Then, at each iteration step the imaging process is simulated by:

$$g_k^{(n)} = (T_k^{-1}(m^{(n)}) * h_k) \downarrow s \quad (2)$$

where T_k^{-1} is the geometric transformation from the SR mosaic m to the LR image g_k , h_k is the point spread function (PSF) of the k 'th LR image, $\downarrow s$ is the downsampling operator, and $*$ is the convolution operator. The result is a set of LR images corresponding to the sequence of the original LR images.

Then, the difference between the simulated frames and the input frames is used to update the SR mosaic:

$$m^{(n+1)} = m^{(n)} + \beta_K \sum_{k=1}^K T_k \left(((g_k - g_k^{(n)}) \uparrow s) * q_k \right) \quad (3)$$

where $\downarrow s$ is the downsampling operator inverse to $\uparrow s$, q_k is the restoration filter determined by h_k , and β_K has the same meaning than in (1). The update process (3) is repeated until a maximum number of iterations is achieved.

The main challenge of current SR methods is 1) the motion estimation and 2) the blur estimation and its restoration. Concerning the problem of motion estimation i.e. estimating the geometric transformations T_k , we refer the reader to our

previous article [7]. This paper addresses the problem of blur estimation and restoration. Thus in the next section, we will focus on the operators h_k and q_k . We describe the blur model in motion direction, its parameter estimation, and the restoration of this blur using the Wiener filter.

3. MODELLING OF BLUR IN MOTION DIRECTION

In the previous article [7], we used an isotropic Gaussian blur similar to [4]. Our aim indeed is to restore directional motion blur that comes from the camera motion in video or organ motion in medical image sequences. Thus, an isotropic blur model is not convenient and we now model the blur in the motion direction. In many cases, there are no extreme changes in motion velocity during the exposure time since it is usually very short. In these cases, extreme changes do not occur in the PSF shape [5]. Thus on the contrary to [5] and [6], we assume a constant velocity of the motion.

Let k_d be the velocity of the motion according to the direction $\vec{d}$. Then, the module transfer function (MTF) is [8]:

$$MTF_{\text{sinc}}(f_d) = \left| \frac{\sin(\pi k_d f_d T)}{\pi k_d f_d T} \right| = |\text{sinc}(\pi b f_d)| \quad (4)$$

with T as the exposure time and $b = k_d T$ as the spatial extend of the blur.

Now let us consider $\vec{d} = \vec{d}_x + \vec{d}_y$. In this case the 1D motion blur in $\vec{d}$ -direction can be represented as a composition of two motion blurs in x and y -direction:

$$\begin{aligned} MTF_{\text{sinc}}(u, v) &= \text{sinc}(\pi b_x u) \cdot \text{sinc}(\pi b_y v) \\ &= MTF_{\text{sinc}}(u) \cdot MTF_{\text{sinc}}(v) \end{aligned} \quad (5)$$

where $b_x = k_x T$ and $b_y = k_y T$ are respectively the spatial extends of the blur in x and y -direction, and $k_x = k_d \cos(\alpha)$ and $k_y = k_d \sin(\alpha)$ represent respectively the x and y -components of the velocity k_d .

The *sinc* MTF in motion direction can thus be represented by the combination of a horizontal *sinc* function and a vertical *sinc* function. The next subsection, then presents the computation of the blur extend for the MTFs in x and y -direction.

3.1. Blur Parameter Estimation

The blur can vary in each image, since it depends on the motion direction, the velocity, and the exposure time. Thus, we estimate the parameters b_x , b_y for the blur extend in x and y -direction respectively from the image itself. Therefore, we use the edge response in the LR images g_k .

We first locate the edges by computing the first derivative ∇g of the image using the Sobel operator and a non maxima suppression of the gradient magnitude. Then a threshold th_1 is applied to the gradient magnitude in order to extract significant edges. We obtain the following set of pixels p :

$$E_{\nabla} = \{p \mid \|\nabla g(p)\| > th_1\} \quad (6)$$

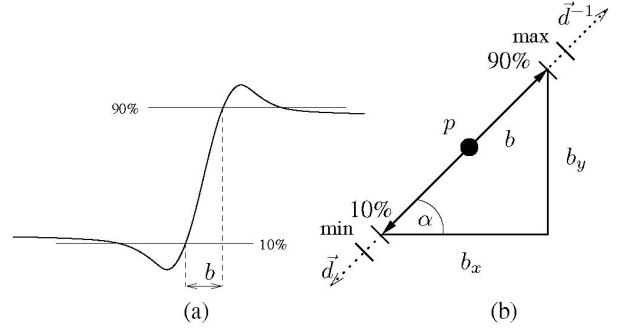


Fig. 1. (a) Edge response in motion direction of a blurred edge, (b) Computation of the edge response in x and y -direction.

