1

Digital Image Stabilization

Marius Tico
Nokia Research Center
Palo Alto, CA, USA

1. Introduction

The problem of image stabilization dates since the beginning of photography, and it is basically caused by the fact that any known image sensor needs to have the image projected on it during a period of time called integration time. Any motion of the camera during this time causes a shift of the image projected on the sensor resulting in a degradation of the final image, called motion blur.

The ongoing development and miniaturization of consumer devices that have image acquisition capabilities increases the need for robust and efficient image stabilization solutions. The need is driven by two main factors: (i) the difficulty to avoid unwanted camera motion when using a small hand-held device (like a camera phone), and (ii) the need for longer integration times due to the small pixel area resulted from the miniaturization of the image sensors in conjunction with the increase in image resolution. The smaller the pixel area the less photons/second could be captured by the pixel such that a longer integration time is needed for good results.

It is of importance to emphasize that we make a distinction between the terms "digital image stabilization" and "digital video stabilization". The latter is referring to the process of eliminating the effects of unwanted camera motion from video data, see for instance Erturk & Dennis (2000); Tico & Vehviläinen (2005), whereas digital image stabilization is concerned with correcting the effects of unwanted motions that are taking place during the integration time of a single image or video frame.

The existent image stabilization solutions can be divided in two categories based on whether they are aiming to correct or to prevent the motion blur degradation. In the first category are those image stabilization solutions that are aiming for restoring a single image shot captured during the exposure time. This is actually the classical case of image capturing, when the acquired image may be corrupted by motion blur, caused by the motion that have taken place during the exposure time. If the point spread function (PSF) of the motion blur is known then the original image can be restored, up to some level of accuracy (determined by the lost spatial frequencies), by applying an image restoration approach Gonzalez & Woods (1992); Jansson (1997). However, the main difficulty is that in most practical situations the motion blur PSF is not known. Moreover, since the PSF depends of the arbitrary camera motion during the exposure time, its shape is different in any degraded image as exemplified in Fig. 1. Another difficulty comes from the fact that the blur degradation is not spatially invariant over the image area. Thus, moving objects in the scene may result in very different blur models in certain image areas. On the other hand, even less dynamic scenes may contain different blur models in different regions in accordance to the distance between the objects and the camera,

Fig. 1. Different camera motions cause different blur degradations.

i.e., during a camera translation close objects have larger relative motions than distant objects, phenomenon known as "parallax".

In order to cope with the insufficient knowledge about the blur PSF one could adopt a blind de-convolution approach, e.g., Chan & Wong (1998); You & Kaveh (1996). Most of these methods are computationally expensive and they have reliability problems even when dealing with spatially invariant blur. Until now, published research results have been mainly demonstrated on artificial simulations and rarely on real world images, such that their potential use in consumer products seems rather limited for the moment.

Measurements of the camera motion during the exposure time could help in estimating the motion blur PSF and eventually to restore the original image of the scene. Such an approach have been introduced by Ben-Ezra & Nayar (2004), where the authors proposed the use of an extra camera in order to acquire motion information during the exposure time of the principal camera. A different method, based on specially designed high-speed CMOS sensors has been proposed by Liu & Gamal (2003). The method exploits the possibility to independently control the exposure time of each image pixel in a CMOS sensor. Thus, in order to prevent motion blur the integration is stopped selectively in those pixels where motion is detected.

Another way to estimate the PSF has been proposed in Tico et al. (2006); Tico & Vehviläinen (2007a); Yuan et al. (2007), where a second image of the scene is taken with a short exposure. Although noisy, the secondary image is much less affected by motion blur and it can be used as a reference for estimating the motion blur PSF which degraded the principal image.

In order to cope with the unknown motion blur process, designers have adopted solutions able to prevent such blur for happening in the first place. In this category are included all optical image stabilization (OIS) solutions adopted nowadays by many camera manufactures. These solutions are utilizing inertial senors (gyroscopes) in order to measure the camera motion, following then to cancel the effect of this motion by moving either the image sensor Konika Minolta Inc. (2003), or some optical element Canon Inc. (2006) in the opposite direction. The miniaturization of OIS systems did not reach yet the level required for implementation in a small device like a camera phone. In addition, most current OIS solutions cannot cope well with longer exposure times. In part this is because the inertial motion sensors, used to measure the camera motion, are less sensitive to low frequency motions than to medium and high frequency vibrations. Also, as the exposure time increases the mechanism may drift due to accumulated errors, producing motion blurred images (Fig. 2).

