Local object-based super-resolution mosaicing from low-resolution video

Petra Krämer, Jenny Benois-Pineau, Jean-Philippe Domenger

► To cite this version:

Petra Krämer, Jenny Benois-Pineau, Jean-Philippe Domenger. Local object-based super-resolution mosaicing from low-resolution video. *Signal Processing*, 2011, 91 (8), pp.1771-1780. hal-00714650

HAL Id: hal-00714650

<https://hal.science/hal-00714650>

Submitted on 7 Jul 2012

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

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

Local object-based super-resolution mosaicing from low-resolution video

Petra Krämer^{a,*}, Jenny Benois-Pineau^b, Jean-Philippe Domenger^b

^a*L3i, University of La Rochelle, La Rochelle, France*

^b*LaBRI, University of Bordeaux 1, Talence, France*

Abstract

Many efficient super-resolution methods have been presented in the past for improving resolution of images degraded by global blurs. Unfortunately, in video, more complex situations can be observed where local blurs appear in each frame which are mainly caused by object motions. To address this problem, we propose in this article a local super-resolution method which allows the restoration of such local blurs. Moreover, the motion of objects in video sequences may be very complex and particularly in very low-resolution sequences it is difficult to estimate their motion exactly enough to superimpose them for super-resolution. To this end, we present a generic method: An interpolation method is proposed to improve the resolution of moving objects and we derive from this a super-resolution method for the scene background.

Keywords: Super-resolution, mosaicing, local blur estimation/restoration

2000 MSC: 68U10, 94A08

*Corresponding author. Fax: + 33 5 46 45 82 42

Email addresses: `pkraemer@univ-lr.fr` (Petra Krämer), `benois-p@labri.fr` (Jenny Benois-Pineau), `domenger@labri.fr` (Jean-Philippe Domenger)

1. Introduction

Super-resolution (SR) is the process of combining a sequence of low-resolution (LR) images in order to produce a higher resolution image or sequence, the so-called SR image or sequence. SR is still a very active research area due to ever expanding application domains. For instance, SR methods have been adopted for still construction from video [38], video format conversion [30], demosaicking [26], improving the video resolution in a camera [4], reduction of compression artifacts [16], and deinterlacing [37]. A good overview of existing SR approaches is given in [36, 14] for raw video and in [41] for compressed video.

Even though powerful SR methods have been presented, only few of them address the problem of estimating and restoring spatially variant blurs. These blurs are mainly caused by object motions or atmospheric turbulence. One of the first methods for object-based SR was presented in [23] where an object with dominant motion is tracked over several frames assuming 2D parametric motion to improve its resolution. However, the method does not account for local blurs. The general approach of [3] is similar to [23], but incorporates motion blurring. The authors of [12] propose a SR method accounting for motion blur by extending the method [38] to the case of multiple moving objects in the scene. In [13], the method [12] is reused in the context of mesh-based object representation. For both, results are only shown for rigid objects. Other methods use optical flow to describe object motion [30] or atmospheric distortions [51]. A gradient-based shift estimator is used in [48] for object motion. Moreover, in [48, 31] the problem of super-resolving very small objects is addressed. The authors of [48] state that for small moving

26 objects the amount of information inside the moving object is too small for
 27 appropriate registration. In order to overcome this problem, they propose a
 28 polygon-based SR approach [49] in order to super-resolve very small objects.
 29 Nevertheless, they still assume rigid object motion. [31] proposes to super-
 30 resolve small objects by non-uniform interpolation followed by restoration.
 31 Objects are assumed to undergo affine motion, but the problem of local blur
 32 is neglected, the shape of the object is approximated by a rectangle and needs
 33 to be initialized by the user. A different approach to handle object motions
 34 are block-based methods such as [4, 2]. Recently, a new class of local SR
 35 methods have been derived from noise filters such as moving least squares
 36 [6], kernel regression [45], non-local means [39], or restoration filters like the
 37 Wiener filter [19]. [32] proposes a SR method based on locally adaptive
 38 bilateral total variation in order to keep edges sharp and flat regions smooth.
 39 As this kind of approaches are at the beginning of investigation, few of them
 40 account for local motion [39, 45] and none for local blur.

41 The motion of objects in real video sequences can be very complex. For
 42 example consider the sequence shown in Fig. A.4(a) with a walking person
 43 undergoing local motions. Particularly in the case of very LR images the
 44 objects are typically represented by only few pixels. For that reason, it is
 45 very difficult to estimate the motion of moving objects and to superimpose
 46 them accurately enough for SR, as also stated in [48]. Thus, the computation
 47 of a parametric motion model [23, 31] or optical flow [30] is not appropriate
 48 in that case. Furthermore, in contrast to [48, 49] where rigid object motion
 49 is assumed, our aim is to propose a generic method which can handle small
 50 objects with complex motions. Therefore, we present in this paper an inter-

51 polation method to improve the the resolution of moving objects and derive
52 from this a SR method to super-resolve the scene background.

53 Recently, blind SR has become in the focus of interest [44, 35, 20, 21].
54 That means that the blur function and its parameters used in the SR process
55 are estimated from the image data itself. Typically, the proposed SR meth-
56 ods rely on blind deconvolution methods. Although several methods for blind
57 deconvolution have been proposed [28, 29, 7], at present only few methods
58 have been presented for the case of spatially variant blurs as well. In [46],
59 the image scene is segmented into regions where the blur is approximated as
60 locally spatial invariant. The authors of [34] assume piece-wise constant blur
61 functions which are stitched together by interpolation to obtain a continuous
62 blur function. The method presented in [47] allows the blur to vary at each
63 pixel location whereas the blur size is derived from the estimated displace-
64 ment vector. In [1], a framework is presented which jointly handles motion
65 estimation, moving object detection, and motion blur deconvolution. A mo-
66 tion blur model is incorporated which is consistent at motion singularities
67 (caused by the moving object occluding and uncovering the background),
68 but the method can only handle one single moving object. The method of
69 [8] estimates motion blur kernels, segmentation, and motion simultaneously
70 allowing multiple moving objects. A different approach is presented in [43]
71 where blur variations due to different depths are considered. The method
72 [25] computes the motion blur from a transparency map. In [22, 18], a block-
73 based method for local blur restoration is presented. The authors of [10]
74 present a method to track motion-blurred targets in video. To this end, local
75 motion blurs are identified by a learning-based scheme.

