



An efficient spatial domain technique for subpixel image registration

Irene G. Karybali^a, Emmanouil Z. Psarakis^{b,*}, Kostas Berberidis^b, Georgios D. Evangelidis^b

^a Department of Information and Communication Systems Engineering, University of the Aegean, 83200 Karlovassi-Samos, Greece

^b Department of Computer Engineering and Informatics, School of Engineering, University of Patras, 26500 Rio-Patras, Greece

ARTICLE INFO

Article history:

Received 27 July 2007

Received in revised form

11 August 2008

Accepted 15 August 2008

Keywords:

Fast image registration

Subpixel accuracy

Correlation coefficient function

ABSTRACT

In this paper a new technique for performing image registration with subpixel accuracy is presented. The proposed technique, which is based on the maximization of the correlation coefficient function, does not require the reconstruction of the intensity values and provides a closed-form solution to the subpixel translation estimation problem. Moreover, an efficient iterative scheme is proposed, which reduces considerably the overall computational cost of the image registration problem. This scheme properly combined with the proposed similarity measure results in a fast spatial domain technique for subpixel image registration.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

Many image processing applications require an image registration scheme in order to estimate the underlying correspondence between two or more images, which have been acquired either by a single sensor at different times, or by different sensors at the same time but from different viewpoints, or by a combination of the above. Two examples of such applications are remote sensing and biomedical imaging. For remote sensing, registration of infrared to visible spectra is very important for studying satellite images of the earth. Image registration is also very useful for the medical community, since can lead to substantially enhanced diagnosis, in surgical planning, or in the fusion of images coming from different modalities.

Many techniques have been proposed for image registration. Two basic categories are the feature-based and the intensity-based techniques [8,42]. Feature-based techniques first identify edges, contours or other features

common to the compared images and then find the mapping between them [22,15,10]. This results in reduced computational complexity. However, the problem of identifying features is rather complicated and these techniques are very sensitive to the accuracy of the feature extraction stage. On the other hand, intensity-based techniques are more computationally demanding, but avoid the difficulties of feature extraction.

Another classification of the existing image registration techniques can be made based on the domain of implementation (frequency or spatial domain). A well-known frequency domain technique is phase correlation [11]. Many other Fourier-based [9,31], as well as wavelet-based [12], techniques have also been proposed in literature. Note that some methods obtain pixel-level registration that may be adequate for some applications. On the other hand, there are applications that require registration with subpixel accuracy.

The most commonly used methods that provide subpixel accuracy are based on interpolation [36]. The phase-correlation interpolation is such an example. In [37], an algorithm for registering multiple frames simultaneously using nonlinear minimization in frequency domain is described, based on the assumption that the original image is bandlimited. In [16], three new algorithms for 2D translation image registration to within

* Corresponding author. Tel.: +30 2610 996969; fax: +30 2610 996971.

E-mail addresses: karybali@aegean.gr (I.G. Karybali), psarakis@ceid.upatras.gr (E.Z. Psarakis), berberid@ceid.upatras.gr (K. Berberidis), evagelid@ceid.upatras.gr (G.D. Evangelidis).

a fraction of a pixel are presented that use nonlinear optimization and matrix-multiply discrete Fourier transforms. The accuracy of these algorithms is equivalent to that of the conventional fast Fourier transform upsampling approach, while the computational cost is significantly reduced. The authors of [20] propose a frequency domain technique for registration of aliased images. Their technique is based on spectrum cancellation, that is, elimination of the aliased frequency components. Reddy and Chatterji in [31] use the so-called phase correlation of the log-polar transform of the images and apply a high-pass emphasis filter to strengthen high frequencies in the estimation. Another Fourier-based algorithm for image registration with subpixel accuracy is presented in [33], where the pure translation case is investigated. The algorithm detects and removes the frequency components that might cause errors in the shift estimation due to aliasing. In [13], it is shown that the signal power in the phase correlation corresponds to the polyphase transform of a filtered unit impulse centered at the point of registration. Recently, in [38] a frequency domain technique has been proposed for the registration of aliased images, based on their low-frequency, aliasing-free (or marginally affected by aliasing) part. The phase difference between the compared images is computed and for the aliasing-free frequencies the corresponding system of linear equations is formed and the optimal shift parameters result from its least squares solution.

It is easier to describe and handle aliasing in frequency domain, but frequency domain methods are more suitable for global motion models. On the other hand, spatial domain methods generally allow for more general motion models. Some typical criteria for intensity-based registration are the minimization of the squared error between the compared images, the correlation maximization [28] and the maximization of mutual information [40,41,27].

Intensity and correlation interpolation can be used to provide subpixel accuracy [36]. However, the accuracy of these methods depends highly on the interpolation algorithms performance. Other approaches are based on the differential properties of the image sequences [36], or formulate the subpixel registration as an optimization problem [37,20,35]. These approaches rely on the image intensity conservation assumption. In [19], an iterative scheme based on Taylor expansions is presented and a pyramidal scheme is used to increase the precision for large motion parameters. Earlier, Lucas and Kanade in [23] presented an image registration technique that makes use of the spatial intensity gradient of the images to find a good match, using a type of Newton–Raphson iteration. Then, in order to handle large motions, Bouguet implemented a pyramid-based version of the Lucas–Kanade technique [4,5]. The authors of [1] use sparsely sampled regional correlation, providing accuracy better than 0.2 pixels. In [26], an error function linear in the model (local affine) parameters is minimized using least-squares. This error function is then augmented with a nonlinear smoothness constraint, and the least-squares solution is used to bootstrap an iterative nonlinear minimization. This entire procedure is built upon a differential multiscale framework, allowing the capture of both large and small-scale transformations.