An image acquisition solution that can prevent motion blur consists of dividing long exposure times in shorter intervals, following to capture multiple short exposed image frames of

(a) (b) (c)

Fig. 2. Optical image stabilization examples at different shutter speeds. The images have been captured with a hand-held camera using Canon EF-S 17-85mm image stabilized lens. The exposure times used in taking the pictures have been: (a) 1/25sec, (b) 1/8sec, and (c) 1/4sec. The images get increasingly blurred as the shutter speed slows down.

the same scene. Due to their short exposure, the individual frames are corrupted by sensor noises (e.g., photon-shot noise, readout noise) Nakamura (2006) but, on the other hand, they are less affected by motion blur. Consequently, a long exposed and motion blur free picture can be synthesized by registering and fusing the available short exposed image frames (see Tico (2008a;b); Tico & Vehviläinen (2007b)). Using this technique the effect of camera motion is transformed from a motion blur degradation into a misalignment between several image frames. The advantage is that the correction of the misalignment between multiple frames is more robust and computationally less intensive than the correction of a motion blur degraded image.

In this chapter we present the design of such a multi-frame image stabilization solution, addressing the image registration and fusion operations. A global registration approach, described in Section 2, assists the identification of corresponding pixels between images. However the global registration cannot solve for motion within the scene as well as for parallax. Consequently one can expect local misalignments even after the registration step. These will be solved in the fusion process described in Section 3.

2. Image registration

Image registration is essential for ensuring an accurate information fusion between the available images. The existent approaches to image registration could be classified in two categories: feature based, and image based methods, Zitova & Flusser (2003). The feature based methods rely on determining the correct correspondences between different types of visual features extracted from the images. In some applications, the feature based methods are the most effective ones, as long as the images are always containing specific salient features (e.g., minutiae in fingerprint images Tico & Kuosmanen (2003)). On the other hand when the number of detectable feature points is small, or the features are not reliable due to various image degradations, a more robust alternative is to adopt an image based registration approach, that

utilizes directly the intensity information in the image pixels, without searching for specific visual features.

In general a parametric model for the two-dimensional mapping function that overlaps an "input" image over a "reference" image is assumed. Let us denote such mapping function by $\mathbf{t}(\mathbf{x};\mathbf{p}) = [t_x(\mathbf{x};\mathbf{p})\ t_y(\mathbf{x};\mathbf{p})]^t$, where $\mathbf{x} = [x\ y]^t$ stands for the coordinates of an image pixel, and $\mathbf{p}$ denotes the parameter vector of the transformation. Denoting the "input" and "reference" images by h and g respectively, the objective of an image based registration approach is to estimate the parameter vector $\mathbf{p}$ that minimizes a cost function (e.g., the sum of square differences) between the transformed input image $h(\mathbf{t}(\mathbf{x};\mathbf{p}))$ and the reference image $g(\mathbf{x})$.

The minimization of the cost function, can be achieved in various ways. A trivial approach would be to adopt an exhaustive search among all feasible solutions by calculating the cost function at all possible values of the parameter vector. Although this method ensures the discovery of the global optimum, it is usually avoided due to its tremendous complexity. To improve the efficiency several alternatives to the exhaustive search technique have been developed by reducing the searching space at the risk of losing the global optimum, e.g., logarithmic search, three-step search, etc, (see Wang et al. (2002)). Another category of image based registration approaches, starting with the work of Lucas & Kanade (1981), and known also as gradient-based approaches, assumes that an approximation to image derivatives can be consistently estimated, such that the minimization of the cost function can be achieved by applying a gradient-descent technique (see also Baker & Matthews (2004); Thevenaz & Unser (1998)). An important efficiency improvement, for Lucas-Kanade algorithm, has been proposed in Baker & Matthews (2004), under the name of "Inverse Compositional Algorithm" (ICA). The improvement results from the fact that the Hessian matrix of the cost function, needed in the optimization process, is not calculated in each iteration, but only once in a pre-computation phase.

In this work we propose an additional improvement to gradient-based methods, that consists of simplifying the repetitive image warping and interpolation operations that are required during the iterative minimization of the cost function. Our presentation starts by introducing an image descriptor in Section 2.1, that is less illumination dependent than the intensity component. Next, we present our registration algorithm in Section 2.2, that is based on matching the proposed image descriptors of the two images instead their intensity components.

2.1 Preprocessing

Most of the registration methods proposed in the literature are based on matching the intensity components of the given images. However, there are also situations when the intensity components do not match. The most common such cases are those in which the two images have been captured under different illumination conditions, or with different exposures.

In order to cope with such cases we propose a simple preprocessing step aiming to extract an illumination invariant descriptor from the intensity component of each image. Denoting by $H(\mathbf{x})$ the intensity value in the pixel $\mathbf{x}$, and with $\text{avg}(H)$ the average of all intensity values in the image, we first calculate $\bar{H}(\mathbf{x}) = H(\mathbf{x})/\text{avg}(H)$, in order to gain more independence from the global scene illumination. Next, based on the gradient of $\bar{H}$ we calculate $H_g(\mathbf{x}) = |\bar{H}_x(\mathbf{x})| + |\bar{H}_y(\mathbf{x})|$ in each pixel, and $\text{med}(H_g)$ as the median value of $H_g(\mathbf{x})$ over the entire image.

