KYBERNETIKA — VOLUME 47 (2011), NUMBER 3, PAGES 439–454

MULTICHANNEL DEBLURRING OF DIGITAL IMAGES

Michal Šorel, Filip Šroubek and Jan Flusser

Blur is a common problem that limits the effective resolution of many imaging systems. In this article, we give a general overview of methods that can be used to reduce the blur. This includes the classical multi-channel deconvolution problems as well as challenging extensions to spatially varying blur. The proposed methods are formulated as energy minimization problems with specific regularization terms on images and blurs. Experiments on real data illustrate very good and stable performance of the methods.

Keywords: image restoration, blind deconvolution, deblurring, spatially varying blur

Classification: 15A29, 92C55

1. INTRODUCTION

In this article, we give a general overview of methods that can be used to reduce image blur, with special emphasis on those developed by the authors. This includes the classical deconvolution problems as well as challenging extensions to spatially varying blur. Our treatment of the topic is based on our book chapter [24].

We consider mainly algorithms fusing information from multiple blurred images to get an image of better quality. We do not treat deblurring methods working with one image that need stronger prior knowledge and other than MAP approaches. Nor we consider approaches requiring hardware adjustments such as special shutters (coded-aperture camera [11]), camera actuators (motion-invariant photography [12]) or sensors (Penrose pixels [5]). We focus on our results [21, 22, 23], described in Section 3, and other relevant references are commented in more detail inside the text.

We first introduce a general model of image acquisition needed for the modeling of image blur. This model is later used for deriving a Bayesian solution of the deblurring problem. Next, we briefly discuss possible sources of blur. In each case we also include possible approaches for blur estimation for both space-invariant and space-variant scenarios.

All the common types of generally spatially varying blur, such as defocus, camera motion or object motion blur, can be described by a linear operator H acting on an image u in the form

$$[Hu](x,y) = \int u(x-s, y-t) h(s,t,x-s,y-t)\,\mathrm{d}s\mathrm{d}t\,, \tag{1}$$

where h is a *point spread function* (*PSF)* or *kernel*. We can look at this formula as a convolution with a PSF that changes with its position in the image. The traditional convolution is a special case thereof, with the PSF independent of coordinates x and y. In practice, we work with a discrete representation of images and the same notation can be used with the following differences. Operator H in (1) corresponds to a matrix and u to a vector obtained by stacking columns of the image into one long vector. Each column of H describes the spread of energy for one pixel of the original image. In the case of the traditional convolution, H is a block-Toeplitz matrix with Toeplitz blocks and each column of H contains the same kernel shifted to the appropriate position.

In the space-variant case, each column may contain a different kernel. An obvious problem of spatially varying blur is that the PSF is now a function of four variables. Except trivial cases, it is hard to express it by an explicit formula. Even if the PSF is known, we must solve the problem of efficient representation. If the PSF changes smoothly without discontinuities, we can store the PSF on a discrete set of positions and use interpolation to approximate the whole function h. If the PSF is not known, the local PSF's must be estimated as in the method described in Section 3. Another type of representation is necessary if we consider moving objects, where the blur changes sharply at object boundaries. Then we usually assume that the blur is approximately space-invariant inside individual objects, and the PSF can be represented by a set of convolution kernels for each object and a corresponding set of object contours. Final case occurs when the PSF depends on the depth. If the relation cannot be expressed by an explicit formula, as in the case of ideal pillbox function for defocus, we must store a table of PSF's for every possible depth.

1.1. General model of image degradation

In this section, we show a general model of image acquisition, which comprises commonly encountered degradations. Depending on the application, some of these degradations are known and some can be neglected.

The image u is degraded by several external and internal phenomena. The external effects are, e. g., atmospheric turbulence and relative camera-scene motion, the internal effects include out-of-focus blur, diffraction and all kinds of aberrations. As the light passes through the camera lens, also warping due to lens distortions occurs. Finally, a camera digital sensor discretizes the image and produces a digitized noisy image $g(x, y)$. An acquisition model, which embraces all the above radiometric and geometric deformations, can be written as a composition of operators

$$g = DLHu + n\,. \tag{2}$$

Operator L denotes lens distortions, blurring operator H describes the external and internal radiometric degradations, D is an operator modeling the camera sensor and n stands for additive noise. The operator D is a filter originating as a result of diffraction, shape of light sensitive elements and void spaces between them. We will assume that the form of D is known. Our goal is to solve an inverse problem, i. e., to estimate u from the observation g.