The technique proposed here has been motivated by the approach suggested recently in [29], where a correlation-based method for stereo correspondence is presented. The proposed technique aims to maximizing the correlation coefficient, which is a measure that provides robustness to photometric distortions. In contrast to the interpolation-based techniques, the proposed one does not require the reconstruction of the intensity values and provides an easily computed closed-form solution to the subpixel translation estimation problem. Moreover, an efficient iterative scheme is proposed, which reduces considerably the overall computational cost of the image registration problem. This scheme properly combined with the subpixel accuracy technique results in a fast spatial domain technique for subpixel image registration. Note that if exhaustive search is used for the maximization of the correlation coefficient, N^2 searches are required, where N is the number of searches in each dimension. Using the proposed scheme for the computation of the correlation coefficient function, the number of searches is in most cases much smaller than N^2 . We deal here only with translation, since this is the most costly part of an image registration problem, as it is pointed in [7], where the rotation and translation distortions are faced separately. Some preliminary results of this work were presented in [17].

The paper is organized as follows. In Section 2 the problem is formulated and the proposed measure along with the closed-form solution are given. In Section 3 experimental results that evaluate the performance of the proposed subpixel registration technique and compare it to other techniques are provided. The new iterative scheme and some experiments concerning its complexity are presented in Section 4. Finally, the work is concluded in Section 5.

2. Subpixel image registration

2.1. Problem formulation

Let $f(i,j)$ be a reference image and $w(i,j)$ a window in $f(i,j)$, with dimensions $n \times n$ and with its *support* defined by the set

$$\mathcal{S} = [0, n-1] \times [0, n-1]. \quad (1)$$

Let also $g(i,j)$ be a search area in a translated version of image $f(i,j)$, $f_t(i,j)$, with dimensions $m \times m$ (where $m > n$). For both $w(i,j)$ and $g(i,j)$, the upper left corner is located at the origin of a global coordinate system. Then, it is clear that all the possible positions of window $w(i,j)$ in the search area $g(i,j)$ take values in the following set:

$$\mathcal{A} = [0, N-1] \times [0, N-1], \quad N = m - n + 1 \quad (2)$$

and their maximum number is upper bounded by the cardinality of set \mathcal{A} , i.e., N^2 .

Let now $s_{\mathbf{x}}(i,j)$ be a window of the search area $g(i,j)$ that has the same size with w , with $\mathbf{x} = [x, y]^t \in \mathcal{A}$, $(i,j) \in \mathcal{S}$ denoting the coordinates of its upper left corner, and the relative coordinates of the pixels of the window with respect of its upper left corner, respectively. Then, the image registration problem can be stated as a searching

problem. Namely, we are searching for a $\mathbf{x}_0 \in \mathcal{A}$ such that the following n^2 relations hold

$$s_{x_0}(i, j) = w(i, j), \quad \forall (i, j) \in \mathcal{S}. \quad (3)$$

In order to achieve our goal, we need to use an appropriate similarity measure. Such a well known measure is the correlation coefficient of windows $w(i, j)$ and $s_x(i, j)$, which is defined as

$$C_x = \tilde{\mathbf{w}}^t \tilde{\mathbf{s}}_x \quad (4)$$

where $\tilde{\mathbf{w}} = \bar{\mathbf{w}}/\|\mathbf{w}\|$ and $\tilde{\mathbf{s}}_x = \bar{\mathbf{s}}_x/\|\mathbf{s}_x\|$ stand for the zero mean Euclidean normalized versions of vectors $\bar{\mathbf{w}} \equiv \text{vec}(w(i, j) - m_w)$ and $\bar{\mathbf{s}}_x \equiv \text{vec}(s_x(i, j) - m_{s_x})$, with $(i, j) \in \mathcal{S}$, m_w and m_{s_x} denoting the mean values of windows $w(i, j)$ and $s_x(i, j)$, respectively, and operator vec stacking each windows columns in a column vector of length n^2 . The correlation coefficient defined in (4) has the advantage of being invariant to linear photometric distortions. This property is required by many image registration applications where the illumination of the scene is nonuniform.

Having defined the desired similarity measure by (4) and assuming that the translation differences between the compared windows are constant, we have to compute the correlation coefficient for all the possible positions of the window $w(i, j)$ in the search area $g(i, j)$, in order to find the integer translation \mathbf{x}_0 that maximizes (4). In fact, we have to solve the following maximization problem:

$$\max_{\mathbf{x} \in \mathcal{A}} C_x. \quad (5)$$

Solving the above defined optimization problem, we can register images with pixel accuracy. However, in many applications subpixel accuracy is required [38,1,26,29,17]. To this end, in the next paragraph, we are going to extend in the two dimensions case the similarity measure proposed in [29] and formulate an appropriate maximization problem in order to obtain the desired subpixel accuracy.

2.2. Proposed measure for subpixel accuracy

For computing translations with subpixel accuracy, the correlation coefficient in (4) has to be redefined. To this end, let us redefine the correlation coefficient as follows:

$$C_x(\mathbf{t}) = \tilde{\mathbf{w}}^t \tilde{\mathbf{s}}_x(\mathbf{t}) \quad (6)$$

where the elements of vector $\mathbf{t} = [t_1 \ t_2]^t$ are continuous variables, which stand for the subpixel translations along the horizontal and vertical axes of the spatial domain, respectively. Then, the corresponding maximization problem takes the following form:

$$\max_{\mathbf{x} \in \mathcal{A}} \max_{\mathbf{t}} C_x(\mathbf{t}) \quad (7)$$

which involves a maximization with respect to the integer translation \mathbf{x} and a maximization related to the subpixel translation \mathbf{t} .

It is clear that the computational cost we need for the solution of the maximization problem defined in (7) as well as the accuracy of the achieved solution, heavily depends on the specific form of the correlation coefficient function defined in (6) (which typically is a nonlinear

function) as well as the strategy we adopt for its solution. Changing the order of maximizations involved in (7) and by sampling the continuous variables \mathbf{t} , the total registration problem can be solved by adopting a direct search strategy at the expense of increased computational cost and finite precision. On the other hand, if we use an approximation of the correlation coefficient function defined in (6) and select this approximation so that the resulting optimizers are simple to compute, then a large computational saving can be achieved. This is the approach we are using here in order to solve the optimization problem under consideration.