Finally, the actual descriptor that we are using in the registration operation is given by the following binary image

$$h(\mathbf{x}) = \begin{cases} 1 & \text{if } H_g(\mathbf{x}) > \text{med}(H_g) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

2.2 Registration algorithm

In the following we describe an image based registration method that is using a multi-resolution coarse to fine strategy. Typically in such algorithm, at each iteration step one of the images should be warped in accordance to the parameters estimated so far. In our method this warping operation is highly simplified on the expense of increase memory usage.

The levels of the multi-resolution representation are over-sampled, and they are obtained by iteratively smoothing the original image descriptor h, such that to obtain smoother and smoother versions of it. Let $\tilde{h}_\ell$ denotes the smoothed image resulted after ℓ-th low-pass filtering iterations ($\tilde{h}_0 = h$). The smoothed image at next iteration can be calculated by applying one-dimensional filtering along the image rows and columns as follows:

$$\tilde{h}_{\ell+1}(x,y) = \sum_r w_r \sum_c w_c \tilde{h}_\ell \left(x - 2^\ell c, y - 2^\ell r \right), \quad (2)$$

where w_k are the taps of a low-pass filter.

The registration approach takes advantage of the fact that each decomposition level ($\tilde{h}_\ell$) is over-sampled, and hence it can be reconstructed by a subset of its pixels. This property allows to enhance the efficiency of the registration process by using only a subset of the pixels in the registration algorithm. The advantage offered by the availability of over-sampled decomposition level, is that the set of pixels that can be used in the registration is not unique. A broad range of geometrical transformations can be approximated by simply choosing a different set of pixels to describe the sub-sampled image level. In this way, the over-sampled image level is regarded as a "reservoir of pixels" for different warped sub-sampled versions of the image, which are needed at different stages in the registration algorithm.

Let $\mathbf{x}_{n,k} = [x_{n,k}\ y_{n,k}]^t$, for n,k integers, denote the coordinates of the selected pixels into the smoothed image ($\tilde{h}_\ell$). A low-resolution version of the image ($\hat{h}_\ell$) can be obtained by collecting the values of the selected pixels: $\hat{h}_\ell(n,k) = \tilde{h}_\ell(\mathbf{x}_{n,k})$. Moreover, given an invertible geometrical transformation function $\mathbf{t}(\mathbf{x};\mathbf{p})$, the warping version of the low resolution image can be obtained more efficiently by simply selecting another set of pixels from the area of the smoothed image, rather than warping and interpolating the low-resolution image $\hat{h}_\ell$. This is: $\hat{h}'_\ell(n,k) = \tilde{h}_\ell(\mathbf{x}'_{n,k})$, where $\mathbf{x}'_{n,k} = \text{round}\left(\mathbf{t}^{-1}(\mathbf{x}_{n,k};\mathbf{p})\right)$.

The process described above is illustrated in Fig.3, where the images shown on the bottom row represent two low-resolutions warped versions of the original image (shown in the top-left corner). The two low-resolution images are obtained by sampling different pixels from the smoothed image (top-right corner) without interpolation.

The registration method used in our approach is presented in Algorithm 1. The algorithm follows a coarse to fine strategy, starting from a coarse resolution level and improving the parameter estimate with each finer level, as details in the Algorithm 2. The proposed algorithm relies on matching image descriptors (1) derived from each image rather than image intensity components.

Algorithm 2 presents the registration parameter estimation at one resolution level. In this algorithm, the constant N_0, specifies the number of iterations the algorithm is performing

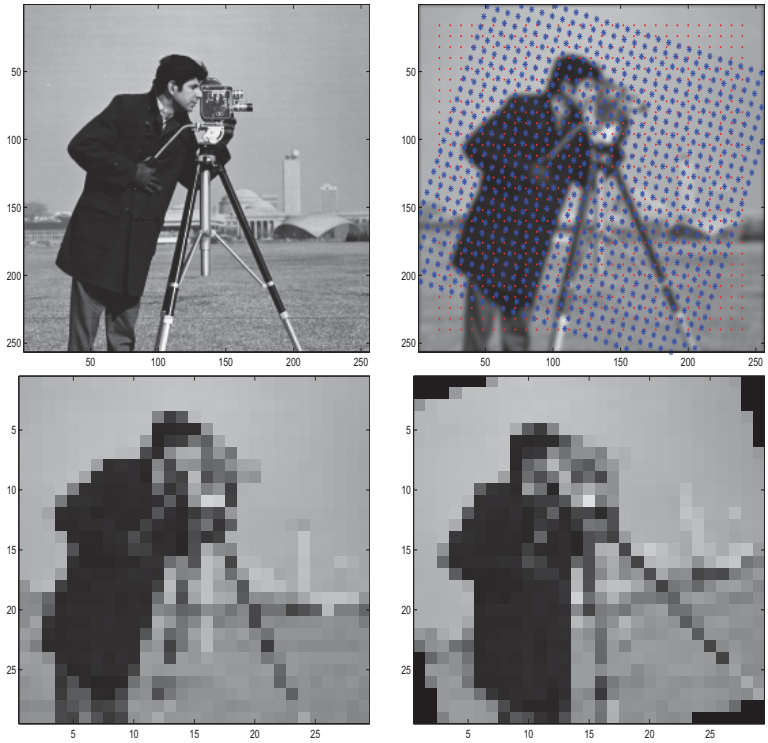


Fig. 3. Low-resolution image warping by re-sampling an over-sampled image decomposition level.

