

Restoration of Multiple Images with Motion Blur in Different Directions

Alex Rav-Acha Shmuel Peleg
School of Computer Science and Engineering
The Hebrew University of Jerusalem
91904 Jerusalem, ISRAEL

Abstract

Images degraded by motion blur can be restored when several blurred images are given, and the direction of motion blur in each image is different.

Given two motion blurred images, best restoration is obtained when the directions of motion blur in the two images are orthogonal.

Motion blur at different directions is common, for example, in the case of small hand-held digital cameras due to fast hand trembling and the light weight of the camera.

Restoration examples are given on simulated data as well as on images with real motion blur.

1 Introduction

Blurred images can be restored when the blur function is known [1]. Restoration of a single motion-blurred image without prior knowledge of the blur function is much harder. Early deblurring methods treated blurs that can be characterized by a regular pattern of zeros in the frequency domain such as uniform motion blur [9]. More recent methods deal with a wider range of blurs, but require strong assumptions on the image model. For example, assuming that the image is spatially isotropic [12], or can be modeled as an autoregressive process [7]. A summary and analysis of many methods for “blind deconvolution” can be found at [5]. In case that the image motion is constant for the entire imaging period, the motion blur can be inferred from motion analysis and used for restoration [11, 10, 2, 6].

Unfortunately, the assumption of constant motion during the entire imaging process does not hold for many cases of motion blur. For example, analysis of images taken with small digital cameras shows that consecutive images covering the same scene have different motion blur. In particular, the direction of motion blur is different from one image to another due to trembling of the hand.

In [8] the image restoration algorithm included an esti-

mation of the PSF (Point Spread Function) from two images. However, it assumes a pure translation between the images, and uses the location of singularities in the frequency domain which are not stable.

In this paper we describe how different images, each degraded by a motion blur in a different direction, can be used to generate a restored image. It is assumed that the motion blur can be described by a convolution with a one dimensional kernel. No knowledge is necessary regarding the actual motion blur other than its direction which is pre-computed either by one of the existing methods [9, 12], or using the scheme offered in this paper. The relative image displacements can be image translations and image rotations.

2 A Model for Motion Blur

Let g denote the observed image, degraded by a motion blur with a one dimensional kernel $m = (m_1, \dots, m_K)$ at an angle α . Let f be the original image. We assume that f was degraded in the following way.

$$g(x, y) = f \overset{\alpha}{*} m \stackrel{\text{def}}{=} \\ = \sum_{k=0}^{K-1} m_k \cdot f(x + k \cos(\alpha), y + k \sin(\alpha))$$

This assumption is valid when the motion blur is uniform for the entire image. Otherwise, the image can be divided into regions having approximately a constant motion blur. For a discrete image f , interpolation is used to estimate the gray levels at non-integer locations.

3 Deblurring in the Spatial Domain

Using two images for deblurring requires alignment between them. However, accurate alignment can be made only by accounting for the motion blur as seen in Fig. 1. This section describes the algorithm for deblurring two images degraded by motion blur in the spatial domain. Both the alignment between the images, and the deblurring are

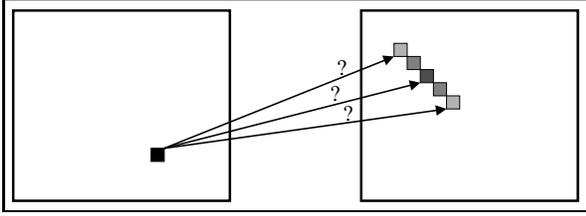


Figure 1. With motion blur the correspondence between images is fuzzy. It can be described by the convolution matrix that turns the left image into the right image.

done simultaneously. In the first sub-section we assume that one image is not blurred. In practice we do not need to restore a blurred image when the original image is given. However, we present this case since it is used as a basis for the deblurring method described in the following sub-section. The last sub-section describes a method for the recovery of the motion blur directions.