76 In our recent work [27], we presented an efficient SR method for the blind
77 restoration of global blurs due to the camera motion. The main drawback of
78 this method is that restoration is performed in frequency domain which does
79 not allow for processing local blurs. Here, we address these shortcomings.
80 The SR method we present in this article performs restoration in spatial
81 domain and thus enables the restoration of local blurs induced by the motion
82 of objects. Based on a segmentation of moving objects, our SR method is
83 able to estimate and locally restore blur. To this end, we extend our blur
84 estimation method [27] for global motion blur due to the camera motion to
85 the case of local blurs. This enhanced method can handle both, motion blur
86 of the camera occurring in the static scene background and motion blurs of
87 various moving objects.

88 We apply our SR method to the construction of super-resolved mosaics
89 (SR mosaicing) from compressed video. The construction of mosaics from
90 compressed video has become important as still new application domains
91 emerge. First of all, a mosaic image can be efficiently used for visually sum-
92 marizing compressed video content which is our case. Furthermore, mosaic-
93 ing, specifically for compressed content, is a powerful tool for the reconstruc-
94 tion of a global scene view from compressed LR videos captured with mobile
95 phones, e.g. document images [17]. In both applications SR mosaicing is a
96 must for the efficient use of LR video data.

97 This article is organized as follows: In Sect. 2, we present the estimation of
98 local blurs, namely on moving objects and the scene background, and the SR
99 method which allows their restoration. Some results are shown and discussed
100 in Sect. 3. Conclusions and perspectives are given in Sect. 4. Moreover,

101 Appendix A presents the derivation of our SR method.

102 **2. Local Object-based Super-resolution**

103 As stated above, it is very difficult to superimpose moving objects in
 104 very LR images. To this end, we propose to super-resolve separately mov-
 105 ing objects and the scene background. This requires preliminary motion
 106 estimation in order to determine the geometric transformations of the back-
 107 ground between the images of the sequence and the reference image, as well
 108 as segmentation and tracking of moving objects. Here, we suppose that this
 109 is already accomplished. We used [27] for registration and [33] for moving
 110 object segmentation and tracking.

111 *2.1. Local Super-resolution in Spatial Domain*

Basic iterative restoration techniques can be easily adapted to the case of local blurs. For this reason, we chose the basic iterative image restoration method [9, 24] as a basis for our SR algorithm. In order to derive from [9, 24] (see also Appendix A) a SR method to restore local blurs, we formulate a degradation model for an arbitrarily shaped region of interest (ROI) considering local motion and downsampling. Indeed, the image formation model [9, 24] was formulated for an ideal unknown image and its observed blurred version of the same resolution. In our case, when zooming on the mosaic, the ideal image is of high resolution, namely SR, and its observed blurred version is of LR. Hence, the ROI in the LR image G can be modelled as:

$$G = S^{-1}.B * [T.F] + V \quad (1)$$

112 where F is the ROI in the SR image, B the point spread function (PSF)
 113 defining the blur of the ROI, T the geometric transformation describing the
 114 local motion of the ROI, S^{-1} the downsampling operator by the factor z ,
 115 and V the noise.

Based on this degradation model, we derive an iterative algorithm to restore and increase the resolution of a ROI:

$$F^i = F^{i-1} + T^{-1}.S \left(G - S^{-1} [B * (T.F^{i-1})] \right) \quad (2)$$

116 where F^i is the ROI in the SR image at the i th iteration and S the upsampling
 117 operator. The proof can be found in Appendix A.1.

Considering a sequence of K LR images where each frame represents the same ROI $G(k)$, $1 \leq k \leq K$, the following iterative SR algorithm can be derived from (2) for a SR mosaic (see Appendix A.2):

$$M^i = M^{i-1} + \mu(K) \sum_{k=1}^K T^{-1}(k).S \left(G(k) - S^{-1} [B(k) * (T(k).M^{i-1})] \right) \quad (3)$$

118 where M^i is the ROI in the SR mosaic at the i th iteration and $\mu(\mathbf{p}, K) = \frac{1}{|\mathbf{p}|}$
 119 with $|\mathbf{p}|$ as the number of available pixels at position $\mathbf{p}$. We can notice that
 120 this equation is quite similar to the SR method we presented in [27], but the
 121 restoration operator is absent. The advantage here is, if we can identify the
 122 blurring operator, the restoration is straight forward and does not require
 123 the synthesis of a restoration filter as e.g. in the case of Wiener filtering.

A problem in (2) and (3) is that B is defined for the SR image which is unknown. Our objective is to directly use the LR PSF instead of making an

assumption on it for the SR image as we did in [27]. It can be shown that for small blurs (see Appendix A.3):

$$(S^{-1}.B) * (S^{-1}.F) \approx \frac{1}{z} S^{-1}(B * F) \quad (4)$$

124 This relation means that the convolution of LR image with the LR PSF can
 125 be approximated for small blurs by $1/z$ times the downsampled blurred SR
 126 image. Thus, $1/z$ corresponds to a normalization factor. Since we use a
 127 normalized convolution mask it is omitted in our computations.

128 Taking into account (4), we can reformulate the Eqs. (2) and (3). Fur-
 129 thermore, the additional problem of strong aliasing occurs in very LR im-
 130 ages. This means that images can not be exactly superimposed on edges
 131 and textures in the SR method and a strong motion compensation error
 132 $G(k) - [(S^{-1}.B(k)) * (S^{-1}.T(k).M^{i-1})]$ results in such regions. This error
 133 amplifies along the iterations and causes spurious artefacts in the SR mo-
 134 saic. To mitigate these artefacts, we incorporate the regularization operator
 135 A [27] in the SR method (3) which penalizes the contribution of edges and
 136 textures. The Eqs. (2) and (3) become:

$$F^i = F^{i-1} + T^{-1}.S \left(G - [(S^{-1}.B) * (S^{-1}.T.F^{i-1})] \right) \quad (5)$$

$$137 \quad M^i = M^{i-1} + \mu(K) \sum_{k=1}^K T^{-1}(k).S.A(k) \left(G(k) - [(S^{-1}.B(k)) * (S^{-1}.T(k).M^{i-1})] \right) \quad (6)$$

138 Note, that T in (5) is not used for motion compensation, but for the estima-
 139 tion of the local blur B as we will describe below. Moreover, there is no need
 140 for A in (5) because its computation is based on motion compensation.

141 In fact, the Eqs. (5) and (6) resemble the steepest descent algorithm [11].

142 If we solve Eq. (1) in the least-square sense and apply the steepest descent
 143 algorithm, there will be a convolution with the transposed PSF B^T before S
 144 in the Eqs. (5) and (6). Thus, our method is less costly than the steepest
 145 descent algorithm as one convolution at high resolution is omitted. Similar
 146 to [24] and the steepest descent algorithm, we introduced a gain factor β_1
 147 and β_2 in the Eqs. (5) and (6) respectively.