Algorithm 1 Global image registration

Input: the input and reference images plus, if available, an initial guess of the parameter vector $\mathbf{p} = [p_1\ p_2\ \cdots\ p_K]^t$.
Output: the parameter vector that overlaps the input image over the reference image.

1- Calculate the descriptors (1) for input and reference images, denoted here by h and g, respectively.

2- Calculate the decomposition levels of the two image descriptors $\{\tilde{h}_\ell, \tilde{g}_\ell \mid \ell_{min} \leq \ell \leq \ell_{max}\}$.

3- For each level ℓ between ℓ_{max} and ℓ_{min}, do Algorithm 2.

after finding a minima of the error function. This is set in order to reduce the chance of ending in a local minima. As shown in the algorithm the number of iterations is reset to N_0, every time a new minima of the error function is found. The algorithm stops only if no other minima is found in N_0 iterations. In our experiments a value $N_0 = 10$ has been used.

Algorithm 2 Image registration at one level

Input: the ℓ-th decomposition level of the input and reference images ($\tilde{h}_\ell$, $\tilde{g}_\ell$), plus the parameter vector $\mathbf{p} = [p_1\ p_2\ \cdots\ p_K]^t$ estimated at the previous coarser level.
Output: a new estimate of the parameter vector $\mathbf{p}_{out}$ that overlaps $\tilde{h}_\ell$ over $\tilde{g}_\ell$.
Initialization: set minimum matching error $E_{min} = \infty$, number of iterations $N_{iter} = N_0$

1- Set the initial position of the sampling points $\mathbf{x}_{n,k}$ in the vertex of a rectangular lattice of period $D = 2^\ell$, over the area of the reference image $\tilde{g}_\ell$.

2- Construct the reference image at this level: $\hat{g}(n,k) = \tilde{g}_\ell(\mathbf{x}_{n,k})$.

3- For each parameter p_i of the warping function calculate the image

$$J_i(n,k) = \hat{g}_x(n,k)\frac{\partial t_x(\mathbf{x};\mathbf{0})}{\partial p_i} + \hat{g}_y(n,k)\frac{\partial t_y(\mathbf{x};\mathbf{0})}{\partial p_i}$$

where $\hat{g}_x$, $\hat{g}_y$ denote a discrete approximation of the gradient components of the reference image.

4- Calculate the first order approximation of the $K \times K$ Hessian matrix, whose element (i,j) is given by:

$$\mathbf{H}(i,j) = \sum_{n,k} J_i(n,k)J_j(n,k)$$

5- Calculate a $K \times K$ updating matrix $\mathbf{U}$, as explain in the text.

Iterations: while $N_{iter} > 0$

6- Construct the warped low-resolution input image in accordance to the warping parameters estimated so far: $\hat{h}(n,k) = \tilde{h}_\ell(\text{round}\left(\mathbf{t}^{-1}(\mathbf{x}_{n,k};\mathbf{p})\right))$.

7- Determine the overlapping area between $\hat{h}$ and $\hat{g}$, as the set of pixel indices Ψ such that any pixel position $(n,k) \in \Psi$ is located inside the two images.

8- Calculate the error image $e(n,k) = \hat{h}(n,k) - \hat{g}(n,k)$, for any $(n,k) \in \Psi$.

9- Calculate a smooth version $\tilde{e}$ of the error image by applying a 2×2 constant box filter, and determine total error $E = \sum_{(n,k)\in\Psi} |\tilde{e}(n,k)|$.

10- If $E \geq E_{min}$ then $N_{iter} = N_{iter} - 1$, otherwise update $E_{min} = E$, $N_{iter} = N_0$, and $\mathbf{p}_{out} = \mathbf{p}$.

11- Calculate the $K \times 1$ vector $\mathbf{q}$, with $\mathbf{q}(i) = \sum_{(n,k)\in\Psi} \tilde{e}(n,k)J_i(n,k)$.