The edges in a blurred image have the form as shown in Fig. 1(a). They are limited by a local maximum and a local minimum determining the rise of the edge. Then, the width of the edge response i.e. the blur extend b is derived by the 10% to 90%-distance.

Hadar et al. [8] considered horizontal blur and estimated the width of the edge response in this direction. On the contrary, we consider directional blur and so estimate the width of the edge response in motion direction. Then, the blur parameters b_x and b_y can simply be obtained by trigonometry which is shown in Fig. 1(b).

The motion direction $\vec{d}$ is known from the motion compensation vector for the geometric transformations T_k^{-1} . Thus, our algorithm does not need additional computation in order to determine the direction of the blur. Yitzhaky and Kopeika [5] determine the motion direction by measuring the direction where the total energy of the image derivative is lowest. Rav-Acha and Peleg [6], propose to calculate the angle by computing one single iteration of their method for each angle, and the angle which gives the strongest deblurring effect is identified as the angle of the motion blur.

We measure the blur extend only on edges that are orthogonal to the motion direction. Therefore, we compute the angle between the gradient $\nabla g(p)$ and the motion vector $\vec{d}(p)$ and retain only the pixels for which the angle is smaller than the threshold th_2 .¹ The resulting pixel set is:

$$E_{\nabla, \perp} = \{p \in E_{\nabla} \mid \arccos \left(\frac{\vec{d}(p) \cdot \nabla g(p)}{\|\vec{d}(p)\| \|\nabla g(p)\|} \right) \frac{180}{\pi} < th_2\} \quad (7)$$

The position p in the LR image g_k indicates the center of the edge, situated between the local minimum and the local maximum. A discrete straight line segment on a pixel grid according to the direction $\vec{d}$ is considered, centered on p . The local maximum and minimum are searched along the segment

¹We inverted $\vec{d}(p)$, if $\angle(\vec{d}(p), \nabla g(p)) > 90^\circ$ for an easier computation. Then, $\vec{d}(p)$ points in the direction of the local maximum.

in the LR image g_k . Then, the 10%- and 90%-limits can be determined using the same method.

The blur extend in motion direction $b(p)$, $p \in E_{\nabla, \perp}$ is computed as the euclidean distance of the 10% and 90%-limits. Now, the blur parameters $b_x(p)$ and $b_y(p)$ can be determined using trigonometric triangle rules (see Fig. 1(b)):

$$\begin{aligned} \cos(\alpha(p)) &= \frac{\vec{d}(p) \cdot (1,0)^T}{\|\vec{d}(p)\|} & \sin(\alpha(p)) &= \frac{\vec{d}(p) \cdot (0,1)^T}{\|\vec{d}(p)\|} \\ b_x(p) &= |b(p) \cdot \cos(\alpha(p))| & b_y(p) &= |b(p) \cdot \sin(\alpha(p))| \end{aligned} \quad (8)$$

Finally, we compute the blur parameters for the whole image b_x and b_y as the average of all estimated values of $b_x(p)$ and $b_y(p)$ respectively.

3.2. SNR Estimation

Highly under-sampled images suffer from quantisation noise which influences the restoration step. The SNR generally varies for different image sequences since it depends on the imaging system. Consequently, we estimate the SNR from LR image. The SNR for the image g_k can be computed as the ratio [9]:

$$SNR = \frac{h^2}{\sigma_N^2} \quad (9)$$

where h is the edge height and σ_N^2 the variance of the noise. These parameters are computed on the pixel sets defined in the last subsection as follows:

$$\begin{aligned} h &= \max\{|\nabla g(p)| \mid p \in E_{\nabla, \perp}\} \\ \sigma_N^2 &= 1/|E_{\nabla}^c| \sum_{p \in E_{\nabla}^c} (|\nabla g(p)| - \mu)^2 \\ \mu &= 1/|E_{\nabla}^c| \sum_{p \in E_{\nabla}^c} |\nabla g(p)| \end{aligned} \quad (10)$$

with E_{∇}^c as the complement of E_{∇} .