Many restoration methods assume that the blurring operator H is known, which is rarely true in practice and it is indispensable to assume at least that H is a traditional convolution with an unknown PSF. This model holds for some types of blurs (see e. g.[25]). In our work, we went one step further and allowed spatially varying blur, which is the most general case that encompasses all the above mentioned radiometric degradations if occlusion is not considered. On the other hand, without additional constraints, the space-variant model is too complex. Various scenarios that are space-variant but still solvable are discussed in Section 3.

If lens parameters are known, one can remove lens distortions L from the observed image g without affecting blurring H, since H precedes L in (2). There is a considerable amount of literature on estimation of distortion [1, 27]. If the lens distortion is known or estimated beforehand, we can include L inside the operator D or it can be consider as a part of the unknown blurring operator H and estimated during the deconvolution process. In any case, we will not consider L explicitly in the model (2) from now on.

In many cases, only one input image is not sufficient to get a satisfactory result and we assume that multiple images of the same scene are available. Having several input images, we will denote the quantities belonging to the kth image by index k. To describe this situation, we use the term multichannel or K-channel in the rest of this text. To be able to describe the common real situation of several images taken by the same camera from slightly different viewpoints, we need to introduce an unknown geometric deformation W_k for each image, which gives us

$$g_k = DH_kW_ku + n_k\,, \tag{3}$$

with D remaining the same in all the images. Deformations W_k can be estimated by a proper image registration method [29]. To be able to work with these pre-registered input images, we need to interchange the order of operators H_k and W_k, which gives

$$g_k = DW_k\tilde{H}_ku + n_k = D_k\tilde{H}_ku + n_k\,. \tag{4}$$

On the right hand side of this formula, we denote the combined operator of W_k and D as $D_k = DW_k$ and assume it is known.

We need to show that the operators H_k and W_k can be really interchanged. Indeed, if the geometric transform W_k is invertible and we consider the blurring operator $\tilde{H}_k = W_k^{-1}H_kW_k$, we get $W_k\tilde{H}_k = W_kW_k^{-1}H_kW_k = H_kW_k$. Moreover, if H_k is a standard convolution with a PSF h_k and W_k denotes a linear geometric transform, then by placing W_k in front of H_k, the new blurring operator $\tilde{H}_k$ remains a standard convolution but with h_k warped according to W_k. If W_k denotes a nonlinear geometric transform, then after interchanging the order, $\tilde{H}_k$ becomes a sparse linear operator that can no longer be described by convolution. It is important to realize that the blurring operator is unknown and instead of H_k we are estimating $\tilde{H}_k$, which is an equivalent problem as long as the nature of both blurring operators remains the same. To avoid extra symbols, we keep the symbol H_k instead of more exact $\tilde{H}_k$ from now on and we will also denote the full degradation operator as $G_k = D_k\tilde{H}_k$.

1.2. Bayesian view of solution

In this paper, we approach the deblurring problem from the Bayesian viewpoint. Other approaches can be considered as approximations of the Bayesian solution.

If we know the degradation operators G_k, the MAP (maximum a posteriori) solution under the assumption of Gaussian noise corresponds to the minimum of a functional

$$E(u) = \sum_k \frac{1}{2\sigma_k^2} \|G_k u - g_k\|^2 + Q(u), \tag{5}$$

where the first term describes an error of our model and the second term $Q(u)$ is a so-called regularization term that corresponds to the negative logarithm of the prior probability of the image u. Noise variance in the kth input image is denoted as σ_k.

The prior probability is difficult to obtain and it is often approximated by a statistics of the image gradient distribution. A good approximation of the prior log-probability for common images is for example total variation regularization [16]

$$Q(u) = \lambda \int |\nabla u| \,, \tag{6}$$

which corresponds to an exponential decay of gradient magnitude. The total variation term can be replaced by an arbitrary suitable regularizer (Tikhonov, Mumford-Shah, etc.) [2, 20].

To minimize functional (5) we can use many existing algorithms, depending on a particular form of the regularization term. If it is quadratic (such as the classical Tikhonov regularization), we can use an arbitrary numerical method for solving the system of linear equations. In the case of total variation, the problem is usually solved by transforming the problem to a sequence of linear subproblems. In our implementations, we use the half-quadratic iterative approach as described for example in [22].