3.1 Deblurring an Image Using the Original Image

Let f and g be two input images. g is a motion-blurred image obtained from f as follows:

(i) f' is a warped version of f

$$f'(x, y) = f(x + p(x, y), y + q(x, y)) \quad (1)$$

(ii) g is a degradation of f' by a motion blur with kernel m and direction α ,

$$g = f' \overset{\alpha}{*} m.$$

It can be shown [4] that the desired displacement $(p(x, y), q(x, y))$ between images f and f' minimizes the following error function in the region of analysis R .

$$Err(p, q) = \sum_{(x, y) \in R} (pf_x + qf_y + f_t)^2, \quad (2)$$

where the partial derivations are as follows:

$$\begin{aligned} f_x &= \frac{\partial f}{\partial x}, f_y = \frac{\partial f}{\partial y}, \\ f_t &= \frac{\partial f}{\partial t} = f - f' = f - g \overset{\alpha}{*} m^{-1}. \end{aligned} \quad (3)$$

We assume that the motion blur operation is invertible, and can be approximated by a convolution with a discrete kernel denoted as m^{-1} . In practice, a one dimensional vector with 16 to 32 elements was found to be sufficient for deblurring.

2-D parametric transformations are used as an approximation for the image motion. This approximation is valid when the differences in depth are small relative to the distances from the camera. Since the direction α of the motion blur is pre-computed, the resulting minimization equations

are linear, and minimization is performed over the deblurring kernel (m^{-1}) and the image displacement parameters. We used one of the following models for image displacement:

1. **Translation:** 2 motion parameters, $p(x, y) = a$, $q(x, y) = b$. In order to minimize $Err(p, q)$, its derivatives with respect to a and b are set to zero. This yields two linear equations for each image point in the $K + 2$ unknowns: K is the size of the deblurring kernel, and the two additional parameters represents the translation.
2. **Translation & Rotation:** This model of motion is described by the following equations.

$$p(x, y) = (\cos(\beta) - 1)x - \sin(\beta)y + a$$

$$q(x, y) = \sin(\beta)x + (\cos(\beta) - 1)y + b$$

for small rotations we use the approximations $\cos(\beta) \approx 1$ and $\sin(\beta) \approx \beta$, to obtain linear equations with 3 parameters. $p(x, y) = a - \beta y$, $q(x, y) = b + \beta x$. For each image point we get three linear equations with $K + 3$ unknowns.

3. More complicated models for image displacement can be used, e.g. an **Affine motion** or an **Homography**.

The computation framework is based on multiresolution and iterations, using a Gaussian pyramid, similar to the framework described in [3], with some main differences:

- The deblurring kernel is computed as well as the motion parameters.
- The number of parameters varies throughout the different levels of the pyramid, since the deblurring of upper levels of the Gaussian pyramid can be represented by smaller kernels.
- Different kernels, related by a convolution with a shifted delta-impulse are equivalent in the above framework. Therefore, the motion component parallel to the motion-blur direction can not converge. In order to handle this variant, the iterations are repeated until convergence only in the motion component perpendicular to the direction of the motion blur.

3.2 Deblurring Two Blurred Images

Let g_1 and g_2 be two images degraded by motion blur in different directions. The following steps are done in order to restore the original image.

1. The blur directions are calculated as described in the next sub-section, or alternatively, using one of the existing methods (for example [9, 12]).

2. Deblur g_1 using the method of 3.1, the known direction of blur, and using g_2 as the original image. The deblurring is done with a one dimensional kernel at the same direction as the direction of motion blur. Call the deblurred image $g_1^{(1)}$.
3. Deblur g_2 using $g_1^{(1)}$ as the original image, giving $g_2^{(1)}$.
4. Repeat steps 2 and 3, always using the latest version of $g_1^{(i)}$ and $g_2^{(i)}$, until convergence.

The principle ensuring the convergence of the images to the original image is that 1-D blurs in different directions are independent, with the exception of degenerate cases. Two images having motion blur in different directions preserve the information of the original image. A more theoretical approach towards the convergence properties is given in the next section.

3.3 Recovery of Motion Blur Directions

Most existing methods cope with the problem of recovering the motion blur directions either by assuming a constant velocity during the entire imaging process, or assuming certain properties of the image model or of the motion blur operator. For example, assuming that the image is spatially isotropic [12] or that the motion blur kernel is uniform [9]. Our aim is to recover the directions of the motion blur using information from two images, while avoiding the constant velocity assumption.

Each iteration of the algorithm described in the previous sub-sections deblurs the original image. An error in the estimation of the direction of the motion blur reduces the deblurring effect. In the extreme case, using the direction of the motion blur of the second image as an estimator for the motion blur direction of the first one will cause the opposite effect, i.e., will blur the image with the motion blur kernel of the second image. One can use this phenomenon to recover the motion blur direction by enumerating over the angles of the motion blur. For each angle, a single iteration of the method described in the first sub-section can be applied, and the angle which gives the strongest deblurring effect is the angle of the motion blur.