148 2.2. Local Blur Estimation

149 Thanks to moving object segmentation we already know the regions of
 150 local blurs contrary to [10]. We assume like in [46] that the blur in the
 151 segmented regions can be locally approximated by a spatially invariant blur
 152 model. Thus, the shape of the blur operator is unknown beforehand and its
 153 parameters can vary in each image. Blur can be due to different reasons such
 154 as motions due to the camera or objects, out-of-focus, the imaging system, or
 155 compression. Thus, the overall blur corresponds to the sum of the different
 156 blurs. Therefore, we consider three different PSFs.

Isotropic Gaussian. It is modelled by a 2D Gaussian function:

$$B_{\text{Gauss2D}}(x, y, \sigma_\alpha) = \frac{1}{2\pi\sigma_\alpha^2} \exp^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma_\alpha^2}\right)} \quad (7)$$

157 where σ_α is the standard deviation in motion direction.

Anisotropic Gaussian in motion direction. The anisotropic Gaussian in
 motion direction can be expressed as [15]:

$$B_{\text{Gauss}}(x, y, \sigma_\alpha, \sigma_{\alpha\perp}, \alpha) = \frac{1}{2\pi\sigma_\alpha\sigma_{\alpha\perp}} \exp^{-\frac{1}{2}\left(\frac{(x\cos\alpha+y\sin\alpha)^2}{\sigma_\alpha^2} + \frac{(-x\sin\alpha+y\cos\alpha)^2}{\sigma_{\alpha\perp}^2}\right)} \quad (8)$$

158 where $\sigma_{\alpha\perp}$ is the standard deviation in orthogonal motion direction. B_{Gauss}
 159 is anisotropic when $\sigma_{\alpha} \neq \sigma_{\alpha\perp}$.

Linear motion blur. Let v be the velocity of the motion according to the direction $\mathbf{d}$. Then, the PSF in motion direction for linear motion is:

$$B_{\text{box}}(\nu) = \int_{-E/2}^{E/2} \delta(\nu - vt) dt \quad (9)$$

160 where E is the exposure time and ν is the directional variable.

Let us consider a displacement vector $\mathbf{d}$ in motion direction developed as $\mathbf{d} = \vec{d}_x + \vec{d}_y$ where $\vec{d}_x$ and $\vec{d}_y$ are the components in x-y orthogonal basis. We suppose that this motion is translational. Let us denote by α the angle with the x-axis (see Fig. 1(b)), then the 1D motion blur in $\mathbf{d}$ direction can be represented as a composition of two motion blurs in horizontal and vertical direction:

$$B_{\text{box}}(x, y) = \int_{-E/2}^{E/2} \delta(x - v \cos(\alpha)t) dt \cdot \int_{-E/2}^{E/2} \delta(y - v \sin(\alpha)t) dt \quad (10)$$

Here $v \cos(\alpha)$ and $v \sin(\alpha)$ respectively represent the horizontal and vertical components, v_x and v_y , of the velocity v (orthogonal projection). Hence:

$$B_{\text{box}}(x, y, b_x, b_y) = \int_{-E/2}^{E/2} \delta(x - b_x) dt \cdot \int_{-E/2}^{E/2} \delta(y - b_y) dt \quad (11)$$

161 with b as the size of the blur in motion direction with b_x, b_y as its horizontal
 162 and vertical components, and $b_x = v_x t$ and $b_y = v_y t$.

163 The problem now consists in estimating the parameters $\sigma_{\alpha}, \sigma_{\alpha\perp}, b_x, b_y$
 164 of Eqs. (7), (8), and (11). Assuming that $b = 3\sigma$ (3σ -property), all blur

models are defined by the blur sizes b , b_x , b_y , and $b_\perp$ which is the blur size in orthogonal motion direction. To estimate these blur sizes, we extend our global blur estimation method [27] which is based on the computation of the edge response in motion direction only on edges orthogonal to the motion direction. Similarly to [47], the motion vector dictates the direction of the blur. For local blur estimation, we distinguish two cases: (i) the blur estimation for the background and (ii) the blur estimation for a moving object. Thus, we propose an adaptation of our blur estimation method [27] for each case.

In order to estimate the blur parameters of the image background, we only take into account the pixels belonging to the scene background obtained from moving object segmentation. We first locate edges by computing the first derivative ∇G of the LR image G using the Sobel operator. Non-maxima suppression and a threshold are applied to the gradient magnitude in order to extract significant edges $\mathcal{E}_\nabla$. To determine the edges which are orthogonal to the motion direction, $\mathcal{E}_{\nabla,\perp}$, we compute the angle between the gradient and the motion vector $\mathbf{d}$ and retain only the pixels for which the angle is smaller than a threshold. The motion vector $\mathbf{d}$ is known from the geometric transformation resulting from motion estimation T and thus no additional computation is needed to determine the direction of the blur.

For each point of $\mathcal{E}_{\nabla,\perp}$ a local estimation of the edge response is accomplished. The edges in a blurred image have the form shown in Figure 1(a). They are limited by a local minimum and a local maximum determining the rise of the edge. Then, the width of the edge response e is defined by the 10% to 90%-distance [42]. Let $\mathbf{m}$ denote the position in the LR image G which

190 indicates the center of the edge, situated between the local minimum and the
 191 local maximum. A discrete straight line segment on a pixel grid according
 192 to the direction $\mathbf{d}$ is considered, centered on $\mathbf{m}$. The local maximum and
 193 minimum are searched along the line segment in the LR image G . Then, the
 194 10% and 90%-limits can be determined using the same method.

195 The local edge response in motion direction $e(\mathbf{m})$, $\mathbf{m} \in \mathcal{E}_{\nabla, \perp}$, is computed
 196 as the Euclidean distance between the 10% and 90%-limits. Now, its hori-
 197 zontal and vertical components, $e_x(\mathbf{m})$ and $e_y(\mathbf{m})$, can be determined using
 198 trigonometric triangle rules. Finally, b , b_x , b_y are, respectively, computed as
 199 the average of local values $e(\mathbf{m})$, $e_x(\mathbf{m})$, $e_y(\mathbf{m})$ for all $\mathbf{m} \in \mathcal{E}_{\nabla, \perp}$. We compute
 200 $b_{\perp}$ in the same way than to b , but we use the orthogonal vector of $\mathbf{d}$ in the
 201 computations in order to compute the edge response of in orthogonal motion
 202 direction on edges in motion direction. The result is a locally constant PSF
 203 of the background.