To this end, let us replace the intensity of each pixel of the window $s_x(i + t_1, j + t_2)$, $(i, j) \in \mathcal{S}$ by its first order Taylor approximation, that is

$$s_x(i + t_1, j + t_2) \approx s_x(i, j) + \nabla^t s_x(i, j) \mathbf{t} \quad (8)$$

where $\nabla s_x(i, j)$ denotes the gradient vector of length 2 of the intensity function $s_x(i, j)$ evaluated at the point with relative coordinates (i, j) , with $(i, j) \in \mathcal{S}$. By using this approximation and operator vec we obtain

$$\mathbf{s}_x(\mathbf{t}) \approx \mathbf{s}_x + \Phi_{s_x} \mathbf{t} \quad (9)$$

where each row of the size $n^2 \times 2$ matrix Φ_{s_x} contains the transpose of a gradient vector of length 2 as defined in (8). We now need the zero mean counterpart of vector (9), that is

$$\bar{\mathbf{s}}_x(\mathbf{t}) \approx \bar{\mathbf{s}}_x + \bar{\Phi}_{s_x} \mathbf{t} \quad (10)$$

where $\bar{\mathbf{s}}_x$ denotes the zero mean counterpart of vector \mathbf{s}_x and $\bar{\Phi}_{s_x}$ denotes the column zero mean version of matrix Φ_{s_x} . This matrix results by subtracting from each column of matrix Φ_{s_x} its arithmetic mean. Using (10), and by defining the following quantities

$$\begin{aligned} u_x &= \tilde{\mathbf{w}}^t \bar{\mathbf{s}}_x \\ \mathbf{u}_x &= \bar{\Phi}_{s_x}^t \tilde{\mathbf{w}} \\ v_x &= \|\bar{\mathbf{s}}_x\|^2 \\ \mathbf{v}_x &= \bar{\Phi}_{s_x}^t \bar{\mathbf{s}}_x \\ \Phi_x &= \bar{\Phi}_{s_x}^t \bar{\Phi}_{s_x} \end{aligned} \quad (11)$$

we obtain the following approximation of the correlation coefficient function $C_x(\mathbf{t})$ defined in (6):

$$\hat{C}_x(\mathbf{t}) = \frac{u_x + \mathbf{u}_x^t \mathbf{t}}{\sqrt{v_x + 2\mathbf{v}_x^t \mathbf{t} + \mathbf{t}^t \Phi_x \mathbf{t}}} \quad (12)$$

where, as we can easily see from (11), u_x, v_x are scalar quantities, $\mathbf{u}_x, \mathbf{v}_x$ are vectors of length 2 and Φ_x is a symmetric positive definite matrix of size 2×2 . We must stress at this point that, by incorporating (9) into (6), the correlation coefficient becomes a function of the continuous translation parameters \mathbf{t} . Note also that for the value of $\mathbf{t} = 0$ the resulting value of the correlation coefficient function $\hat{C}_x(0)$ coincides with the correlation coefficient C_x defined in (4). Thus, having defined the correlation coefficient as a function of the continuous translation parameters \mathbf{t} and for a given $\mathbf{x}_0 \in \mathcal{A}$, we have to solve the

following maximization problem:

$$\max_{\mathbf{t}} \hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t}). \quad (13)$$

This is the goal of the next subsection.

2.3. Closed-form solution

The correlation coefficient $\hat{\mathcal{C}}_{\mathbf{x}}(\mathbf{t})$ defined by (12) is a nonlinear function of the continuous translation parameters \mathbf{t} . However, its maximization results in a closed-form solution, which is given in the next theorem.

Theorem 1. Let $\mathbf{x}_0 \in \mathcal{A}$ be given, the correlation coefficient function $\hat{\mathcal{C}}_{\mathbf{x}}(\mathbf{t})$ be defined as in (12), the denominator of (12) be nondegenerate and matrix $\Phi_{\mathbf{x}} - \mathbf{u}_{\mathbf{x}}\mathbf{v}_{\mathbf{x}}^t/u_{\mathbf{x}}$ be of full rank; then, $\hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t})$ attains its unique extremum at

$$\mathbf{t}^0 = \left(\Phi_{\mathbf{x}} - \frac{\mathbf{u}_{\mathbf{x}}\mathbf{v}_{\mathbf{x}}^t}{u_{\mathbf{x}}} \right)^{-1} \left(\frac{v_{\mathbf{x}}}{u_{\mathbf{x}}} \mathbf{u}_{\mathbf{x}} - \mathbf{v}_{\mathbf{x}} \right). \quad (14)$$

Furthermore, this extremum corresponds to a maximum iff the following matrix:

$$H_0 = \hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t}^0) \left(\Phi_{\mathbf{x}} - \frac{\mathbf{u}_{\mathbf{x}}\mathbf{u}_{\mathbf{x}}^t}{\hat{\mathcal{C}}_{\mathbf{x}_0}^2(\mathbf{t}^0)} \right) \quad (15)$$

is positive definite.