12- Update the parameter vector $\mathbf{p} = \mathbf{p} + \mathbf{U}\mathbf{q}$

The parameter update (i.e., line 12 in Algorithm 2) makes use of an updating matrix $\mathbf{U}$ calculated in step 5 of the algorithm. This matrix depends of the functional form of the geometrical transformation assumed between the two images, $\mathbf{t}(\mathbf{x};\mathbf{p})$. For instance, in case of affine trans-

formation

$$\mathbf{t}(\mathbf{x};\mathbf{p}) = \begin{bmatrix} (1-p_3)x + p_4 y + p_1 \\ p_5 x + (1-p_6)y + p_2 \end{bmatrix}, \tag{3}$$

the parameter update matrix is

$$\mathbf{U} = \mathrm{diag}\,(D, D, 1, 1, 1, 1)\,\mathbf{H}^{-1}, \tag{4}$$

whereas in case of a projective transformation

$$\mathbf{t}(\mathbf{x};\mathbf{p}) = \begin{bmatrix} (1-p_3)x + p_4 y + p_1 \\ p_5 x + (1-p_6)y + p_2 \end{bmatrix} / (p_7 x + p_8 y + 1), \tag{5}$$

we have

$$\mathbf{U} = \mathrm{diag}\,(D, D, 1, 1, 1, 1, 1/D, 1/D)\,\mathbf{H}^{-1}. \tag{6}$$

In our implementation of multi-resolution image decomposition (2), we used a symmetric filter w of size 3, whose taps are respectively $w_{-1} = 1/4$, $w_0 = 1/2$, and $w_1 = 1/4$. Also, in order to reduce the storage space the first level of image decomposition (i.e., $\tilde{h}_1$), is sub-sampled by 2, such that any higher decomposition level is half the size of the original image.

3. Fusion of multiple images

The pixel brightness delivered by an imaging system is related to the exposure time through a non-linear mapping called "radiometric response function", or "camera response function" (CRF). There are a variety of techniques (e.g., Debevec & Malik (1997); Mitsunaga & Nayar (1999)) that can be used for CRF estimation. In our work we assume that the CRF function of the imaging system is known, and based on that we can write down the following relation for the pixel brightness value:

$$I(\mathbf{x}) = \mathrm{CRF}\,(g(\mathbf{x})\Delta t) \tag{7}$$

where $\mathbf{x} = [x\ y]^T$ denotes the spatial position of an image pixel, $I(\mathbf{x})$ is the brightness value delivered by the system, $g(\mathbf{x})$ denotes the irradiance level caused by the light incident on the pixel $\mathbf{x}$ of the imaging sensor, and Δt stands for the exposure time of the image.

Let I_k, for $k \in \{1, \ldots, K\}$ denote the K observed image frames whose exposure times are denoted by Δt_k. A first step in our algorithm is to convert each image to the linear (irradiance) domain based on knowledge about the CRF function, i.e.,

$$g_k(\mathbf{x}) = (1/\Delta t_k)\,\mathrm{CRF}^{-1}\,(I_k(\mathbf{x}))\,,\ \text{for all } k \in \{1, \ldots, K\}. \tag{8}$$

We assume the following model for the K observed irradiance images:

$$g_k(\mathbf{x}) = f_k(\mathbf{x}) + n_k(\mathbf{x}), \tag{9}$$

where where $\mathbf{x} = [x\ y]^T$ denotes the spatial position of an image pixel, g_k is the k-th observed image frame, n_k denotes a zero mean additive noise, and f_k denotes the latent image of the scene at the moment the k-th input frame was captured. We emphasize the fact that the scene may change between the moments when different input frames are captured. Such changes could be caused by unwanted motion of the camera and/or by the motion of different objects in the scene. Consequently, the algorithm can provide a number of K different estimates of the latent scene image each of them corresponding to a different reference moment.

In order to preserve the consistency of the scene, we select one of the input images as reference, following to aim for improving the selected image based on the visual data available in all captured images. In the following, we denote by g_r, ($r \in \{1, \ldots, K\}$) the reference image observation, and hence the objective of the algorithm is to recover an estimate of the latent scene image at moment r, i.e., $f = f_r$.

The restoration process is carried out based on a spatiotemporal block processing. Assuming a division of g_r in non-overlapping blocks of size $B \times B$ pixels, the restored version of each block is obtained as a weighted average of all blocks located in a specific search range, inside all observed images.

Let $\mathbf{X}_{\mathbf{x}}^B$ denote the sub-set of spatial locations included into a block of $B \times B$ pixels centered in the pixel $\mathbf{x}$, i.e.:

$$\mathbf{X}_{\mathbf{x}}^B = \left\{ \mathbf{y} \in \Omega \mid [-B \ -B]^T < 2\,(\mathbf{y} - \mathbf{x}) \leq [B \ B]^T \right\}, \tag{10}$$

where the inequalities are componentwise, and Ω stands for the image support. Also, let $g(\mathbf{X}_{\mathbf{x}}^B)$ denote the $B^2 \times 1$ column vector comprising the values of all pixels from an image g that are located inside the block $\mathbf{X}_{\mathbf{x}}^B$.