We propose to use this strategy with two exceptions:

- It is preferred to work on the lower resolution levels of the Gaussian pyramids of the images. The accuracy achieved in this way is high enough to obtain the direction of the motion blur, and the computation is faster.
- The sharpness of the recovered image as a function of the estimated direction is approximately monotonic. Thus, partial search can be used.

4 Frequency-Domain Algorithm

4.1 Direct Motion-Blur Reconstruction

In this section we prove the convergence of the deblurring algorithm in the frequency domain. This algorithm is equivalent to the spatial domain algorithm described in the previous section. For simplicity, we deal only with the case of two input images in which the two directions of the motion blur are perpendicular, and the motion between the two images is a pure translation.

In this case, the two input images g_1 and g_2 are observed by two systems modeled as:

$$g_1 = m_1 * f \quad (4)$$

$$g_2 = m_2 * f \quad (5)$$

Where g_1 and g_2 are images degraded by horizontal and vertical motion blur respectively. The displacement between the two images is already expressed by the convolutions, which represent both the motion blur and the image displacement.

Let F be the Discrete Fourier Transform (DFT) of the original image f . Let G_1 and G_2 be the DFT of the input images g_1 and g_2 respectively, and let M_1 and M_2 be the DFT of the motion-blur kernels m_1 and m_2 respectively. Relations 4 and 5 are equivalent to:

$$G_1 = M_1 \cdot F \quad (6)$$

$$G_2 = M_2 \cdot F \quad (7)$$

All the Fourier Transforms described in this section are two-dimensional. However, since each motion blur kernel (m_1 or m_2) is one-dimensional by definition, it has a uniform frequency response along the direction perpendicular to the direction of the kernel. In other words, M_1 is uniform along the y coordinate, and M_2 is uniform along the x coordinate.

The method described in Sect. 3 finds an horizontal blur kernel h that minimizes the l_2 -norm error function $\|g_1 * h - g_2\|_2$. Since minimizing the l_2 -norm error function in the spatial and frequency domains are equivalent, we wish to find a horizontal blur kernel h whose Fourier transform H minimizes the error function

$$\|H \cdot G_1 - G_2\|_2 \quad (8)$$

Since H is uniform along the y coordinate, we will refer to it as a one-dimensional vector, i.e: $H(i) = H(i, j)$ for all $1 \leq j \leq N$, where i and j stand for the x and y coordinates respectively. For each column we minimize the expression

$$\sum_{j=1}^N |H(i) \cdot G_1(i, j) - G_2(i, j)|^2 \quad (9)$$

When $\sum_{j=1}^N |G_1(i, j)|^2 \neq 0$ this minimum is achieved for:

$$H(i) = \frac{\sum_{j=1}^N G_2(i, j) \cdot \bar{G}_1(i, j)}{\sum_{j=1}^N |G_1(i, j)|^2} \quad (10)$$

With \bar{X} denoting the complex conjugate of X . This blur is a weighted average of the i^{th} row in G_2 , which minimizes its l_2 distance to the respective row in G_1 . The reconstruction of the second image using the first one is symmetrical up to changes in the x and y directions.

4.2 Iterative Reconstruction

Similarly to the spatial-domain approach, the algorithm can be enhanced by iteratively updating the first image using the second one and vice versa. Each iteration reduces the motion blur effect upon the images, which in turn enables better results when applying the iteration. The iterative algorithm is derived from Eq. 10, and can be summarized using the following equations:

$$G_1^{(0)} = G_1 \quad (11)$$

$$G_2^{(0)} = G_2 \quad (12)$$

$$G_1^{(n+1)}(i, j) = G_1^{(n)}(i, j) \frac{\sum_k G_2^{(n)}(i, k) \cdot \bar{G}_1^{(n)}(i, k)}{\sum_k |G_1^{(n)}(i, k)|^2} \quad (13)$$

$$G_2^{(n+1)}(i, j) = G_2^{(n)}(i, j) \frac{\sum_k G_1^{(n+1)}(k, j) \cdot \bar{G}_2^{(n)}(k, j)}{\sum_k |G_2^{(n)}(k, j)|^2} \quad (14)$$

4.3 A Convergence Proof Sketch

It can be shown that the transformation relating the DFT of the blur kernels in two consecutive steps is linear. Moreover, it can be described by a stochastic matrix A , with non-negative elements:

$$A(i, j) = \frac{1}{\sum_k |F(k, j)|^2} \cdot \sum_{l=1}^N \frac{|F(l, j)|^2 \cdot |F(l, i)|^2}{\sum_k |F(l, k)|^2} \quad (15)$$

Where $A(i, j)$ is the element in the i^{th} column and the j^{th} row of A .