Proof. Let us consider the scalar function $\hat{\mathcal{C}}_{\mathbf{x}_0} : \mathbf{R}^2 \rightarrow \mathbf{R}$ defined in (12) which we would like to maximize with respect to the elements t_i , $i = 1, 2$, of vector \mathbf{t} . Let us also consider that $n(\mathbf{t})$, $d^{1/2}(\mathbf{t})$ denote the numerator and denominator of function $\hat{\mathcal{C}}_{\mathbf{x}_0}$, respectively. Then, it is well known that a necessary condition for the function $\hat{\mathcal{C}}_{\mathbf{x}_0}$ to attain an extremum at a point \mathbf{t}^0 in 2D space, is that its partial derivatives with respect to t_i evaluated at this point are equal to zero, that is

$$\left. \frac{\partial \hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t})}{\partial t_i} \right|_{\mathbf{t}=\mathbf{t}^0} = 0, \quad i = 1, 2 \quad (16)$$

or equivalently

$$\left. \frac{\partial n(\mathbf{t})}{\partial t_i} d^{1/2}(\mathbf{t}) - \frac{\partial d^{1/2}(\mathbf{t})}{\partial t_i} n(\mathbf{t}) \right|_{\mathbf{t}=\mathbf{t}^0} = 0, \quad i = 1, 2. \quad (17)$$

From the definitions of $n(\mathbf{t})$ and $d^{1/2}(\mathbf{t})$ we have that

$$\frac{\partial n(\mathbf{t})}{\partial t_i} = u_{x_i}, \quad i = 1, 2 \quad (18)$$

and

$$\frac{\partial d^{1/2}(\mathbf{t})}{\partial t_i} = \frac{v_{x_i} + \phi_{x_i}^t \mathbf{t}}{d^{1/2}(\mathbf{t})}, \quad i = 1, 2 \quad (19)$$

where u_{x_i} , v_{x_i} , ϕ_{x_i} are the i -th element of vectors $\mathbf{u}_{\mathbf{x}}$, $\mathbf{v}_{\mathbf{x}}$ and the i -th column of matrix $\Phi_{\mathbf{x}}$, respectively.

By substituting now (18) and (19) into (17) we obtain

$$u_{x_i} d^{1/2}(\mathbf{t}^0) - \frac{v_{x_i} + \phi_{x_i}^t \mathbf{t}^0}{d^{1/2}(\mathbf{t}^0)} n(\mathbf{t}^0) = 0, \quad i = 1, 2 \quad (20)$$

or equivalently

$$u_{x_i} d(\mathbf{t}^0) - (v_{x_i} + \phi_{x_i}^t \mathbf{t}^0) n(\mathbf{t}^0) = 0, \quad i = 1, 2. \quad (21)$$

Using matrix notation, (21) can be equivalently rewritten as

$$\mathbf{u}_{\mathbf{x}}^t d(\mathbf{t}^0) - (\mathbf{v}_{\mathbf{x}}^t + \mathbf{t}^{0t} \Phi_{\mathbf{x}}) n(\mathbf{t}^0) = \mathbf{0}^t \quad (22)$$

By right-multiplying the above relation by \mathbf{t}^0 we obtain

$$\mathbf{u}_{\mathbf{x}}^t \mathbf{t}^0 d(\mathbf{t}^0) - (\mathbf{v}_{\mathbf{x}}^t \mathbf{t}^0 + \mathbf{t}^{0t} \Phi_{\mathbf{x}} \mathbf{t}^0) n(\mathbf{t}^0) = 0 \quad (23)$$

Note though that, by definition, the following relations also hold true

$$\mathbf{u}_{\mathbf{x}}^t \mathbf{t}^0 = n(\mathbf{t}^0) - u_{\mathbf{x}} \quad (24)$$

$$\mathbf{v}_{\mathbf{x}}^t \mathbf{t}^0 + \mathbf{t}^{0t} \Phi_{\mathbf{x}} \mathbf{t}^0 = d(\mathbf{t}^0) - (v_{\mathbf{x}} + \mathbf{v}_{\mathbf{x}}^t \mathbf{t}^0) \quad (25)$$

and consequently, (23) can be rewritten as follows:

$$\frac{n(\mathbf{t}^0)}{d(\mathbf{t}^0)} = \frac{u_{\mathbf{x}}}{v_{\mathbf{x}} + \mathbf{v}_{\mathbf{x}}^t \mathbf{t}^0}. \quad (26)$$

From (22), (26) and after some simple mathematical manipulations we conclude that, if the matrix $\Phi_{\mathbf{x}} - \mathbf{u}_{\mathbf{x}}\mathbf{v}_{\mathbf{x}}^t/u_{\mathbf{x}}$ is nonsingular, then the function $\hat{\mathcal{C}}_{\mathbf{x}_0}$ attains a unique extremum at the point

$$\mathbf{t}^0 = \left(\Phi_{\mathbf{x}} - \frac{\mathbf{u}_{\mathbf{x}}\mathbf{v}_{\mathbf{x}}^t}{u_{\mathbf{x}}} \right)^{-1} \left(\frac{v_{\mathbf{x}}}{u_{\mathbf{x}}} \mathbf{u}_{\mathbf{x}} - \mathbf{v}_{\mathbf{x}} \right) \quad (27)$$

of 2D parameter space.

A sufficient condition for \mathbf{t}^0 to be the maximizer of $\hat{\mathcal{C}}_{\mathbf{x}_0}$ is that the Hessian matrix of $\hat{\mathcal{C}}_{\mathbf{x}_0}$ evaluated at \mathbf{t}^0 is negative definite [30]. In order to prove that the above-mentioned condition is equivalent to (15) of Theorem 1, we should take the second partial derivatives of $\hat{\mathcal{C}}_{\mathbf{x}_0}$, which, as can be easily shown, are given by

$$\frac{\partial^2 \hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t})}{\partial t_i \partial t_j} = \frac{\partial}{\partial t_j} \left\{ \frac{\frac{\partial n(\mathbf{t})}{\partial t_i} d^{1/2}(\mathbf{t}) - \frac{\partial d^{1/2}(\mathbf{t})}{\partial t_i} n(\mathbf{t})}{d(\mathbf{t})} \right\}, \quad (28)$$

$$i = 1, 2, \quad j = 1, 2.$$

Our goal is now to evaluate the above partial derivatives at the unique extremum \mathbf{t}^0 defined in (27). By substituting into (28) the partial derivatives of functions $n(\mathbf{t})$ and $d^{1/2}(\mathbf{t})$ given by (18), (19) and using (21), after some simple manipulations we obtain