204 In the case of very LR, objects are typically represented by only few
 205 pixels. Therefore, an edge detection inside the object is not reasonable and
 206 we directly use the boundary of the segmented region as significant edges
 207 $\mathcal{E}_{\nabla}$. To determine the motion vector $\mathbf{d}$ at each pixel of the object boundary,
 208 a rough guess of the motion is sufficient which can be obtained from object
 209 tracking. Furthermore, the computations of the edge response and the blur
 210 sizes are the same as described for the background.

211 2.3. Convolution of the ROI

212 Instead of convolving the ROI with a 1D convolution kernel in motion di-
 213 rection which is complex and costly, e.g. in [47] bilinear interpolation is used
 214 to compute the convolution in motion direction, we propose here to compute

215 a 2D convolution kernel using the horizontal and vertical components of the
 216 blur size, b_x and b_y , and compute then a traditional convolution.

We denote $\mathbf{K}$ as the 2D convolution kernel of the size $K_x \times K_y$. In the case of the Gaussian blur model, we determine the kernel size with respect to the 3σ -property:

$$K_x = \lceil 3\sigma_\alpha \rceil + 1 = \lceil b_x \rceil + 1 \quad K_y = \lceil 3\sigma_{\alpha\perp} \rceil + 1 = \lceil b_y \rceil + 1 \quad (12)$$

217 where $\sigma_\alpha = \sigma_{\alpha\perp} = \sigma_\alpha$ for the isotropic Gaussian, and $\lceil \cdot \rceil$ is the ceil operator.
 218 K_x does not necessarily equal K_y in case of the anisotropic model. Never-
 219 theless, we fix 3×3 as the minimum size of the kernel in both cases. We
 220 prefer a slightly larger convolution kernel, therefore also the ceil operator, to
 221 avoid a hard cut-off of the convolution kernel. Then, having determined the
 222 size of the kernel $\mathbf{K}$, its values $\mathbf{K}(x, y)$ are computed by Eq. (7) for isotropic
 223 Gaussian and by Eq. (8) for the anisotropic Gaussian. $\mathbf{K}$ is normalized af-
 224 terwards.

In case of linear motion blur the kernel is computed as:

$$K_x = \text{round}(b_x) + 1 \quad K_y = \text{round}(b_y) + 1 \quad (13)$$

$$\mathbf{K}(x, y) = \begin{cases} 0.5 & \text{if } \text{round}(b_x) \text{ even and } x = 0 \text{ or } x = K_x - 1 \\ 0.5 & \text{if } \text{round}(b_y) \text{ even and } y = 0 \text{ or } y = K_y - 1 \\ 1 & \text{otherwise} \end{cases} \quad (14)$$

225 Then, $\mathbf{K}$ is normalized. We choose 3×1 or 1×3 as the minimum kernel size
 226 depending on whether $b_x > b_y$ or not. It happens that $b_x < 1$ and $b_y < 1$ in
 227 our computations since b_x and b_y are, respectively, computed as the average
 228 of several estimated values. Thus, fixing the minimal the kernel size to 3×1

229 or 1×3 for the linear motion blur means that we make the critical assumption
 230 that there is a minimal blur of one pixel in horizontal or vertical direction
 231 in the ROI. Using this blur model we can not treat blur sizes smaller than
 232 one pixel due to the discretisation as the linear motion blur is a constant
 233 function. This is different for the Gaussian blur model. As the Gaussian is a
 234 continuous declining function, we can handle blur sizes smaller than 1. This
 235 only results in a value near 1 in the center of the convolution kernel and in
 236 small values near 0 at the borders.

237 The segmented objects can be of arbitrary form with irregular boundaries.
 238 For the convolution on object boundaries, we chose MPEG-4-like padding
 239 for boundary macroblocks [40] to extrapolate the object in the region of
 240 undetermined pixels underlying the convolution mask.

241 **3. Results**

242 In this section, we show some results obtained using the presented SR
 243 method for mosaic construction of compressed video. We evaluate the con-
 244 vergence of our algorithms using quadratic error measures and analyze the re-
 245 sults in terms of visual quality, spectrum widening and computational times.

246 *3.1. Mosaic Construction*

247 We used DC images of MPEG-2 compressed streams as LR image se-
 248 quence. They are a good example of very LR images as they are 8-fold
 249 smaller than the original frames. They are strongly aliased, and contain blur
 250 due to camera or object motions, and block averaging during compression.
 251 For registration, moving object segmentation and tracking, we refer the read-
 252 ers to [27] and [33] respectively. The SR background mosaic is constructed

253 using (6), some of the foreground objects are restored using (5) and inserted
 254 into the SR background mosaic. We used a zoom factor $z = 2$ for all exper-
 255 iments. Then, the resulting mosaic gives an appropriate scene overview for
 256 video summarization.

257 In some cases, additional postprocessing is applied. First, holes can ap-
 258 pear in the mosaic due to the exclusion of objects during the blending. As
 259 they are usually of small size, we interpolate the lacking pixels from the
 260 neighborhood. Furthermore, to remove visual artefacts in the vicinity of
 261 background borders, we apply a simple median filtering which removes high
 262 frequency noise while preserving edges. Finally, the insertion of an object
 263 causes typically seams at the borders of the object. For realistic object in-
 264 sertion, we apply a 3×3 mean filter on the object borders.

265 3.2. Evaluation of the Proposed Method

The quadratic error measure for the SR background mosaic (6) is:

$$\bar{\epsilon}^i = \frac{1}{K} \sum_{k=1}^K \left(\frac{1}{N(k)} \sum_{\mathbf{m}} \|G(\mathbf{m}, k) - [(S^{-1}.B(k)) * (S^{-1}.T(k).M^{i-1})](\mathbf{m})\|^2 \right) \quad (15)$$

266 where $N(k)$ is the number of pixels $\mathbf{m}$ in $G(k)$. We consider color images
 267 $G(k)$ and compute here the squared norm of vector difference. We use the
 268 error measure (15) additionally as stopping criterion for (6). The maximum
 269 number of iterations is achieved when $\bar{\epsilon}$ converges.

Similarly, we define an error criterion for moving objects (see Eq. (5)):

$$\epsilon^i = \frac{1}{N} \sum_{\mathbf{m}} \|G(\mathbf{m}) - [(S^{-1}.B) * (S^{-1}.T.F^{i-1})](\mathbf{m})\|^2 \quad (16)$$

270 where N is the number of pixels $\mathbf{m}$ in G .