The derivative of functional (5) with the total variation regularizer (6) can be written as

$$\frac{\partial E(u)}{\partial u} = \sum_k \frac{G_k^*(G_k u - g_k)}{\sigma_k^2} - \lambda \mathrm{div}\left(\frac{\nabla u}{|\nabla u|}\right). \tag{7}$$

$G_k^* = H_k^* D_k^*$ is an operator adjoint to G_k. The operator adjoint to H_k defined in (1) can be written as

$$[H^* u](x, y) = \int u(x - s, y - t) h(-s, -t, x, y) \, \mathrm{d}s\mathrm{d}t. \tag{8}$$

We can imagine this correlation-like operator as putting the PSF to all image positions and computing dot product. The same relation holds for D_k^*, which corresponds to the convolution with the original PSF rotated by 180 degrees.

If we know the operators G_k, the solution is in principle known, though the implementation of the above formulas can be quite complicated. In practice however, the operators G_k are not known and must be estimated.

Especially in the case of spatially varying blur, it turns out to be indispensable to have at least two observations of the same scene, which gives us additional information that makes the problem more tractable. Moreover, to solve such a complicated

ill-posed problem, we must exploit the internal structure of the operator, according to the particular problem we solve. Some parts of the composition of sub-operators in (2) are known, some can be neglected or removed separately – for example geometrical distortion. All the above cases are elaborated in Section 3.

Without known PSF's it is in principle impossible to register precisely images blurred by motion. Consequently, it is important that image restoration does not necessarily require pixel precision of the registration. The registration error can be compensated in the algorithm by shifting the corresponding part of the space-variant PSF. Thus the PSF estimation provides robustness to misalignment. As a side effect, misalignment due to lens distortion does not harm the algorithm as well.

In general, if each operator $G_k = G(\boldsymbol{\theta}_k)$ depends on a set of parameters $\boldsymbol{\theta}_k = \{\theta_k^1, \ldots, \theta_k^P\}$, we can again solve the problem in the MAP framework and maximize the joint probability over u and $\{\boldsymbol{\theta}_k\} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$. As the image and degradation parameters can be usually considered independent, the negative logarithm of probability gives a similar functional

$$E(u, \{\boldsymbol{\theta}_k\}) = \sum_{k=1}^{K} \frac{1}{2\sigma_k^2} \|G(\boldsymbol{\theta}_k)u - g_k\|^2 + Q(u) + R(\{\boldsymbol{\theta}_k\}) \,, \tag{9}$$

where the additional term $R(\{\boldsymbol{\theta}_k\})$ corresponds to a (negative logarithm of) prior probability of degradation parameters. The derivative of the error term in (9) with respect to the ith parameter θ_k^i of $\boldsymbol{\theta}_k$, equals

$$\frac{\partial E(u, \{\boldsymbol{\theta}_k\})}{\partial \theta_k^i} = \frac{1}{\sigma_k^2} \left\langle \frac{\partial G(\boldsymbol{\theta}_k)}{\partial \theta_k^i} u, G(\boldsymbol{\theta}_k)u - g_k \right\rangle + \frac{\partial R(\{\boldsymbol{\theta}_k\})}{\partial \theta_k^i} \,, \tag{10}$$

where $\langle \cdot \rangle$ is the standard inner product in L_2. In discrete implementation, $\frac{\partial G(\boldsymbol{\theta}_k)}{\partial \theta_k^i}$ is a matrix that is multiplied by the vector u before computing the inner product with the residual error.

Each parameter vector $\boldsymbol{\theta}_k$ can contain registration parameters for images, PSF's, depth maps, etc. according to the type of degradation we consider.

Unfortunately in practice, it may be difficult to minimize the functional (9), especially in the case of spatially varying blur. Details are discussed in [24].

An alternative to MAP approach is to estimate the PSF in advance and then proceed with (non-blind) restoration by minimization over the possible images u. This can be regarded as an approximation to MAP. One such approach is demonstrated in Section 3.2.

We should also note that MAP approach may not give optimal results, especially if we do not have enough information and the prior probability becomes more important. This is a typical situation for blind deconvolution of one image. It was documented (blind deconvolution method [8] and analysis [11]) that in these cases marginalization approaches can give better results. On the other hand, in the case of multiple input images, which is discussed in this article, the MAP approach is usually appropriate.

2. POINT-SPREAD FUNCTIONS

This section discusses the shape of the PSF for the most frequent types of blur and indicates the relation of the involved PSF to camera parameters, camera motion and three-dimensional structure of an observed scene; for details, see [24].

To model defocus, image processing applications widely use a simple model based on geometrical optics, where the shape of the PSF corresponds to a circular spot, often called informally pillbox, with a radius inversely proportionally to the distance from the plane of focus.