$$\left. \frac{\partial^2 \hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t})}{\partial t_i \partial t_j} \right|_{\mathbf{t}=\mathbf{t}^0} = \frac{-\frac{\phi_{x_{ij}} n(\mathbf{t}^0)}{d^{1/2}(\mathbf{t}^0)} + \frac{u_{x_i} u_{x_j} d^{1/2}(\mathbf{t}^0)}{n(\mathbf{t}^0)}}{d(\mathbf{t}^0)} \quad (29)$$

$$i = 1, 2, \quad j = 1, 2$$

where $\phi_{x_{ij}}$ is the ij element of $\Phi_{\mathbf{x}}$. By using (29) we can easily see that the Hessian matrix of $\hat{\mathcal{C}}_{\mathbf{x}_0}$ can be expressed as follows:

$$H_{\hat{\mathcal{C}}_{\mathbf{x}_0}} = -\frac{\hat{\mathcal{C}}_{\mathbf{x}_0}(\mathbf{t}^0)}{d(\mathbf{t}^0)} \left(\Phi_{\mathbf{x}} - \frac{\mathbf{u}_{\mathbf{x}}\mathbf{u}_{\mathbf{x}}^t}{\hat{\mathcal{C}}_{\mathbf{x}_0}^2(\mathbf{t}^0)} \right). \quad (30)$$

Since now $d(\mathbf{t}^0) > 0$, $H_{\hat{C}_{\mathbf{x}_0}}$ is negative definite iff the following matrix:

$$H_0 = \hat{C}_{\mathbf{x}_0}(\mathbf{t}^0) \left(\Phi_{\mathbf{x}} - \frac{1}{\hat{C}_{\mathbf{x}_0}(\mathbf{t}^0)} \mathbf{u}_{\mathbf{x}} \mathbf{u}_{\mathbf{x}}^t \right) \quad (31)$$

is positive definite, and this concludes the proof of theorem. \square

With Theorem 1 at our disposal, we can adopt a straightforward strategy in solving the total registration problem we are interested in. But before this, let us comment on the computational complexity of the proposed optimizer. The basic cost for the solution of the optimization problem is due to the computation of the quantities defined in (11) as well as to the verification of the kind of extremum attained at the optimal solution by examining the positiveness of matrix H_0 defined in (15). For the former, as we can easily see from (11), the computational complexity is $O(n^2)$. For the latter, based on the uniqueness of the optimum solution, the kind of the unique stationary point can be easily verified. Specifically, this can be achieved by simply comparing the optimum value $\hat{C}_{\mathbf{x}_0}(\mathbf{t}^0)$ of the correlation coefficient function with the value of the function at the pixel location \mathbf{x}_0 , i.e. $\hat{C}_{\mathbf{x}_0}(0)$, thus reducing the required computational cost.

Concluding, the solution of the optimization problem (13) has a computational complexity of $O(n^2)$ and is of the same order with the computational complexity of the correlation coefficient. In order to access the performance of the proposed subpixel image registration technique, we conducted a large number of experiments. The results we obtained are discussed in the next section.

3. Performance evaluation of the subpixel image registration technique

In this section we apply the proposed technique to a series of image pairs in order to evaluate the accuracy of our registration scheme. Moreover, we compare its performance against two leading image registration techniques; a frequency domain technique presented in [38] and a spatial domain one based on a pyramidal implementation [4,5] of the Lucas–Kanade (LK) image registration algorithm [23].

In the image registration technique presented in [38] the estimation of motion parameters is based on the low frequency, aliasing-free part of images. Similar ideas for the translation parameters estimation are used in [20,33,13,32]. Since, as it is reported in [38], the technique outperforms popular frequency as well as spatial domain-based registration techniques [19,25,24], we use it here as a point of reference and compare it against our technique. However, as we will see in the next paragraphs, the size of the critical frequency band vitally degrades the performance of this technique.

As we mentioned above, the proposed technique is also compared to the pyramidal implementation of the LK algorithm. LK image registration algorithm is an iterative scheme which minimizes the l_2 norm of the intensity difference of the two image profiles with respect to the

motion parameters, and its performance measure gives rise to the most popular iterative algorithms for the image registration problem.

Pyramidal implementation of LK algorithm constitutes a powerful scheme for handling large image motions. In this scheme, a pyramidal representation of each image of a pair is built up in a recursive fashion. Starting from the level with the lowest resolution (upper level of the image pair pyramids) the corresponding image registration problem is solved and the optimally estimated motion parameters are used as initial values in the next level of the pyramid. The optimal values of the motion parameters result from the solution of the registration problem at the bottom level of pyramids which is the level with the highest resolution. Finally, the selection of “interest” points appropriate for tracking is done according to the feature selection process described in [4].

In order to evaluate the performance of the proposed technique and compare it against the aforementioned methods, we have conducted two set of simulations. In the first set, image pairs are created by using the Matlab code provided in [2], while in the second one they are created by using the Matlab Camera Tool. In the first set, an image pair is created from a high-resolution image (originally up-sampled by a factor 2) via down-sampling. Moreover, in order to avoid boundary effects the initial up-sampled image is multiplied by a Tukey window, thus avoiding all boundary effects. Using this up-sampled image a shifted copy is created. Finally, the up-sampled image and its shifted version are filtered by using an ideal low-pass filter with normalized cutoff frequency 0.12, and then they are down-sampled by a factor eight. The above-mentioned procedure results in an image pair which is used as input to the registration algorithms under comparison.

In the second set, as we previously mentioned, we use the Matlab Camera Tool in order to create image pairs. To this end, we create a plane in 3D space and we set the angle from which the camera views the current 3D plot, so as to have the same depth for all the plane points. In particular we set the azimuth equal to 0° and the vertical elevation equal to 90° . In this way, by moving the camera and capturing images from different viewpoints we can simulate the planar motion model. Since in this paper we investigate the pure translation case, we make use of the *camdolly* Matlab function.