The restored image is calculated block by block as follows

$$\hat{f}(\mathbf{X}_{\mathbf{x}}^B) = \frac{1}{Z} \sum_{k=1}^{K} \sum_{\mathbf{y} \in \mathbf{X}_{\mathbf{x}}^S} w_k\,(\mathbf{x}, \mathbf{y})\, g_k(\mathbf{X}_{\mathbf{y}}^B), \text{ for all } \mathbf{X}_{\mathbf{x}}^B, \tag{11}$$

where $Z = \sum_{k=1}^{K} \sum_{\mathbf{y} \in \mathbf{X}_{\mathbf{x}}^S} w_k\,(\mathbf{x}, \mathbf{y})$, is a normalization value, $\mathbf{X}_{\mathbf{x}}^S$ denotes the spatial search range of size $S \times S$ centered in $\mathbf{x}$, and $w_k\,(\mathbf{x}, \mathbf{y})$ is a scalar weight value assigned to an input block $\mathbf{X}_{\mathbf{y}}^B$ from image g_k.

The weight values are emphasizing the input blocks that are more similar with the reference block. Note that, at the limit, by considering only the most similar such block from each input image we obtain the block corresponding to the optical flow between the reference image and that input image, as in Tico & Vehviläinen (2007b). In such a case the weighted average (11) comprises only a small number of contributing blocks for each reference block. If more computational power is available, we can chose the weight values such that to use more blocks for the restoration of each reference block, like for instance in the solution presented in Tico (2008a), where the restoration of each reference block is carried out by considering all visually similar blocks found either inside the reference image or inside any other input image. Although the use of block processing is more efficient for large images, it might create artifacts in detailed image areas. In order to cope with this aspect, the solution presented in Tico (2008a), proposes a mechanism for adapting the block size to the local image content, by using smaller blocks in detail areas and larger blocks in smooth areas of the image. We conclude this section by summarizing the operations of a multi-frame image stabilization solutions in Algorithm 3.

Algorithm 3 Stabilization algorithm

Input: multiple input images of the scene.
Output: one stabilized image of the scene.

1- Select a reference image either in a manual or an automatic manner. Manual selection can be based on preferred scene content at the moment the image frame was captured, whereas automatic selection could be trivial (i.e., selecting the first frame), or image quality based (i.e., selecting the higher quality frame based on a quality criteria). In our work we select the reference image frame as the one that is the least affected by blur. To do this we employ a sharpness measure, that consists of the average energy of the image in the middle frequency band, calculated in the FFT domain.

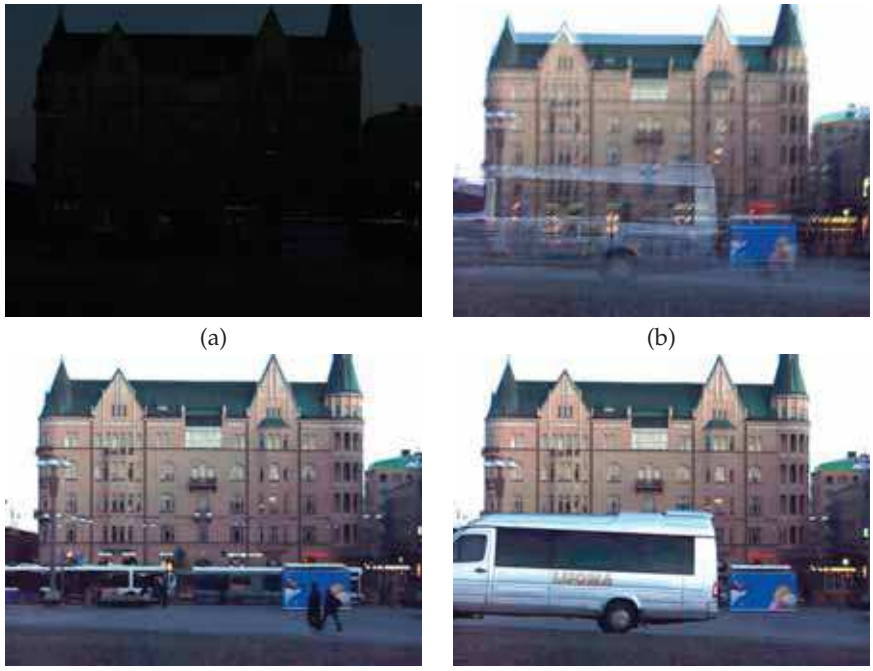
2- Convert the input images to a linear color space by compensating for camera response function non-linearity.

3- Register the input images with respect to the reference image.

4- Estimate the additive noise variance in each input image. Instead using a global variance value for the entire image, in our experiments we employed a linear model for the noise variance with respect to the intensity level in order to emulate the Poisson process of photon counting in each sensor pixel.