271 Since the type of blur is unknown in our LR image sequence and to test
 272 the performance of the blur models in our SR method, we compare the PSFs
 273 presented in Sect. 2.2. Fig. 2(a) shows the graphs of error measures for
 274 the three different PSFs for the restoration of the background mosaic of the
 275 sequence “Comportements”. One image of the LR sequence is illustrated in
 276 (a). The anisotropic Gaussian PSF B_{Gauss} gives the best results according
 277 to error measure $\bar{\epsilon}$. The isotropic Gaussian PSF B_{Gauss2D} is close to the
 278 B_{Gauss} as we deal only with small blur in this sequence. The result obtained
 279 with the linear motion blur PSF B_{box} is not satisfying. This is due to the
 280 discretization of the convolution kernel. We made the assumption that at
 281 least one pixel blur appears in horizontal or vertical direction. As there is
 282 only small blur in this sequence ($b < 1$), the blur model is not appropriate
 283 and ringing artefacts appear in the mosaic. For all blur models convergence is
 284 achieved after few iterations. The error measure (16) of the object restoration
 285 is shown in Fig. 2(b). We observe similar characteristics for the three PSFs.

286 Fig. A.4 shows some results of mosaic construction for the sequence “Com-
 287 portements”. The corresponding computational times¹ are shown in Tab. 1.
 288 The initial mosaic (initial background mosaic combined with the bilinearly
 289 interpolated object) before applying the SR method is shown in (b). Its result
 290 after applying our method is shown in (c) which is much less blurred (see also
 291 the difference image (c)). If we compare the spectrum (b) and (c), we can
 292 see that the spectrum has widened after SR. There is only an increase of 8s
 293 for the iterations of the SR algorithm. This means that the concatenation of

¹They were obtained on a 3.2 GHz Intel Core 2 Duo processor using a non optimized C++ code and the VXL image library [50].

294 motion models in the geometric transform and motion compensation vector
 295 field computation is quite expensive. The motion estimation method we used
 296 computes a motion model for each I- and P-Frame. However, DC images are
 297 extracted at I-Frame basis, so that for each pair of DC images 5 motion mod-
 298 els have to be concatenated. The use of another motion estimation method
 299 might improve those computational times.

Method	Iterations	Time
(b) Initial mosaic	0	13.34s
(c) Proposed method	19 (+9)	16.36s
(d) Accelerate gradient descent	18 (+19)	18.74s
(e) Upsampled mosaic + deblurring	0	3.84s
(f) Downsampled frames	0	13.98s

Tab. 1: Computational times for the SR mosaics of Fig. A.4.

300 To improve the computational times of the SR iterations we implemented
 301 the gradient descent algorithm for the L2-norm of Eq. (2) for the objects and
 302 for the background mosaic similar to [11]. In [5], it was shown that the con-
 303 jugated gradient method is much slower than the steepest descent method,
 304 but the accelerate gradient descent was shown to be two times faster than
 305 the steepest descent method. Therefore, we compare our method with an ac-
 306 celerate gradient scheme similar to that one of [5]. The gradient is computed
 307 at the initial point and we keep the same gradient while the error functional
 308 decreases. The result is shown in (d). Visually there is no difference with
 309 respect to our method, Moreover, the computational time is the higher be-
 310 cause on the one hand its needs more iterations until convergence and on
 311 the other hand it is more complex due to an additional convolution with the
 312 transposed PSF and convolution at high resolution.

313 Additionally, we show in (e) the result for the interpolated mosaic con-
314 structed at LR which was deblurred afterwards by a pseudo-inverse filter. We
315 used the isotropic Gaussian PSF and the parameters were estimated from the
316 interpolated mosaic itself [27]. The result is not satisfying as strong ringing
317 artefacts appear on the image border as well on strong edges (also visible
318 in the difference image). This is due to the restoration which causes an
319 amplification of lower and middle frequencies, but a cutoff of high frequen-
320 cies. The computational time is low as the computation of motion vector
321 fields is achieved at LR, and the computation of regularization operator and
322 iterations of the SR algorithm are omitted.

323 Furthermore, we show the mosaic constructed from downsampled full-
324 resolution frames in (e). Downsampling was achieved by a Gaussian pyramid.
325 If we compare the resulting mosaic with that one obtained by the proposed
326 method, more less middle and high frequencies are present in the spectrum
327 which lets the mosaic appear less sharper. In the difference image we notice
328 an elevated error of the blue channel in the background which might be due
329 to video compression.

330 Fig. A.3 shows the result for the sequence "Hiragasy" which contains two
331 moving objects. One image of the LR sequence is illustrated in (a). We tested
332 on this sequences as well the three blur models with similar results than for
333 the previous example. The SR mosaic after 9 iterations for the background
334 and 3 iterations for the objects of our method using the anisotropic Gaussian
335 PSF is shown in (b). Both objects contain more high frequency details than
336 in the LR image.

337 We used $\beta_1 = 2.5$ and $\beta_2 = 1.5$ for the computation of the results shown

in Figs. A.3 and A.4. This choice seems to be a good compromise between precision and convergence for these images. If β_1, β_2 are chosen to small, convergence is very slow. If they are chosen to high, a high error of (15) and (16) results, and the SR mosaic is blurred.

4. Conclusion and Perspectives

We presented in this article a generic super-resolution method which performs blind restoration of local blurs in spatial domain. Thus, this method allows taking into account moving objects. Based on a segmentation of moving objects, background and moving objects are processed separately. As it is often impossible in very low-resolution image sequence to superimpose moving objects accurately enough for super-resolution, we proposed an interpolation method to improve the resolution of moving objects and a super-resolution method for the scene background. Consequently, we proposed a blur estimation method to estimate local blurs in motion direction.

We tested several blur functions in our restoration scheme. Best results were obtained for the anisotropic Gaussian blur whereas the results for the isotropic Gaussian blur were quite close. In case of small blurs, the linear motion blur does not seem an appropriate blur model due to our discretization of the convolution kernel.

Visual results are quite satisfying. High frequencies could be restored in our experiments for the background as well as moving objects. Computational times are fast, but can still be improved by optimizing the source code. Our super-resolution method assumes that the type of the blur function is known. Thus, future work will focus on the computation of the blur function

362 from the low-resolution image itself.

363 **Appendix A. Proofs**

In this section, we present the mathematical derivation of the spatial domain restoration methods presented in Sect. 2. Both methods are derived from the deconvolution method [9, 24]. They model the blurred image as:

$$\tilde{F}(\gamma) = \int_{-\infty}^{+\infty} F(x)B(\gamma - x) dx \quad (\text{A.1})$$

364 where $\tilde{F}$ is the blurred image, F the unknown optimal image and B the PSF.