In all of the experiments, we use the images shown in Fig. 1. Specifically, by using a generator of normally distributed random numbers with a standard deviation of 10, for each one of the above-mentioned images we created 500 translated image pairs and used them as input to the registration algorithms under comparison. For all the competing algorithms, in order to measure the quality of their optimal estimations, the mean absolute error between the actual translations and their estimated counterparts as well as their standard deviations are computed. In addition, in order to evaluate the performance of the competing algorithms in the presence of noise, the case of 32 dB additive white Gaussian noise (which corresponds to ± 10 graylevels) is also considered.



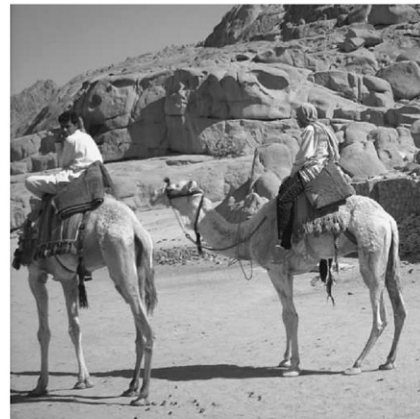
Building



Windmill



Bridge



Camels



Musicians



Well

Fig. 1. Images used in the experiments.

In Table 1, the mean absolute errors and standard deviations achieved by the proposed technique are presented. The Sampled Building, Sampled Windmill, etc., refer to images generated according to [38]. Note that for these images our algorithm performs a bit worse than in the case of images generated by the Matlab Camera Tool. However, its performance remains in a good level. In addition, noise has small effects on the perfor-

mance of the proposed algorithm for the images generated by the Matlab Camera Tool, while it has actually no effect for the sampled images.

As we already mentioned, in [38] the translation is estimated based on a low-frequency part of the phase difference existing between the images. We conducted experiments for different sizes of this low-frequency part (with the radial ranging from 1 to 20) and we computed

Table 1

Performance evaluation of the proposed subpixel registration (axis X, axis Y)

Images	Mean value of absolute error	Standard deviation of absolute error
Building	(0.0136, 0.0140)	(0.0105, 0.0111)
Windmill	(0.0138, 0.0134)	(0.0106, 0.0108)
Bridge	(0.0171, 0.0154)	(0.0117, 0.0117)
Camels	(0.0123, 0.0097)	(0.0096, 0.0072)
Musicians	(0.0165, 0.0158)	(0.0125, 0.0118)
Well	(0.0124, 0.0152)	(0.0094, 0.0118)
Building with Noise	(0.0247, 0.0281)	(0.0231, 0.0259)
Windmill with Noise	(0.0361, 0.0332)	(0.0329, 0.0301)
Bridge with Noise	(0.0570, 0.0446)	(0.0543, 0.0446)
Camels with Noise	(0.0452, 0.0229)	(0.0442, 0.0208)
Musicians with Noise	(0.0298, 0.0212)	(0.0269, 0.0172)
Well with Noise	(0.0222, 0.0226)	(0.0186, 0.0195)
Sampled Building	(0.0368, 0.0336)	(0.0383, 0.0269)
Sampled Windmill	(0.0321, 0.0486)	(0.0275, 0.0422)
Sampled Bridge	(0.0266, 0.0229)	(0.0238, 0.0167)
Sampled Camels	(0.0419, 0.0358)	(0.0333, 0.0272)
Sampled Musicians	(0.0459, 0.0289)	(0.0372, 0.0234)
Sampled Well	(0.0333, 0.0443)	(0.0275, 0.0386)
Sampled Building with Noise	(0.0368, 0.0339)	(0.0395, 0.0276)
Sampled Windmill with Noise	(0.0321, 0.0486)	(0.0285, 0.0438)
Sampled Bridge with Noise	(0.0287, 0.0225)	(0.0268, 0.0188)
Sampled Camels with Noise	(0.0419, 0.0362)	(0.0344, 0.0284)
Sampled Musicians with Noise	(0.0463, 0.0287)	(0.0384, 0.0238)
Sampled Well with Noise	(0.0338, 0.0439)	(0.0283, 0.0388)

Table 2

Performance evaluation of the technique presented in [38] (axis X, axis Y)

Images	Mean value of absolute error	Standard deviation of absolute error
Sampled Building	(0.2489, 0.2069)	(0.8837, 0.8526)
Sampled Windmill	(0.0021, 0.0046)	(0.0116, 0.0313)
Sampled Bridge	(0.2074, 0.2393)	(0.7223, 0.8895)
Sampled Camels	(0.2015, 0.1992)	(0.7996, 0.7121)
Sampled Musicians	(0.2964, 0.1813)	(1.1097, 0.9866)
Sampled Well	(0.6079, 0.4809)	(1.8299, 1.5158)
Sampled Building with Noise	(0.2558, 0.2127)	(0.8816, 0.8515)
Sampled Windmill with Noise	(0.0098, 0.0128)	(0.0120, 0.0312)
Sampled Bridge with Noise	(0.2124, 0.2444)	(0.7196, 0.8887)
Sampled Camels with Noise	(0.2066, 0.2045)	(0.7982, 0.7107)
Sampled Musicians with Noise	(0.3016, 0.1867)	(1.1076, 0.9863)
Sampled Well with Noise	(0.6176, 0.4899)	(1.8273, 1.5149)

scale) for the Sampled Building and Sampled Windmill (for the noise free case) are shown, for different values of radial. In Table 2 the means and standard deviations of the smallest absolute errors we achieved for the sampled images are shown. This techniques behavior remains approximately the same in the presence of noise. Simulation results indicate that the proposed technique is more robust as compared to [38], which seems to be very sensitive to the radials size as well as to the size of the displacements.