In practice, due to lens aberrations and diffraction effects, PSF will be a circular blob, with brightness falling off gradually rather than sharply. Therefore, most algorithms use two-dimensional Gaussian function with a limited support instead of a sharply cut pillbox-like shape. Gaussian shapes are adequate for good quality lenses or in the proximity of the image center, where the optical aberrations are usually well corrected. A more precise approach is to consider optical aberrations. However, an issue arises in this case that aberrations must be described for the whole range of possible focal lengths, apertures and planes of focus. In practice, it is also useful to take diffraction effects into account as many cameras are close to their diffraction limits.

Another important type of blur is the blur caused by camera shake or, in general, the blur caused by camera motion during exposure. To model this situation, we need to express the PSF as a function of the camera motion and distance from camera. In the case of a general camera motion, it can be computed from the formula for *velocity field* [9, 22] that gives apparent velocity of the scene for the point (x, y) of the image at time instant τ as

$$v(x,y,\tau) = \frac{1}{\mathrm{d}(x,y,\tau)} \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \end{bmatrix} T(\tau) + \begin{bmatrix} xy & -1-x^2 & y \\ 1+y^2 & -xy & -x \end{bmatrix} \Omega(\tau), \quad (11)$$

where $\mathrm{d}(x, y, \tau)$ is the depth corresponding to point (x, y) and $\Omega(\tau)$ and $T(\tau)$ are three-dimensional vectors of rotational and translational velocities of the camera at time τ. Both vectors are expressed with respect to the coordinate system originating in the optical center of the camera with axes parallel to x and y axes of the sensor and to the optical axis. All the quantities, except $\Omega(\tau)$, are in focal length units. The *depth* $\mathrm{d}(x, y, \tau)$ is measured along the optical axis. The function d, for a fixed τ, is called *depth map*.

The apparent curve $[\bar{x}(x, y, \tau), \bar{y}(x, y, \tau)]$ drawn by the given point (x, y) can be computed by the integration of the velocity field over the time when the shutter is open. Having the curves for all the points in the image, the two-dimensional space-variant PSF can be expressed as

$$h(s,t,x,y) = \int \delta(s - \bar{x}(x,y,\tau), t - \bar{y}(x,y,\tau))\,\mathrm{d}\tau, \quad (12)$$

where δ is the two-dimensional Dirac delta function.

Analytical form of (12) is usually not used directly, because the analytical forms of velocity vectors $\Omega(\tau)$ and $T(\tau)$ are not available. Instead, our algorithms [22, 23]

use a discrete representation extending standard convolution masks (see Section 3.2). The analytical form is in a sense used in the recent single-image deconvolution paper [26].

In addition, in real algorithms, it is necessary to use certain assumptions on these vectors, which simplifies the computations. The key assumption we use for the blur caused by camera shake is that camera translations can be neglected $T(\tau) = 0$, which means that the velocity field is independent of depth and changes slowly and without discontinuities. Consequently, the blur can be considered locally constant and can be locally approximated by convolution. This property can be used to efficiently estimate the space-variant PSF, as described in Section 3.2.

A more complicated special case it to disallow camera rotation and assume that the change of depth is negligible with an implication that also the velocity in the direction of view can be considered zero. It can be easily seen [22] that in this special case, the PSF can be expressed explicitly using the knowledge of the PSF for one fixed depth of scene.

In many real scenarios, the observed scene is not static but contains moving objects. Local movements cause occlusion of the boundary and an additional varying blur. To include these two phenomena in the acquisition model is complicated as it requires segmentation based on motion detection. Most restoration methods assume a rigid transform (e. g. homography) as the warping operator W in (4). If the registration parameters can be calculated, we can spatially align input images. If local motion occurs, the warping operator must implement a non-global transform, which is difficult to estimate. In addition, warping by itself cannot cope with occlusion. A reasonable approach is to segment the scene according to results obtained by local-motion estimation and deal with individual segments separately. Several attempts in this direction were explored in literature recently. Since PSF's may change abruptly, it is essential to precisely detect boundaries, where the PSF's change, and consider boundary effects. An attempt in this direction was for example proposed in [3], where level-sets were utilized. Another interesting approach is to identify blurs and segment the image accordingly by using local image statistics as proposed, e. g., in [10].

3. ALGORITHMS

This section describes several types of deblurring algorithms based on the MAP framework explained in the introduction, with accent on our results [18, 22, 23].