Then, by approximating successively the desired optimal image as:

$$F(\gamma) = \tilde{F}(\gamma) + \Delta(\gamma) \quad (\text{A.2})$$

The following iterative scheme results:

$$F^i(\gamma) = F^{i-1}(\gamma) + \left[\tilde{F}(\gamma) - \int_{-\infty}^{+\infty} F^{i-1}(x)B(\gamma - x) dx \right] \quad (\text{A.3})$$

365 In the following, we first derive the interpolation method (2) from the
 366 deconvolution method [9, 24]. Then, we derive the SR method (3) from the
 367 latter. Finally, we derive a relationship between the convolution of a LR
 368 image with a LR PSF and the downsampled blurred SR image.

369 *Appendix A.1. Image Interpolation*

370 Here, we demonstrate the derivation of the spatial domain restoration
 371 method (2) allowing to increase resolution from the successive approxima-
 372 tions (A.2). Therefore, we consider an extended image formation model

373 relating a SR image with a LR image by incorporating motion and down-
 374 sampling: SR image $\rightarrow$ motion $\rightarrow$ blur $\rightarrow$ downsampling $\rightarrow$ LR image
 375 Hence, we can rewrite Eqs. (A.1) and (A.2) as:

$$G(y) = S^{-1} \int_{-\infty}^{+\infty} (T.F(x)) B(\gamma - x) dx \quad (\text{A.4})$$

$$F(\gamma) = T^{-1} (S.G(y) + \Delta(\gamma)) \quad (\text{A.5})$$

376 where y is the position in the observed LR image G , γ is the corresponding
 377 position in the SR image F , T is the geometrical transformation from F
 378 to the G , T^{-1} is the inverse geometric transformation, S is the upsampling
 379 operator, S^{-1} is the downsampling operator, and B the PSF.

Inserting (A.5) in (A.4):

$$G(y) = S^{-1} \int_{-\infty}^{+\infty} (S.G(y)) B(\gamma - x) dx + S^{-1} \int_{-\infty}^{+\infty} \Delta(x) B(\gamma - x) dx \quad (\text{A.6})$$

380 Denoting:

$$G^1(y) = S^{-1} \int_{-\infty}^{+\infty} (S.G(y)) B(\gamma - x) dx \quad \Delta(\gamma) = \int_{-\infty}^{+\infty} \Delta(x) B(\gamma - x) dx \quad (\text{A.7})$$

Then, (A.6) becomes:

$$G(y) = G^1(y) + S^{-1}.\Delta(\gamma) \quad \Leftrightarrow \quad \Delta(\gamma) = S (G(y) - G^1(y)) \quad (\text{A.8})$$

Inserting (A.8) in (A.5):

$$F(\gamma) = T^{-1}.S.G(y) + T^{-1}.S (G(y) - G^1(y)) \quad (\text{A.9})$$

Assuming that:

$$F^0(\gamma) = T^{-1}.S.G(y) \mid T \quad \Leftrightarrow \quad T.F^0(\gamma) = S.G(y) \quad (\text{A.10})$$

Inserting in (A.7):

$$G^1(y) = S^{-1} \int_{-\infty}^{+\infty} (T.F^0(x)) B(\gamma - x) dx \quad (\text{A.11})$$

381 Finally, we derive from (A.9):

$$382 \quad F^i(\gamma) = F^{i-1}(\gamma) + T^{-1}.S \left(G(y) - S^{-1} \int_{-\infty}^{+\infty} (T.F^{i-1}(x)) B(\gamma - x) dx \right) \quad (\text{A.12})$$

383 *Appendix A.2. Super-resolution*

Here, we demonstrate the derivation of the SR method (3) from the restoration method (A.12). Thus, we consider now a sequence of LR images $G_k, 1 \leq k \leq K$ and rewrite (A.12) as:

$$F_k^i(\gamma) = F_k^{i-1}(\gamma) + T_k^{-1}.S \left(G_k(y) - S^{-1} \int_{-\infty}^{+\infty} (T_k.F_k^{i-1}(x)) B_k(\gamma - x) dx \right) \quad (\text{A.13})$$

384 where F_k is the k th SR image, G_k is the k th LR image, T_k is the geometrical
 385 transformation from F_k to G_k , T_k^{-1} is the inverse geometric transformation,
 386 S is the upsampling operator, S^{-1} is the downsampling operator, and B_k the
 387 PSF of k th LR image.

For the construction of the mosaic M , we assume:

$$M = \frac{1}{K} \sum_{k=1}^K F_k \quad (\text{A.14})$$

388 Thus (A.13) becomes:

$$\begin{aligned}
\frac{1}{K} \sum_{k=1}^K F_j^i(\gamma) &= \frac{1}{K} \sum_{j=1}^K \left(F_k^{i-1}(\gamma) + T_k^{-1} \cdot S \left(G_k(y) - S^{-1} \int_{-\infty}^{+\infty} (T_k \cdot F_k^{i-1}(x)) B_k(\gamma - x) dx \right) \right) \\
&\Leftrightarrow M^i(\gamma) = M^{i-1}(\gamma) + \frac{1}{K} \sum_{j=1}^K T_k^{-1} \cdot S \left(G_k(y) - S^{-1} \int_{-\infty}^{+\infty} \left(T_k \cdot F_k^{i-1}(x) \right) B_k(\gamma - x) dx \right) \quad (\text{A.15})
\end{aligned}$$

Assuming that F_k is a cut-out of M , then $T_k F_k = T_k M$:

$$M^i(\gamma) = M^{i-1}(\gamma) + \frac{1}{K} \sum_{k=1}^K T_k^{-1} \cdot S \left(G_k(y) - S^{-1} \int_{-\infty}^{+\infty} (T_k \cdot M^{i-1}(x)) B_k(\gamma - x) dx \right) \quad (\text{A.16})$$

Appendix A.3. Relationship between Convolution at Low and High Resolution

Our objective is to establish the relationship between $S^{-1}(B * F)$ and $(S^{-1} \cdot B) * (S^{-1} \cdot F)$ where $*$ is the convolution operator and S^{-1} is the down-sampling operator by the factor z . Considering the PSF B and the SR image F , then the blurred SR image $\tilde{F}$ is:

$$\tilde{F}(\gamma) = \int_{-\infty}^{+\infty} F(x) B(\gamma - x) dx \quad (\text{A.17})$$