The results we obtained from the application of the pyramidal implementation of LK technique, are contained in Table 3. As we can easily see, they are much better than those achieved by the technique proposed in [38], but worse than those achieved by the proposed technique.

Concluding, the proposed technique seems to outperform the other techniques and its computational cost, as we will see in the next section, can be significantly reduced without sacrificing the accuracy.

Let us now proceed with the solution of the total registration problem we are interested in. Based on Theorem 1, for a given position \mathbf{x}_0 of the window $s_{\mathbf{x}}(i, j)$ in the search area, i.e., for $\mathbf{x}_0 \in \mathcal{A}$, we can find the optimum values of the translation parameter \mathbf{t}^0 as well as the achieved maximum value $\hat{C}_{\mathbf{x}}(\mathbf{t}^0)$ of the correlation coefficient function. Hence, the solution of the total registration problem we are interested in, can result from the solutions of N^2 subproblems of the form of (13) by choosing the optimizer that achieves the maximum value of the correlation coefficient function. Since the overall computational cost is $N^2 O(n^2)$, an iterative scheme for fast image registration is proposed in the next section, which with high probability reduces significantly this cost.

4. An iterative scheme for fast image registration

The computationally intensive part of a registration process is the evaluation of the involved measure (the

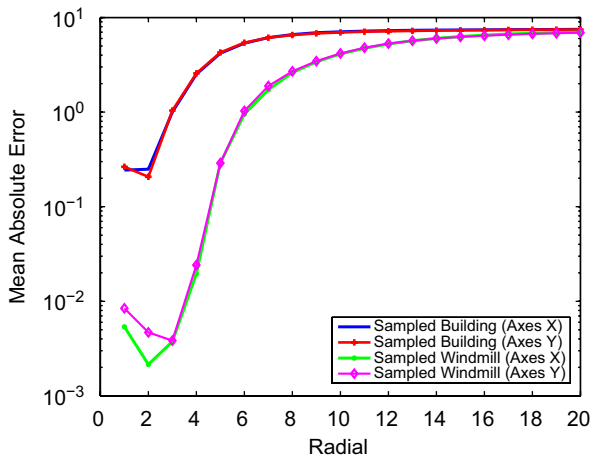


Fig. 2. Mean absolute error for the method presented in [38].

the absolute error for all radial values. In the case of images generated by the Matlab Camera Tool this technique has not even pixel accuracy. The error ranges from 8 to 12 pixels. In the case of sampled images, small errors are achieved for small radials, as we can see in Fig. 2, where the achieved mean absolute errors (in log

Table 3

Performance evaluation of the pyramid-based Lucas–Kanade technique (axis X, axis Y)

Images	Mean value of absolute error	Standard deviation of absolute error
Building	(0.0334, 0.0347)	(0.0304, 0.0391)
Windmill	(0.0271, 0.0299)	(0.0278, 0.0327)
Bridge	(0.0576, 0.0391)	(0.0741, 0.0454)
Camels	(0.0306, 0.0312)	(0.0293, 0.0332)
Musicians	(0.0397, 0.0418)	(0.0465, 0.0525)
Well	(0.0250, 0.0285)	(0.0227, 0.0273)
Building with Noise	(0.0545, 0.0606)	(0.0685, 0.0683)
Windmill with Noise	(0.0629, 0.0713)	(0.0840, 0.0916)
Bridge with Noise	(0.1716, 0.1136)	(0.1876, 0.1450)
Camels with Noise	(0.0968, 0.1001)	(0.1135, 0.1246)
Musicians with Noise	(0.1174, 0.1257)	(0.1459, 0.1524)
Well with Noise	(0.0445, 0.0530)	(0.0439, 0.0524)
Sampled Building	(0.0527, 0.0482)	(0.0654, 0.0555)
Sampled Windmill	(0.0515, 0.0542)	(0.0782, 0.0837)
Sampled Bridge	(0.0749, 0.0573)	(0.1225, 0.0828)
Sampled Camels	(0.0393, 0.0372)	(0.0450, 0.0478)
Sampled Musicians	(0.0534, 0.0519)	(0.0748, 0.0807)
Sampled Well	(0.0318, 0.0313)	(0.0272, 0.0327)
Sampled Building with Noise	(0.0550, 0.0511)	(0.0669, 0.0564)
Sampled Windmill with Noise	(0.0548, 0.0589)	(0.0821, 0.0875)
Sampled Bridge with Noise	(0.0919, 0.0738)	(0.1318, 0.1012)
Sampled Camels with Noise	(0.0472, 0.0481)	(0.0538, 0.0639)
Sampled Musicians with Noise	(0.0654, 0.0623)	(0.0895, 0.0901)
Sampled Well with Noise	(0.0335, 0.0341)	(0.0299, 0.0362)

correlation coefficient function in our case) for all the different relative image positions. It is well-known that cross-correlation can be efficiently implemented in the transform domain. Unfortunately, the correlation coefficient function cannot be computed via a simple and efficient frequency domain procedure, and hence, it has to be computed in the spatial domain [28,14]. In literature, several fast, but approximate spatial domain matching methods have been developed [6]. In [21], a properly normalized cross-correlation measure is obtained via transform domain convolution. The authors of [34] use a Fourier-based scheme for computing the normalized correlation measure, along with nonlinear prefiltering and thresholding, in order to perform a fast cross-spectral registration. Their registration scheme achieves accuracy of ± 2 pixels.

In this section, we propose a new spatial-domain iterative algorithm for fast image registration. When exhaustive search is used, the number of the required searches is N^2 , where N is the number of searches in each dimension. The aim of the proposed algorithm is to reduce this number by iteratively performing searches across a column and along a row. If the correlation coefficient function is concave, this algorithm converges to the global maximum. Otherwise, a proper re-initialization of the algorithm should be performed, as described in the next subsection, in order to reach the global maximum.