We will progress from simple to more complex scenarios, where we need to estimate a higher number of unknown parameters. The simplest case is the space-invariant blur. An algorithm of this type, originally published in [18], is described in Section 3.1. If the blur is caused by a complex camera movement, it generally varies across the image but not randomly. The PSF is constrained by six degrees of freedom of a rigid body motion. Moreover, if we limit ourselves to only camera rotation, we not only get along with three degrees of freedom, but we also avoid the dependence on a depth map. This case in multi-image scenario is treated in Section 3.2, which describes mainly our algorithm [23]. A recent single-image camera shake

deblurring algorithm can be found in [26]. If the PSF depends on the depth map, the problem becomes more complicated. Section 3.3 indicates possible solutions for two such cases: defocus with a known optical system and blur caused by camera motion. In the latter case, the camera motion must be known or we must be able to estimate it from the input images [22].

3.1. Space-invariant blind deconvolution

We assume the K-channel acquisition model in (4) with H_k being traditional convolution with an unknown PSF h_k. The corresponding functional to minimize is (9) where $\{\boldsymbol{\theta}_k\} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$ comprises registration parameters and PSF's h_k. If the acquired images g_k are not ideally registered, operators D_k must compensate for geometric misalignments. Since we restrict ourselves to the space-invariant case, admissible geometric transformations are only linear (at most affine) otherwise the nature of H_k would change from space-invariant to space-variant after registration. The space-invariant case allows us to construct an intrinsically multichannel regularization term $R(\{\boldsymbol{\theta}_k\})$, which is a function of h_k's and utilizes relations between all the input images. An exact derivation is given in [18]. Here, we leave the discussion by stating that it is of the form

$$R(\{h_k\}) \propto \sum_{\substack{1 \le i,j \le K \\ i \neq j}} \|h_i * g_j - h_j * g_i\|^2 , \tag{13}$$

with the asterisk denoting the operation of convolution. This term is positive and convex and if no noise is present it is equal to zero for any set of kernels $\{f * h_k\}$, where h_k is the true kth PSF and f is an arbitrary function. It means, that R is zero in the correct solution but there are infinitely many other kernels with the same property. This drawback can be eliminated by forcing positivity on PSFs and limiting their maximum allowed size.

One image from the input sequence is selected as a reference image g_r ($r \in 1, \ldots, K$) and registration is performed with respect to this image. If the camera position changes slightly between acquisitions, we assume affine model. The algorithm runs in two steps:

1. Initialize parameters $\{\boldsymbol{\theta}_k\}$: Estimate affine transformations between the reference frame g_r and each g_k for $k \in 1, \ldots, K$. Construct accordingly decimation operators D_k. Initialize $\{h_k\}$ with delta functions.

2. Minimization of $E(u, \{\boldsymbol{\theta}_k\})$ in (9): alternate between minimization with respect to u and with respect to $\{h_k\}$. Run this step for a predefined number of iterations or until a convergence criterion is met.

For images blurred by PSF's larger than about 20 pixels, the convergence of the minimization slows down and a hierarchical (also called multiscale) approach is necessary. The input images are first downsampled to several predefined scales . We start with the coarsest scale (smallest images) and run the deconvolution (Step 2) on it. The estimated PSF's correspond to the given scale and their support is thus

much smaller. Then, we upsample the estimated PSF's to the next scale and use them as initial values for the deconvolution on this scale. This procedure repeats until we reach the scale of the original input images.

Fig. 1. Multichannel blind deconvolution of a scene degraded by space-invariant blurs: The two photos (1800×1080 pixels) on the top acquired by a common digital camera are blurred due to camera shake. The bottom left image shows the estimated sharp image from the two blurry ones using the blind deconvolution algorithm. The bottom right image shows the estimated PSF's (50×30 pixels). **Best viewed electronically.**

Performance of the aforementioned algorithm was tested on common digital cameras. We conducted several experiments with handheld digital cameras shooting under low light conditions, which produce images blurred by camera shake. We took two or more images in a row in order to have multiple acquisitions of the given scene and then we applied the deconvolution algorithm. One such example of blind deconvolution from two blurred images is given in Figure 1. The input images (1800×1080 pixels) shown in the top row are heavily blurred by camera motion. The PSF extends over 40 pixels, which is too wide to apply the deconvolution algorithm directly on the input images in their original scale. However, the multiscale approach provides a stable solution. Notice that the estimated PSF's (bottom right) match the shape of the trajectories of several pinheads in the input images (see also the lower-left close-up in Figure 2). Likewise, the estimated image (bottom left) is sharp and without any artifacts. Three close-ups of the first input image and the corresponding parts in the estimated image are illustrated in Figure 2.

Fig. 2. Close-ups of images in Figure 1: The left column shows three different details of the first input image and the right column shows the corresponding sections in the estimated output image. **Best viewed electronically.**

3.2. Smoothly changing blur

In many situations, the blur is spatially variant. This section treats the space-variant restoration in situations where the PSF changes without sharp discontinuities, which means that the blur can be locally approximated by convolution. A typical case is the blur caused by camera shake, when the rotational component of camera motion is usually dominant and consequently, according to (11), the blur does not depend on the depth map. On the other hand, the PSF can be significantly spatially variant, especially for short focal length lenses.