A is a probability matrix thus describing a contraction mapping. One can conclude, that the all-ones vector is an eigen-vector of A with eigen-value 1, and there is no vector with a bigger eigen-value.

If $\text{deg}(A - I) \geq N - 1$, there is no other eigen-vector with eigen-value 1, and we receive that

$$\lim_{n \rightarrow \infty} M_2^{(n)} = \lim_{n \rightarrow \infty} A^n \cdot M_2^{(0)} = \vec{1}$$

and equivalently,

$$\lim_{n \rightarrow \infty} G_2^{(n)} \stackrel{\text{def}}{=} \lim_{n \rightarrow \infty} M_2^{(n)} \cdot F = F$$

With an exponential convergence. The convergence of $G_1^{(n)}$ to F follows immediately the convergence of $G_2^{(n)}$ to F .

4.4 Failure points

As shown in the previous sub-section, the condition for convergence is that $\text{deg}(A - I) \geq N - 1$, where A is the matrix relating the DFT of the blur kernels in two consecutive steps. A simple case where this condition does not hold is when $A = I$. This happens when the image includes only parallel diagonal lines. In this case, applying motion blur in the x and y directions yield the same degraded images, and thus there is no information for recovery.

5 Examples

We have implemented both the spatial-domain and the frequency-domain methods, and tested them on simulated and real cases of motion blur. The images with real motion blur were restored in the spatial-domain using a 2-D image displacement model describing rotations and translations. The iterations described in Sect. 3 converged after a few steps.

5.1 Restoration from Synthetic Blur

The images in Fig. 2(a)-(c) were obtained by blurring the original image of Fig. 2(d) using a Gaussian-like motion blur. The direction of the motion blur is vertical in Fig. 2(a), horizontal in Fig. 2(b) and diagonal in Fig. 2(c). From comparison of Fig. 2(e) and 2(f) it is clear that the images are better recovered when the directions of motion blur in the two images are orthogonal. The frequency-domain method enables the recovery of images degraded by a wide blur kernel, but limits the motion between the two images to pure translation.

5.2 Restoration from Real Motion Blur

The images shown in Fig. 3 were taken by a camera moving relative to a flat poster. The motion blur in Fig. 3(a) and Fig. 3(b) were obtained by vertical and horizontal motions respectively.

Fig. 3(c) and Fig. 3(d) show a clear enhancement of each of the images. Due to the small rotation between the images, any method assuming only a pure translation would have failed for this sequence.

Fig. 4 shows how using different estimations for the angle of the motion blur direction for images 3(a) and Fig. 3(b) affects the enhancement of the images. These diagrams can be used to recover the directions of the motion blur from two images. For practical reasons the 3rd level of the Gaussian pyramid was used instead of the original images. As can be seen from 4(a) and 4(b), the horizon-