5- Restore each block of the reference image in accordance to (11).

6- Convert the resulted irradiance estimate $\hat{f}(\mathbf{x})$, of the final image, back to the image domain, $\hat{I}(\mathbf{x}) = \mathrm{CRF}(\hat{f}(\mathbf{x})\Delta t)$, based on the desired exposure time Δt. Alternatively, in order to avoid saturation and hence to extend the dynamic range of the captured image, one can employ a tone mapping procedure (e.g., Jiang & Guoping (2004)) for converting the levels of the irradiance image estimate into the limited dynamic range of the display system.



(a) (b)

(c) (d)

Fig. 4. Real image example: (a) one input image out of four, (b) overlapping all four input images, (c,d) two different results obtained by selecting different reference moments.

(a) (b)

Fig. 5. Fragment from Fig. 4: (a) the digitally gained reference image, (b) the same fragment from the result.

(a)

(b)

Fig. 6. Real imaging examples: (a) auto-exposed image taken with a camera phone (exposure time: 1.8 sec), (b) stabilized image by fusing four frames with exposure time of 0.3 sec each.

(a)

(b)

Fig. 7. Applying the proposed algorithm onto a single input image (a), delivers a noise filtered version (b), of the input image.

4. Examples

A visual example of the presented method is shown in Fig. 4. In total a number of four short exposed image frames (like the one shown in Fig. 4(a)) have been captured. During the time the individual images have been captured the scene was changed due to moving objects, as reveal by Fig. 4 (b). Applying the proposed algorithm we can recover a high quality image at any moment by choosing the reference frame properly, as exemplified by Fig. 4 (c) and (d). The improvement in image quality achieved by combining multiple images is demonstrated by the fragment in Fig. 5 that shows significant reduction in image noise between one input image Fig. 5(a) and the result Fig. 5(b).

Two examples using images captured with a mobile phone camera are shown in Fig. 6 and Fig. 7. In both cases the algorithm was applied onto the Bayer RAW image data before image

pipeline operations. A simple linear model for the noise variance with respect to the intensity level was assumed in order to emulate the Poisson process of photon counting in each sensor pixel Nakamura (2006), for each color channel.

Fig. 6(a), shows an image obtained without stabilization using the mobile device set on automatic exposure. Due to unwanted camera motion the resulted image is rather blurry. For comparison, Fig. 6(b), shows the image obtained with our proposed stabilization algorithm by fusing several short exposed images of the same scene. An example when the proposed algorithm is applied onto a single input image is shown in Fig. 7. In this case the algorithm acts as a noise filtering method delivering the image Fig. 7(b), by reducing the noise present in the input image Fig. 7(a).

5. Conclusions and future work

In this chapter we presented a software solution to image stabilization based on fusing the visual information between multiple frames of the same scene. The main components of the algorithm, global image registration and image fusion have been presented in detail along with several visual examples. An efficient coarse to fine image based registration solution is obtained by preserving an over-sampled version of each pyramid level in order to simplify the warping operation in each iteration step. Next the image fusion step matches the visual similar image blocks between the available frames coping thereby with the presence of moving objects in the scene or with the inability of the global registration model to describe the camera motion. The advantages of such a software solution against the popular hardware opto-mechanical image stabilization systems include: (i) the ability to prevent blur caused by moving objects in a dynamic scene, (ii) the ability to deal with longer exposure times and stabilized not only high frequency vibrations but also low frequency camera motion during the integration time, and (iii) the reduced cost and size required for implementation in small mobile devices. The main disadvantage is the need to capture multiple images of the scene. However, nowadays most camera devices provide a "burst" mode that ensures fast capturing of multiple images. Future work would have to address several other applications that can take advantage of the camera "burst" mode by fusing multiple images captured with similar of different exposure and focus parameters.

6. References

Baker, S. & Matthews, I. (2004). Lucas-Kanade 20 Years On: A Unifying Framework, *International Journal of Computer Vision* .

Ben-Ezra, M. & Nayar, S. K. (2004). Motion-Based Motion Deblurring, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(6): 689–698.

Canon Inc. (2006). Shift-Method Optical Image Stabilizer.
URL: *www.canon.com/technology/dv/02.html*

Chan, T. F. & Wong, C.-K. (1998). Total Variation Blind Deconvolution, *IEEE Transactions on Image Processing* **7**(3): 370–375.

Debevec, P. E. & Malik, J. (1997). Recovering High Dynamic Range Radiance Maps from Photographs, *Proc. of International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*.

Erturk, S. & Dennis, T. (2000). Image sequence stabilization based on DFT filtering, *IEE Proc. On Vision Image and Signal Processing* **147**(2): 95–102.

Gonzalez, R. C. & Woods, R. E. (1992). *Digital Image Processing*, Addison-Wesley.

Jansson, P. (1997). *Deconvolution of image and spectra*, Academic Press.