3.3. Restoration

We use the Wiener filter for the image restoration in the SR algorithm because it takes into account image noise. Since we separated the MTF for the x and y -direction (see equation (5)), the restoration is also performed separately:

$$W_k(f) = \frac{|MTF_k(f, b_f)|^*}{|MTF_k(f, b_f)|^2 + 1/SNR_k}, \quad f = u, v \quad (11)$$

We restore the lines using $MTF_{\text{sinc}}(u)$ and then the columns using $MTF_{\text{sinc}}(v)$. The image restoration in the update process (3) is then accomplished in the spectral domain using the Fourier transform F :

$$\begin{aligned} & \left((g_k - g_k^{(n)}) \uparrow s \right) * q_k = \\ & F^{-1} \left(F \left((g_k - g_k^{(n)}) \uparrow s \right) \cdot Q_k \right) \\ & \text{with } Q_k = W_k(u) \cdot W_k(v) \end{aligned} \quad (12)$$

The blur operator h_k of equation (2) is computed similarly in the spectral domain.

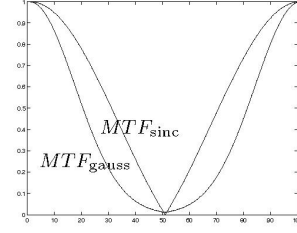


Fig. 2. The normalised functions MTF_{gauss} for $\sigma = 1$ and MTF_{sinc} for $b = 1$.

4. COMPARISON WITH CLASSICAL 2D BLUR MODELS

In the previous article [7], we modelled the PSF as a 2D isotropic Gaussian. In this section, we compare the performance of the *sinc* MTF (5) in the SR method with these of the isotropic Gaussian MTF and an anisotropic Gaussian MTF. The Fourier transform of a Gaussian is a Gaussian too. Then, the corresponding MTF is:

$$\begin{aligned} MTF_{\text{gauss}}(u, v) &= \frac{1}{2\pi\sigma_u\sigma_v} \exp\left(-\frac{1}{2}\left(\frac{u^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right)\right) \\ &= \frac{1}{\sqrt{2\pi}\sigma_u} \exp\left(-\frac{u^2}{2\sigma_u^2}\right) \frac{1}{\sqrt{2\pi}\sigma_v} \exp\left(-\frac{v^2}{2\sigma_v^2}\right) \end{aligned} \quad (13)$$

σ_u and σ_v are respectively inverse related to σ_x and σ_y . In order to obtain σ_x , σ_y from the blur extend, we apply the 6 σ -rule: then $\sigma_x = b_x/6$ and $\sigma_y = b_y/6$. For the isotropic Gaussian, we use $\sigma_x = \sigma_y = b/6$.

In order to compare the Gaussian with the *sinc* MTF, we have to normalise both functions with respect to the size of the spectrum. Therefore, we suppose that all values bigger than $MTF_{\text{gauss}}(\pm 3\sigma)$ equal zero. For MTF_{sinc} , we compute the *sinc* function only until the first zero-crossing. We normalize both functions so that the size of the MTF equals the size of the image spectrum for $\sigma = 1$ or $b = 1$. Fig. 2 shows the normalised functions MTF_{gauss} and MTF_{sinc} .

We use the mean square error (MSE) of the SR mosaic with respect to the LR images as a measure of performance:

$$MSE = \frac{1}{K} \sum_{k=1}^K \left(\frac{1}{|p|} \sum_p (g_k(p) - g_k^{(n)}(p))^2 \right) \quad (14)$$

In this experiment we used two LR sequences of 5 DC images extracted from the SFRS documentaries “Le chancre coloré du platane” and “La joueuse de Tympanon” and a MRI sequence of 5 images. Fig. 3 shows the MSE results for the three MTFs with respect to the number of iterations for the “Chancre” and the “Tympanon” sequences. In the “Chancre” sequence shown in Fig. 4(a), the motion is not parallel to the x or y -axis. In this case the anisotropic Gaussian MTF yields

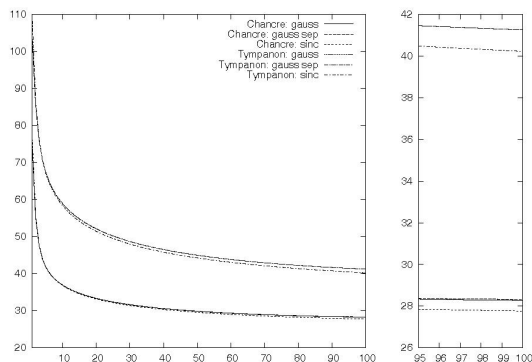


Fig. 3. Left: MSE for the sequences “Chancre” and “Tympanon”, right: zoom on the iterations 95–100