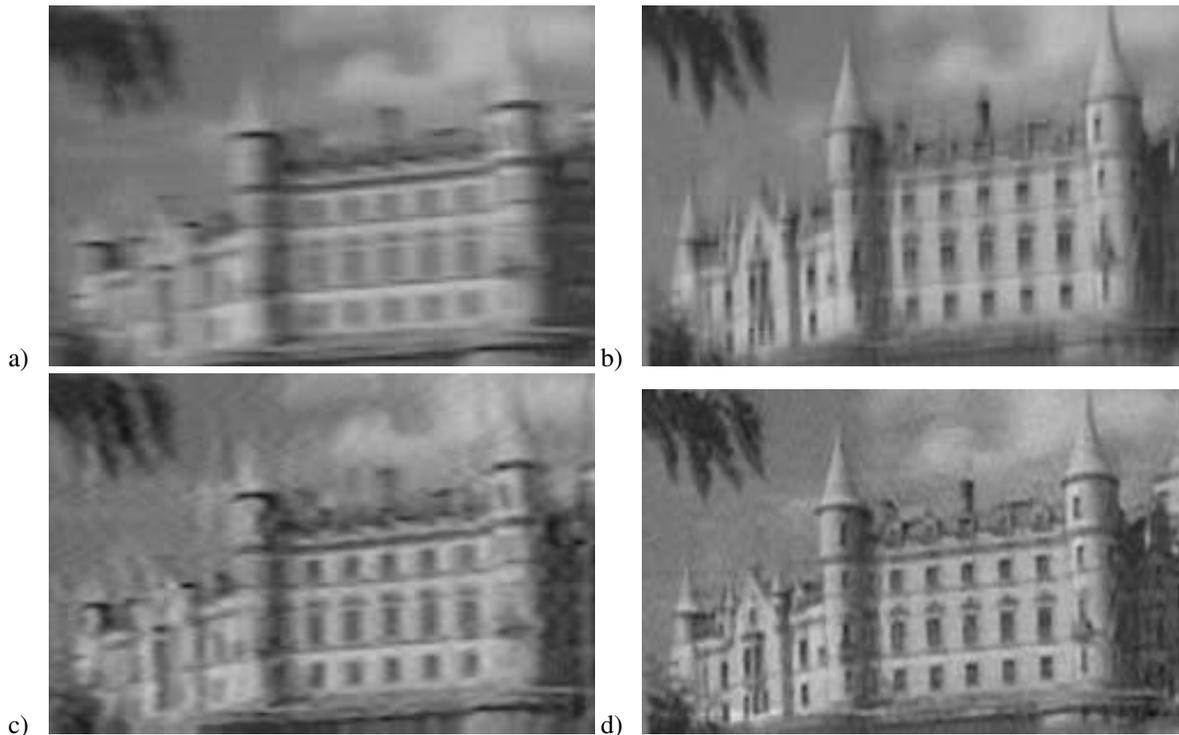


Figure 3. Restoration from two “real” blurred images, related by a translation and a small rotation. (a) and (b) are the input images, degraded by horizontal and vertical motion blur respectively. (c) and (d) are the resulting images after the 4th iteration.

tal and vertical directions yielded the strongest deblurring effect on the first and second image respectively.

Fig. 5 shows a frontal view of a hotel wall. Fig. 5(a) and Fig. 5(b) were taken with fast pan and tilt of the camera. The motion blur induced from the pan or tilt can fit our motion blur model (uniform blur) since the focal length was large.

The restored image after the 4th iteration is shown in Fig. 5(c). The deblurring achieved by applying our method is emphasized by the enlarged Fig. 5(d-e-f). Note also the hotel name, blurred in Fig. 5(a), and the bricks blurred in Fig. 5(b). They are both sharper in the restored image. This demonstrates the use of combined information from both of the images.

6 Concluding Remarks

Two images of the same scene, having motion blur in different directions prove to preserve large amount of information of the original scene. A simple and yet effective method for recovering this information is presented. This method does not require the knowledge of the blur kernel, and does not assume any relation between the image displacement and the motion blur.

Recovering the parameters of image displacement is done simultaneously with the deblurring of the image, which enables an accurate computation of the displacement parameters, and a better restoration of real blurred images.

This method can be used, for example, for images blurred due to hand tremble, where most assumptions about the relations between motion and motion blur fail.

More investigations are needed regarding a possible use of more than two images. For example, it is logical to assume that three images of the same scene, blurred in directions that are in 60° one from another, can be better enhanced together.

- [1] H.C. Andrews and B.R Hunt. Digital image restoration. In *Prentice Hall*, 1977.
- [2] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence. In *ECCV96*, pages II:573–582, 1996.
- [3] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation.