Fig. 3. A night photo (1550 × 980 pixels) taken from hand with shutter speed 2.5s (left). The same photo was taken once more at ISO 1600 with 2 stops under-exposure to achieve a hand-holdable shutter time 1/30s (middle). The right image shows the result of algorithm [23] that fuses the information from both. **Best viewed electronically.**

In principle, in this case, known blind deconvolution methods such as [17] could be applied locally and the results could be fused. Unfortunately, it is not easy to put the patches together without artifacts on the seams, not mentioning time complexity. An alternative way is to use first the estimated PSF's to approximate the spatially varying PSF by interpolation of adjacent kernels and then compute the image of a better quality by the minimization of the functional (5). The main problem of these procedures is that they are relatively slow, especially if applied on too many positions. We investigated the latter idea for the purpose of image stabilization in [23] using a special setup that simplifies the involved computations and makes them more stable.

We assume that the user can set the exposure time of the camera, which is an acceptable assumption as we can always balance noise with motion blur by setting a suitable shutter speed. In particular, we set the exposure time of one of the images to be so short, that the image is sharp, of course at the expense of underexposure, which is equivalent to noise amplification. The whole idea was investigated relatively recently [13, 19, 28].

On the left side of Figure 3, we can see a night photo of a historical building taken at ISO 100 with shutter speed 2.5s. The same photo was taken once more at ISO 1600 with 2 stops under-exposure to achieve a hand-holdable shutter time 1/30s. The algorithm detailed in [23] fuses the information they contain to get one sharp noiseless picture.

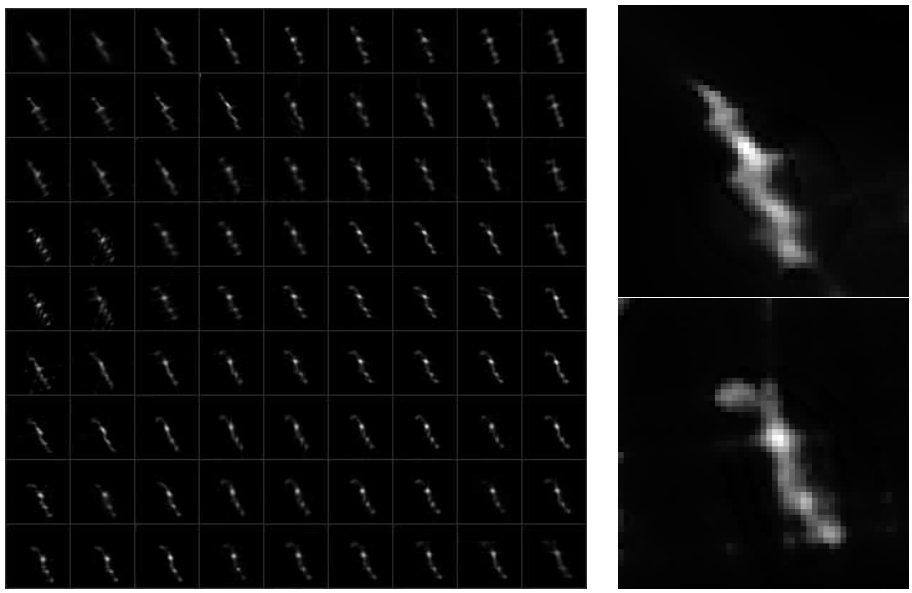

Fig. 4. If the blur changes gradually, we can estimate convolution kernels on a grid of positions (9×9) and approximate the PSF in the rest of the image by interpolation from four adjacent kernels. The right side shows two of the computed PSFs from upper-left and bottom-right corners of the image. The size of the PSFs is 51×51 elements.

At the input, we have two images g_1 and g_2, the first one blurred and the second underexposed and noisy because of the high ISO setting. The model (4) has a simple form – the operator D is identity for both images and blurring operator H_2 is identity for the noisy image. The geometric deformation is removed in the registration step.

The algorithm works in the following three phases. The first is a rough image registration. Note that precise registration is not possible for principle reasons because of ambiguity as discussed in Section 1.2. On the other hand this fact does not harm the algorithm because the registration error is compensated by the shift of the corresponding part of the PSF. The second step is the estimation of convolution kernels on a grid of windows (Figure 4 left) followed by an adjustment at places where the estimation failed. It means that instead of estimating the whole function h in (12), we represent it by a set of standard convolution masks. Finally, we get the sharp and almost noiseless image by minimizing the functional (5). The PSF described by the operator H for the blurred image is approximated by interpolation from the kernels estimated in the previous step.