Denoting B_z and F_z as the subsamples of $B(\gamma)$ and $F(\gamma)$ by the factor z . If we neglect the aliasing effect, then:

$$B_z(\gamma) = B(z\gamma) \quad F_z(\gamma) = F(z\gamma) \quad (\text{A.18})$$

Denoting $\tilde{F}_z(\gamma)$ as the result of the convolution $(S^{-1}B) * (S^{-1}F)$:

$$\begin{aligned}
\tilde{F}_z(\gamma) &= \int_{-\infty}^{+\infty} F_z(x) B_z(\gamma - x) dx = \int_{-\infty}^{+\infty} F(zx) B(z(\gamma - x)) dx \\
&= \int_{-\infty}^{+\infty} F(zx) B(z\gamma - zx) dx = \frac{1}{z} \int_{-\infty}^{+\infty} F(zx) B(z\gamma - zx) dzx \quad (\text{A.19})
\end{aligned}$$

394 Replacing $y = zx$:

$$\begin{aligned}\tilde{F}_z(\gamma) &= \frac{1}{z} \int_{-\infty}^{+\infty} F(y)B(z\gamma - y) \, dy = \frac{1}{z} \tilde{F}(z\gamma) \\ &\Leftrightarrow (S^{-1}.B) * (S^{-1}.F) = \frac{1}{z} S^{-1}(B * F)\end{aligned}\tag{A.20}$$

395 The constant $1/z$ corresponds to a normalization factor and in case of a
396 normalized convolution mask it can be neglected. In reality, Eq. (A.20) is an
397 approximation which only holds for small blurs as we did not consider the
398 spectrum folding in Eq. (A.18).

- 399 [1] L. Bar, B. Berkels, M. Rumpf, and G. Sapiro. A variational framework
400 for simultaneous motion estimation and restoration of motion-blurred
401 video. In *Proc IEEE Int Conf Comput Vis*, pages 1–8, 2007.
- 402 [2] D. Barreto, L. Alvarez, R. Molina, A. Katsaggelos, and G. Callicó.
403 Region-based super-resolution for compression. *Multidimensional Syst*
404 *Signal Process*, 18(2-3):59–81, 2007.
- 405 [3] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-
406 resolution from an image sequence. In *European Conf Computer Vision*,
407 volume 2, pages 573–582, 1996.
- 408 [4] M. Ben-Ezra, A. Zomet, and S. Nayar. Video super-resolution using
409 controlled subpixel detector shifts. *IEEE Trans Pattern Anal Mach*
410 *Intell*, 27(6):977–987, 2005.
- 411 [5] J. Benois-Pineau, J. Braquelaire, and A. Ali-Mhammad. Interactive fine
412 object-based segmentation of generic video scenes fro object-based in-
413 dexing. In *Proc Int Workshop Image Analysis for Multimedia Interactive*
414 *Services*, pages 200–203, 2003.
- 415 [6] N. Bose and N. Ahuja. Superresolution and noise filtering using moving
416 least squares. *IEEE Trans Image Process*, 15(8):2239–2248, 2006.

- 417 [7] M. Bronstein, A. Bronstein, M. Zibulevsky, and Y. Zeevi. Blind decon-
418 volution of images using optimal sparse representations. *IEEE Trans*
419 *Image Process*, 14(6):726–736, 2005.
- 420 [8] S. Cho, Y. Matsushita, and S. Lee. Removing non-uniform motion blur
421 from images. In *Proc IEEE Int Conf Comput Vis*, pages 1–8, 2007.
- 422 [9] P. V. Cittert. Zum Einfluß der Spaltbreite auf die Intensitätsverteilung
423 in Spektrallinien. II. *Zeitschrift für Physik*, 69:298–308, 1931.
- 424 [10] S. Dai, M. Yang, Y. Wu, and A. Katsaggelos. Tracking motion-blurred
425 targets in video. In *Proc IEEE Int Conf Image Proc*, pages 2389–2392,
426 2006.
- 427 [11] M. Elad and Y. Hel-Or. A fast super-resolution reconstruction algorithm
428 for pure translational motion and common space-invariant blur. *IEEE*
429 *Trans Image Process*, 10:1187–1193, 2001.
- 430 [12] P. Eren, M. Sezan, and A. Tekalp. Robust, object-based high resolu-
431 tion image reconstruction from low-resolution video. *IEEE Trans Image*
432 *Process*, 6(10):1446–1451, 1997.
- 433 [13] P. Eren and A. Tekalp. Bi-directional 2-d mesh representation for video
434 object rendering, editing and superresolution in the presence of occlu-
435 sion. *Signal Processing: Image Communication*, 18(5):321–336, 2003.
- 436 [14] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar. Advances and chal-
437 lenges in super-resolution. *Int J Imaging Syst Technol*, 14(2):47–57,
438 2004.
- 439 [15] J. M. Geusebroek, A. W. M. Smeulders, and J. van de Weijer. Fast
440 anisotropic gauss filtering. *IEEE Trans Image Process*, 12(8):938–943,
441 2003.
- 442 [16] B. Gunturk, Y. Altunbasak, and R. Mersereau. Multiframe resolution-
443 enhancement methods for compressed video. *IEEE Signal Process Lett*,
444 9(6):170–174, 2002.
- 445 [17] J. Hannuksela, P. Sangi, J. Heikkilä, X. Liu, and D. Doermann. Doc-
446 ument image mosaicing with mobile phones. In *Proc Int Conf Image*
447 *Analysis and Processing*, pages 575–580, 2007.

- 448 [18] S. Har-Noy and T. Nguyen. Lcd motion blur reduction: A signal pro-
449 cessing approach. *IEEE Trans Image Process*, 17(2):117–125, 2008.
- 450 [19] R. Hardie. A fast image super-resolution algorithm using an adaptive
451 wiener filter. *IEEE Trans Image Process*, 16(12):2953–2964, 2007.
- 452 [20] H. He and L. Kondi. A regularization framework for joint blur estimation
453 and super-resolution of video sequences. In *Proc IEEE Int Conf Image*
454 *Proc*, volume III, pages 329–332, 2005.
- 455 [21] Y. He, K. Yap, L. Chen, and L. Chau. Blind super-resolution image
456 reconstruction using a maximum a posteriori estimation. In *Proc IEEE*
457 *Int Conf Image Proc*, pages 1729–1732, 2006.
- 458 [22] H. Hu and G. de Haan. Adaptive image restoration based on local robust
459 blur estimation. In *Advanced Concepts for Intelligent Vision Systems*,
460 pages 461–472, 2007.
- 461 [23] M. Irani and S. Peleg. Motion analysis for image enhancement: Reso-
462 lution, occlusion, and transparency. *J Vis Commun Image Represent*,
463 4:324–335, 1993.
- 464 [24] P. Jansson. Method for determining the response function of a high-
465 resolution infrared spectrometer. *J Opt Soc Am*, 60(2):184–191, 1970.
- 466 [25] J. Jia. Single image motion deblurring using transparency. In *Proc IEEE*
467 *Int Conf Computer Vision and Pattern Recognition*, pages 1–8, 2007.
- 468 [26] T. Komatsu and T. Saito. Super-resolution sharpening-demosaicking
469 method for removing image blurs caused by an optical low-pass filter.
470 In *Proc IEEE Int Conf Image Proc*, volume I, pages 845–848, 2005.
- 471 [27] P. Krämer, O. Hadar, J. Benois-Pineau, and J. Domenger. Super-
472 resolution mosaicing from MPEG compressed video. *Signal Processing:*
473 *Image Communication*, 22(10):845–865, 2007.
- 474 [28] D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE Signal*
475 *Process Mag*, 13(3):43–64, 1996.
- 476 [29] D. Kundur and D. Hatzinakos. Blind image deconvolution revisited.
477 *IEEE Signal Process Mag*, 13(6):61–63, 1996.