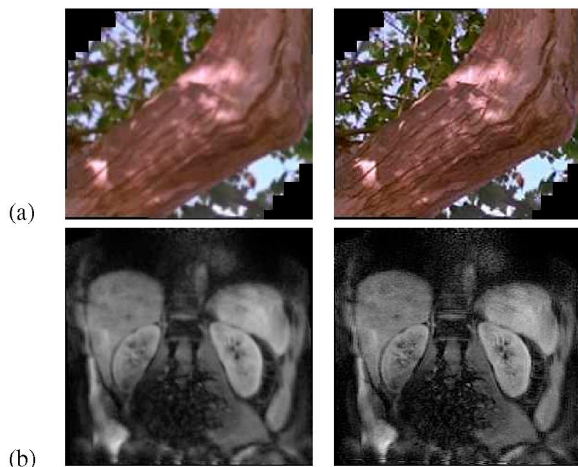


Fig. 4. (a) Left: the initial SR mosaic for the “Chancre” sequence, right: the SR mosaic after 10 iterations © SFRS, (b) left: the initial SR mosaic for the MRI sequence, right: the SR mosaic after 10 iterations.³

a slightly better result than the isotropic Gaussian MTF. For the “Tympanon” sequence (we showed it in [7]) and the MRI sequence of Fig. 4(b) the motion is parallel to the y -axis. The SNR (9) of both sequences is higher as well, though the Wiener filter behaves like a “pass all filter” in the x -direction and restores more high frequencies. In the case of imperfect motion or strong aliasing effects this can yield to an error. The *sinc* MTF gives clearly better results in terms of the MSE. This can be explained by our normalisation. If $b \ll 1$, then the Gaussian MTF will be closer to the constant 1 than the *sinc* MTF. Thus the *sinc* MTF will more attenuate high frequencies than the Gaussian.

³We show here the images after only 10 iterations since we then already have satisfying results without high frequency amplification due to aliasing or border ringing artifacts due to the Fourier transform.

5. CONCLUSION AND PERSPECTIVES

In this paper, we made a step further to the truly use of motion information in the problem of restoration of high resolution stills from low resolution video. We presented a 1D blur model in motion direction and an appropriate estimation method for its parameters. Then, we compared the performance of this blur model with respect to common 2D blur models and it gives clearly better results. The focus of future work will be the treatment of local blurs.

6. ACKNOWLEDGMENTS

We thank Chrit Moonen from the IMF/University of Bordeaux 2, France and Pascal Desbarats from LaBRI/University of Bordeaux 1, France for providing the biomedical image sequence.

7. REFERENCES

- [1] A.J. Patti, M.I. Sezan, and A.M. Tekalp, “Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time,” *IP*, pp. 1064–1078, 1997.
- [2] T. Komatsu and T. Saito, “Super-resolution sharpening-demosaicking method for removing image blurs caused by an optical low-pass filter,” in *ICIP 2005*, 2005, vol. 1, pp. 845–848.
- [3] M. Ben-Ezra, A. Zomet, and S. K. Nayar, “Video super-resolution using controlled subpixel detector shifts,” *PAMI*, vol. 27, no. 6, pp. 977–987, 2005.
- [4] M. Irani and S. Peleg, “Motion analysis for image enhancement: Resolution, occlusion, and transparency,” *JVCIP*, vol. 4, pp. 324–335, 1993.
- [5] Y. Yitzhaky and N. S. Kopeika, “Identification of blur parameters from motion blurred images,” *Graph. Models Image Process.*, vol. 59, no. 5, pp. 310–320, 1997.
- [6] A. Rav-Acha and S. Peleg, “Restoration of multiple images with motion blur in different directions,” 2000, pp. 22–28.
- [7] P. Krämer, O. Hadar, J. Benois-Pineau, and J.-P. Domenger, “Super-resolution mosaicing from mpeg compressed video,” in *ICIP 2005*, 2005, vol. 1, pp. 893–896.
- [8] O. Hadar, I. Dror, and N.S. Kopeika, “Image resolution limits resulting from mechanical vibrations, part iv: Real-time numerical calculation of optical transfer, functions and experimental verification,” *Opt. Eng.*, vol. 33, no. 2, pp. 566–578, 1994.
- [9] W. K. Pratt, *Digital Image Processing*, Wiley, 1978.