The second step is a critical part of the algorithm. In the example in Figure 3, we took $9 \times 9 = 81$ square sub-windows, in which we estimated corresponding convolution kernels. The blur kernel corresponding to each square is calculated as a least

squares fit between a patch in the noisy image and the corresponding patch in the blurred image, all this subject to non-negativity constrain. The estimated kernels are assigned to centers of the windows where they were computed. In the rest of the image, the PSF is approximated by bilinear interpolation from blur kernels in the four adjacent sub-windows [14]. Note that such interpolation is not physically correct and we could get better results by a kind of warping. The time consumption of the algorithm would nevertheless significantly grow up. The kernel estimation procedure can naturally fail, either because of a lack of texture or because of pixel saturation. In [23], we identify such kernels and replace them by the average of the nearest neighbors. For the identification, we use two simple measures – sum of the kernel values and kernel entropy.

For minimization of the functional (5), we used a variant of the half-quadratic iterative approach, solving iteratively a sequence of linear subproblems, as described for example in [22]. Note that the blurring operator can be speeded up by Fourier transform computed separately on each square corresponding to the neighborhood of four adjacent PSF's [14]. Instead of the half-quadratic approach, we could use one of faster versions of the iterative shrinkage algorithm, such as [4].

Finally, we should mention that we actually do not solve the problem in the strict MAP sense but results show that the chosen approach is sufficient. Details of the algorithm can be found in [23].

3.3. Depth-dependent blur

Certain types of blur, such as defocus and the blur caused by camera motion depend on the distance from camera (depth). Then, if the scene contains significant depth variations, the methods requiring PSF without discontinuities are not suitable. Artifacts would appear especially at the edges of objects. In this case, it is indispensable to estimate both the unknown image and depth map. In the MAP framework, it can be done again by minimization of a functional in the form (9), with the parameter vector $\{\boldsymbol{\theta}_k\}$ containing the whole depth map.

First such approach appeared for out-of-focus images in [15] proposing to use simulated annealing to minimize the corresponding cost functional. This guarantees global convergence, but in practice, it is prohibitively slow. Later, this approach was adopted by Favaro et al. [6] who modeled the camera motion blur by a Gaussian PSF, locally deformed according to the direction and extent of blur. To make the minimization feasible, they took advantage of special properties of Gaussian PSF's as to view the corresponding blur as an anisotropical diffusion. This model can be appropriate for small blurs corresponding to short locally linear translations. An extension of [6] proposed in [7] segments moving objects but it keeps the limitations of the original paper concerning the shape of the PSF.

The main assumption of existing algorithms is that the relation between the PSF and the depth is known. An exception is our paper [22], where this relation is estimated for a camera motion constrained to movement in one plane without rotation.

4. CONCLUSION

To summarize, the general problem of the restoration of images blurred by spatially varying blur still has not been fully resolved. In this paper, we went through the special cases where at least a partial solution is known and explained the basic principles our published algorithms are based on.

Many open questions and unresolved problems remain. Especially challenging is the situation when the PSF is not continuous, e. g. the case of several independently moving objects (motion blur) or if the PSF depends on the depth of scene (defocus, camera motion).

ACKNOWLEDGEMENT

This work has been supported by the Czech Ministry of Education, Youth, and Sports under the project 1M0572 (Research Center DAR).

(Received November 9, 2010)

REFERENCES

[1] M. Ahmed and A. Farag: Non-metric calibration of camera lens distortion. In: Proc. Internat. Conf. of Image Processing 2001, Vol. 2, pp. 157–160.

[2] M. R. Banham and A. K. Katsaggelos: Digital image restoration. IEEE Signal Process. Mag. *14* (1997), 2, 24–41.

[3] L. Bar, N. A. Sochen, and N. Kiryati: Restoration of images with piecewise space-variant blur. In: SSVM 2007, pp. 533–544.

[4] A. Beck and M. Teboulle: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. Trans. Image Process. *18* (2009), 11, 2419–2434.

[5] M. Ben-Ezra, Z. Lin, and B. Wilburn: Penrose pixels super-resolution in the detector layout domain. In: Proc. IEEE Internat. Conf. Computer Vision 2007, pp. 1–8.