Jiang, D. & Guoping, Q. (2004). Fast tone mapping for high dynamic range images, *Proc. of 17th Intl. Conf. on Pattern Recognition (ICPR)*, Vol. 2, pp. 847–850.

Konika Minolta Inc. (2003). Anti-Shake Technology, *www.konicaminolta.com/ products/consumer/digital camera/dimage/dimage-a2/02.html* .

Liu, X. & Gamal, A. E. (2003). Synthesis of high dynamic range motion blur free image from multiple captures, *IEEE Transaction on Circuits and Systems-I* **50**(4): 530–539.

Lucas, B. D. & Kanade, T. (1981). An Iterative Image Registration Technique with an Application to Stereo Vision, *Proc. of 7th Intl Conf on Artificial Intelligence (IJCAI)*, Vancouver,Canada, pp. 674–679.

Mitsunaga, T. & Nayar, S. K. (1999). Radiometric self calibration, *Proc. of Conference on Computer Vision and Pattern Recognition*.

Nakamura, J. (2006). Basics of image sensors, *in* J. Nakamura (ed.), *Image Sensors and Signal Processing for Digital Still Cameras*, CRC Press, pp. 53–94.

Thevenaz, P. & Unser, M. (1998). A Pyramid Approach to Subpixel Registration Based on Intensity, *IEEE Transactions on Image Processing* **7**(1): 27–41.

Tico, M. (2008a). Adaptive block-based approach to image stabilization, *Proc. of the IEEE International Conference of Image Processing (ICIP)*, Vol. 1, San Diego, CA, USA, pp. 521–524.

Tico, M. (2008b). Multiframe image denoising and stabilization, *Proc. of the 15th European Signal Processing Conference (EUSIPCO)*, Lausanne, Switzerland.

Tico, M. & Kuosmanen, P. (2003). Fingerprint matching using an orientation-based minutia descriptor, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **25**(8): 1009–1014.

Tico, M., Trimeche, M. & Vehviläinen, M. (2006). Motion blur identification based on differently exposed images, *Proc. of the IEEE International Conference of Image Processing (ICIP)*, Atlanta, GA, USA, pp. 2021–2024.

Tico, M. & Vehviläinen, M. (2005). Constraint motion filtering for video stabilization, *Proc. of the IEEE International Conference of Image Processing (ICIP)*, Vol. 3, Genova, Italy, pp. 569–572.

Tico, M. & Vehviläinen, M. (2007a). Image stabilization based on fusing the visual information in differently exposed images, *Proc. of the IEEE International Conference of Image Processing (ICIP)*, Vol. 1, San Antonio, TX, USA, pp. 117–120.

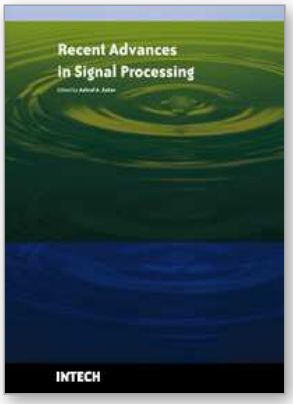
Tico, M. & Vehviläinen, M. (2007b). Robust image fusion for image stabilization, *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, USA.

Wang, Y., Ostermann, J. & Zhang, Y.-Q. (2002). *Video Processing and Communications*, Prentice Hall.

You, Y.-L. & Kaveh, M. (1996). A regularization approach to joint blur identification and image restoration, *IEEE Trans. on Image Processing* **5**(3): 416–428.

Yuan, L., Sun, J., Quan, L. & Shum, H.-Y. (2007). Image deblurring with blurred/noisy image pairs, *ACM Transactions on Graphics* **26**(3).

Zitova, B. & Flusser, J. (2003). Image registration methods: a survey, *Image and Vision Computing* **21**: 977–1000.



Recent Advances in Signal Processing

Edited by Ashraf A Zaher

ISBN 978-953-307-002-5
Hard cover, 544 pages
Publisher InTech
Published online 01, November, 2009
Published in print edition November, 2009

The signal processing task is a very critical issue in the majority of new technological inventions and challenges in a variety of applications in both science and engineering fields. Classical signal processing techniques have largely worked with mathematical models that are linear, local, stationary, and Gaussian. They have always favored closed-form tractability over real-world accuracy. These constraints were imposed by the lack of powerful computing tools. During the last few decades, signal processing theories, developments, and applications have matured rapidly and now include tools from many areas of mathematics, computer science, physics, and engineering. This book is targeted primarily toward both students and researchers who want to be exposed to a wide variety of signal processing techniques and algorithms. It includes 27 chapters that can be categorized into five different areas depending on the application at hand. These five categories are ordered to address image processing, speech processing, communication systems, time-series analysis, and educational packages respectively. The book has the advantage of providing a collection of applications that are completely independent and self-contained; thus, the interested reader can choose any chapter and skip to another without losing continuity.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821