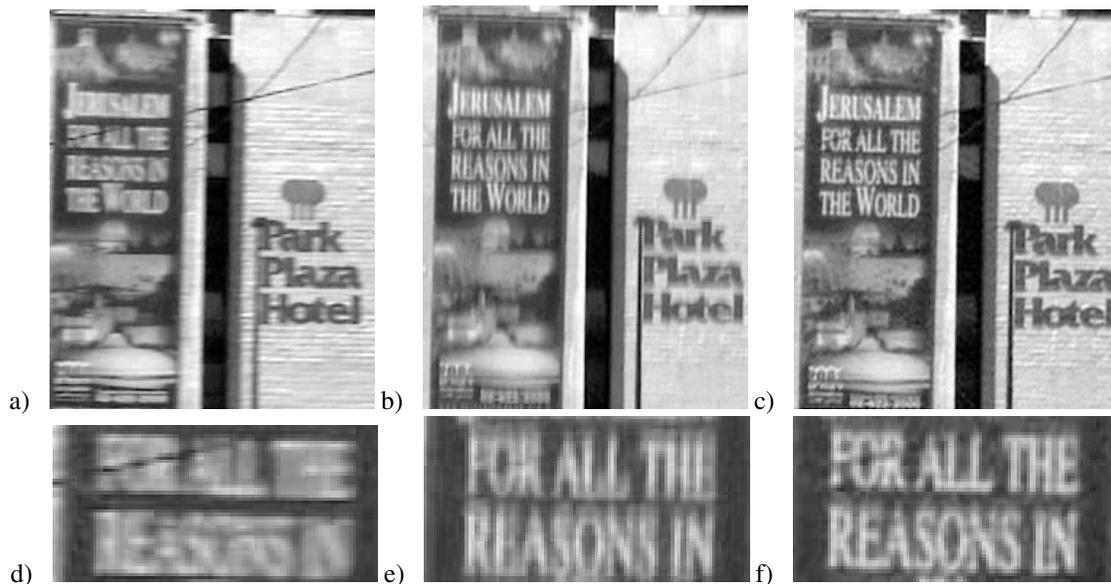


Figure 5. An example of recovering two out-doors blurred images. (a) and (b) were degraded by horizontal and vertical motion blur due to the fast panning and tilting of the hand. (c) is the resulting image after the 4th iteration. (d)-(f) show a zoom view of (a)-(c) respectively.

- In *European Conference on Computer Vision*, pages 237–252, 1992.
- [4] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [5] D. Kundur and D. Hatzinakos. Blind image deconvolution revisited. *SPMag*, 13(6):61–63, November 1996.
- [6] A.J. Patti, M.I. Sezan, and M. Tekalp. Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time. *IEEE Transactions on Image Processing*, 6(8):1064–1078, August 1997.
- [7] S. Reeves and R. Mersereau. Blur identification by the method of generalized cross-validation. *IEEE Transactions on Image Processing*, 1:301–311, July 1992.
- [8] H. Shekarforoush and R. Chellappa. Data-driven multichannel superresolution with application to video sequences. *JOSA-A*, 16(3):481–492, March 1999.
- [9] T. Stockham, T. Cannon, and R. Ingebretsen. Blind deconvolution through digital signal processing. *Proceedings of the IEEE*, 63:678–692, 1975.
- [10] D.L. Tull and A.K. Katsaggelos. Iterative restoration of fast-moving objects in dynamic image sequences. *OptEng*, 35(12):3460–3469, December 1996.
- [11] D.L. Tull and A.K. Katsaggelos. Regularized blur-assisted displacement field estimation. In *ICIP96*, page 18A9, 1996.
- [12] Y. Yitzhaky, I. Mor, A. Lantzman, and N.S. Kopeika. Direct method for restoration of motion blurred images. *JOSA-A*, 15(6):1512–1519, June 1998.



Figure 2. Restoration from simulated motion blur using the frequency-domain approach. The direction of motion blur is vertical in (a), horizontal in (b), and diagonal in (c). (d) The original image. (e) The image recovered from (a) and (b). (f) The image recovered from (b) and (c). Better restoration is obtained when the directions of motion blur are perpendicular.

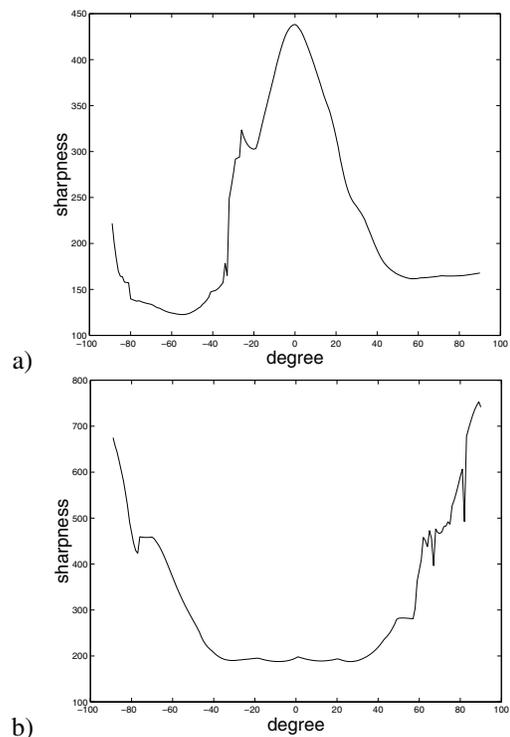


Figure 4. The effect of an error in the estimation of motion blur direction on the sharpness of the resulting image. (a) and (b) are diagrams relating the estimated direction of the motion blur to the achieved sharpness of the image after one iteration of the spatial-domain algorithm. The input images for both diagrams are shown in 3(a) and Fig. 3(b).