[6] P. Favaro, M. Burger, and S. Soatto: Scene and motion reconstruction from defocus and motion-blurred images via anisothropic diffusion. In: ECCV 2004 (T. Pajdla and J. Matas, eds.), Lecture Notes in Comput. Sci. *3021*, Springer Verlag, Berlin – Heidelberg 2004, pp. 257–269.

[7] P. Favaro and S. Soatto: A variational approach to scene reconstruction and image segmentation from motion-blur cues. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition 2004, Vol. 1, pp. 631–637.

[8] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman: Removing camera shake from a single photograph. ACM Trans. Graph. *25* (2006), 3, 787–794.

[9] D. J. Heeger and A. D. Jepson: Subspace methods for recovering rigid motion. Internat. J. Computer Vision *7* (1992), 2, 95–117.

[10] A. Levin: Blind motion deblurring using image statistics. In: NIPS 2006, pp. 841–848.

[11] A. Levin, R. Fergus, F. Durand, and W. T. Freeman: Image and depth from a conventional camera with a coded aperture. ACM Trans. Graph. *26* (2007), 3, 70.

[12] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman: Motion-invariant photography. In: SIGGRAPH '08: ACM SIGGRAPH 2008 Papers, ACM, New York 2008, pp. 1–9.

[13] S. H. Lim and A. D. Silverstein: Method for Deblurring an Image. US Patent Application, Pub. No. US2006/0187308 A1, 2006.

[14] J. G. Nagy and D. P. O'Leary: Restoring images degraded by spatially variant blur. SIAM J. Sci. Comput. *19* (1998), 4, 1063–1082.

[15] A. N. Rajagopalan and S. Chaudhuri: An MRF model-based approach to simultaneous recovery of depth and restoration from defocused images. IEEE Trans. Pattern Anal. Mach. Intell. *21* (1999), 7, 577–589.

[16] L. I. Rudin, S. Osher, and E. Fatemi: Nonlinear total variation based noise removal algorithms. Physica D *60* (1992), 259–268.

[17] F. Šroubek and J. Flusser: Multichannel blind iterative image restoration. IEEE Trans. Image Process. *12* (2003), 9, 1094–1106.

[18] F. Šroubek and J. Flusser: Multichannel blind deconvolution of spatially misaligned images. IEEE Trans. Image Process. *14* (2005), 7, 874–883.

[19] M. Tico, M. Trimeche, and M. Vehvilainen: Motion blur identification based on differently exposed images. In: Proc. IEEE Internat. Conf. Image Processing 2006, pp. 2021–2024.

[20] D. Tschumperlé and R. Deriche: Vector-valued image regularization with pdes: A common framework for different applications. IEEE Trans. Pattern Analysis Machine Intelligence *27*(2005), 4, 506–517.

[21] M. Šorel: *Multichannel Blind Restoration of Images with Space-Variant Degradations.* PhD Thesis, Charles University, Prague 2007.

[22] M. Šorel and J. Flusser: Space-variant restoration of images degraded by camera motion blur. IEEE Trans. Image Process. *17* (2008), 2, 105–116.

[23] M. Šorel and F. Šroubek: Space-variant deblurring using one blurred and one underexposed image. In: Proc. Internat. Conf. Image Processing, 2009, pp. 157–160.

[24] M. Šorel, F. Šroubek, and J. Flusser: Towards superresolution in the presence of spatially varying blur. In: Super-Resolution Imaging (P. Milanfar, eds.), CRC Press 2010.

[25] F. Šroubek, G. Cristobal, and J. Flusser: A unified approach to superresolution and multichannel blind deconvolution. IEEE Trans. Image Process. *16* (2007), 2322–2332.

[26] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce: Non-uniform deblurring for shaken images. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition 2010.

[27] W. Yu: An embedded camera lens distortion correction method for mobile computing applications. IEEE Trans. Consumer Electronics *49* (2003), 4, 894–901.

[28] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum: Image deblurring with blurred/noisy image pairs. In: SIGGRAPH '07: ACM SIGGRAPH 2007 Papers, ACM, New York 2007, p. 1.

[29] B. Zitová and J. Flusser: Image registration methods: a survey. Image and Vision Computing *11* (2003), 21, 977–1000.

Michal Šorel, Institute of Information Theory and Automation – Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 4, 182 08 Praha 8. Czech Republic.
e-mail: sorel@utia.cas.cz

Filip Šroubek, Institute of Information Theory and Automation – Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 4, 182 08 Praha 8. Czech Republic.
e-mail: sroubekf@utia.cas.cz

Jan Flusser, Institute of Information Theory and Automation – Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 4, 182 08 Praha 8. Czech Republic.
e-mail: flusser@utia.cas.cz