- 478 [30] M. Kunter and T. Sikora. Super-resolution mosaicing for object based
479 video format conversion. In *Proc Int Workshop Image Analysis for Mul-*
480 *timedia Interactive Services*, 2006.
- 481 [31] A. Létienne, F. Champagnat, C. Kulcsár, G. Le Besnerais, and P. Viaris
482 De Lesegno. Fast super-resolution on moving objects in video sequences.
483 In *European Signal Processing Conf*, 2008.
- 484 [32] X. Li, Y. Hu, X. Gao, D. Tao, and B. Ning. A multi-frame image super-
485 resolution method. *Signal Processing*, 90(2):405–414, 2010.
- 486 [33] F. Manerba, J. Benois-Pineau, R. Leonardi, and B. Mansencal. Multiple
487 moving object detection for fast video content description in compressed
488 domain. *EURASIP J Adv Signal Process*, 2008(1):1–13, 2008.
- 489 [34] J. Nagy and D. O’Leary. Restoring images degraded by spatially variant
490 blur. *SIAM J Sci Comput*, 19(4):1063–1082, 1998.
- 491 [35] N. Nguyen, P. Milanfar, and G. Golub. A computationally efficient
492 image superresolution algorithm. *IEEE Trans Image Process*, 10(4):573–
493 583, 2001.
- 494 [36] S. Park, M. Park, and M. Kang. Super-resolution image reconstruction:
495 a technical overview. *IEEE Signal Process Mag*, 20(3):21–36, 2003.
- 496 [37] A. Patti, M. Sezan, and A. Tekalp. Robust methods for high quality stills
497 from interlaced video in the presence of dominant motion. *IEEE Trans*
498 *on Circuits and Systems for Video Technology*, 7(2):328–342, 1997.
- 499 [38] A. Patti, M. Sezan, and A. Tekalp. Superresolution video reconstruction
500 with arbitrary sampling lattices and nonzero aperture time. *IEEE Trans*
501 *Image Process*, 6(8):1064–1078, 1997.
- 502 [39] M. Protter, M. Elad, H. Takeda, and P. Milanfar. Generalizing the
503 non-local-means to super-resolution reconstruction. *IEEE Trans Image*
504 *Process*, 18(1):36–51, 2009.
- 505 [40] I. Richardson. *H.264 and MPEG-4 Video Compression*. Wiley, 2003.
- 506 [41] C. Segall, R. Molina, and A. Katsaggelos. High-resolution images from
507 low-resolution compressed video. *IEEE Signal Process Mag*, 20(3):37–
508 48, 2003.

- 509 [42] S. Smith. *The Scientist and Engineer's Guide to Digital Signal Process-*
510 *ing*. California Technical Publishing, 1997.
- 511 [43] M. Sorel and J. Flusser. Space-variant restoration of images degraded by
512 camera motion blur. *IEEE Trans Image Process*, 17(2):105–116, 2008.
- 513 [44] F. Sroubek, G. Cristóbal, and J. Flusser. A unified approach to super-
514 resolution and multichannel blind deconvolution. *IEEE Trans Image*
515 *Process*, 16(9):2322–2332, 2007.
- 516 [45] H. Takeda, P. Milanfar, M. Protter, and M. Elad. Super-resolution
517 without explicit subpixel motion estimation. *IEEE Trans Image Process*,
518 18(9):1958–1975, 2009.
- 519 [46] H. Trussell and S. Fogel. Identification and restoration of spatially vari-
520 ant motion blurs in sequential images. *IEEE Trans Image Process*,
521 1(1):123–126, 1992.
- 522 [47] D. Tull and A. Katsaggelos. Iterative restoration of fast-moving objects
523 in dynamic image sequences. *Opt Eng*, 35(12):3460–3469, 1996.
- 524 [48] A. van Eekeren, K. Schutte, J. Dijk, D. de Lange, and L. van Vliet.
525 Super-resolution on moving objects and background. In *Proc IEEE Int*
526 *Conf Image Proc*, pages 2709–2712, 2006.
- 527 [49] A. van Eekeren, K. Schutte, and L. van Vliet. Multiframe super-
528 resolution reconstruction of small moving objects. *IEEE Trans Image*
529 *Process*, 19(11), 2010.
- 530 [50] VXL. C++ Libraries for Computer Vision Research and Implementa-
531 tion. <http://vxl.sourceforge.net/>.
- 532 [51] L. Yaroslavsky, B. Fishbain, G. Shabat, and I. Ideses. Superresolution
533 in turbulent videos: making profit from damage. *Opt Lett*, 32(20):3038–
534 3040, 2007.

Fig. A.1: (a) Edge response in motion direction of a blurred edge, (b) Computation of the edge response in horizontal and vertical direction.

Fig. A.2: The error measure versus the number of iterations for sequence “Comportements”: (a) for the background mosaic with $\beta_1 = 1$, (b) for the moving object with $\beta_2 = 1$.

Fig. A.3: Mosaicing results for the sequence “Hiragasy” (5 LR images): (a) one image of the LR sequence, (b) the SR mosaic. © CERIMES-SFRS

Fig. A.4: Mosaicing results for the sequence “Comportements” (10 LR images): (a) one image of the LR sequence, (b) the initial SR mosaic, (c) the SR mosaic for the proposed method, (d) the SR mosaic for the accelerate gradient descent, (e) the upsampled LR mosaic after deblurring, (f) the mosaic constructed from downsampled full-resolution frames. © CERIMES-SFRS