4.1. Algorithm description

As we already said, the total computational cost needed for solving the maximization problem defined in (7) depends on the strategy we adopt for its solution, as well as the form of the correlation coefficient function. More specifically, it depends on the number of the local maxima the correlation coefficient function has. To be more precise, if we knew that the correlation coefficient function is concave, then the following simple algorithm converges to the global maximum of the correlation coefficient function, independently of the starting point.

Initialization: Choose randomly a column y_0 of the search area, with $y_0 \in [0, N - 1]$. Set $y_c = y_0$.

- S_1 : For all rows $x_i \in [0, N - 1]$ of the chosen column y_c compute the correlation coefficient function C_{x_i} , with $\mathbf{x}_i = [x_i, y_c]^t$, and find the location $\mathbf{x}_1 = [x_1, y_c]^t$ where the correlation coefficient function attains its maximum value C_{x_1} .
- S_2 : For all columns $y_i \in [0, N - 1]$ of the row x_1 compute the correlation coefficient function C_{x_1} , with $\mathbf{x}_i = [x_1, y_i]^t$, and find the location $\mathbf{x}_2 = [x_1, y_2]^t$ where the correlation coefficient function attains its maximum value C_{x_2} .
- S_3 : If the location \mathbf{x}_2 (resulting from S_2) coincides with \mathbf{x}_1 (resulting from S_1) then stop, else set $y_c = y_2$ and goto S_1 .

However, if the correlation coefficient function has many local maxima, then the probability of the algorithm to be trapped in one of them increases. Hence, we must provide a mechanism which will prevent the algorithm from being trapped in local maxima. To this end, let us modify the last step of the algorithm as follows:

- S_3 : If \mathbf{x}_2 (resulting from S_2) coincides with \mathbf{x}_1 (resulting from S_1) then goto S_4 , else set $y_c = y_2$ and goto S_1 .
- S_4 : If $C_{x_2} > T$ then stop, else choose randomly a row of y_0 (not used before) and goto S_2 .

Notice that the condition added in S_4 ensures that the algorithm will converge at a point where the value of the correlation coefficient function is greater than the value of threshold T (assuming of course that such a point exists). On the other hand, if the condition in S_4 is not satisfied, then the algorithm starts a new cycle to find out a new maximum. This continues until the algorithm finds out a point inside the search area, where the condition in S_4 is satisfied.

It is clear that the convergence of the algorithm to the global maximum depends on the value assigned to threshold T . Assigning, in an optimal way, an appropriate value to threshold T , requires the probability density function (PDF) of the employed similarity measure and the formulation of a hypothesis testing problem [39,18], but this is currently under investigation and is beyond the scope of this paper. In this paper the value of this threshold is estimated in a different way. Note that in many computer vision applications the basic image

registration problem must be solved for a whole image sequence, not just for an image pair. By taking into account this fact, we can use the first two frames of the sequence to evaluate the corresponding correlation coefficient function and use it as a guide for the estimation of threshold T . This practical approach is used in all the experiments we have conducted. Note though that this approach cannot be used in all types of image registration problems and should be used with caution. However, if we restrict ourselves in sequences captured with a high temporal sampling rate and assuming that the statistical properties of the underlying noise do not change during the capturing of the sequence the above-mentioned approach is applicable. Indeed, in such a case we are expecting that the correlation coefficient function (and consequently the desired threshold value) will not dramatically change from pair to pair, retaining almost the same value for a large number of frames in the image sequence.

Let us now concentrate on the computational cost of the proposed algorithm. It is clear that the number of searches needed by the proposed algorithm to reach the global maximum of the correlation coefficient function depends on the correlation coefficient function shape and ranges from $2N - 1$ (best case) to N^2 (worst case, if of course we do not recompute the correlation coefficient function each time we pass from an image position). We will present in next subsection results for two of the images shown in Fig. 1, Building and Windmill. The correlation coefficient functions of these images, evaluated for displacements $(-13, 32)$, for window size 64×64 and search area size 264×264 , have quite different shapes, as shown in Fig. 3. Note that image Building has repeated structures, and as a result its correlation coefficient function has many local maxima as shown in Fig. 3(a). It is also clear that because of the several random steps involved in the proposed algorithm it is expected that different runnings of the algorithm, even on the same set of data, will result into different computational costs. Moreover, the computational cost of the algorithm is

expected to be affected from other factors, such as the size of the window w we would like to register, the size of the search area, as well as the location of the window inside the search area. By realizing that the complexity of the proposed algorithm is a random variable, we are going, in the next subsection, to estimate its empirical cumulative distribution function (ECDF).

4.2. Algorithm complexity

In this subsection, in order to evaluate the complexity of the proposed algorithm, we perform a number of simulations. Before we proceed with the presentation of our results, let us first introduce the figures of merit we are going to adopt, as well as the experimental setup we used. To this end, let

$$r = \frac{c}{N^2} \tag{32}$$

be the ratio of the computational cost (or complexity) c of the proposed algorithm to the cost of the exhaustive search. Then, it is evident that the cumulative distribution function (CDF) of the above defined random variable $F_r(\cdot)$ can be used in accessing the performance of the proposed algorithm in a probabilistic way. Specifically, using $F_r(\cdot)$, we can easily evaluate the probability of each one of the following events:

$$\mathcal{R}_r = \{r \leq r, r \in (0, 1]\}. \tag{33}$$

In addition, using $F_r(\cdot)$, we are able to easily find the CDF of random variables that are based on transformations of the random variable r and we are interested in. Such a random variable is the following one:

$$s = 1 - r \tag{34}$$

which expresses the achieved speedup by the proposed algorithm. Indeed, using the CDF of the random variable r , we can easily prove that the probability of the events

$$\mathcal{S}_s = \{s \geq s, s \in [0, 1)\} \tag{35}$$

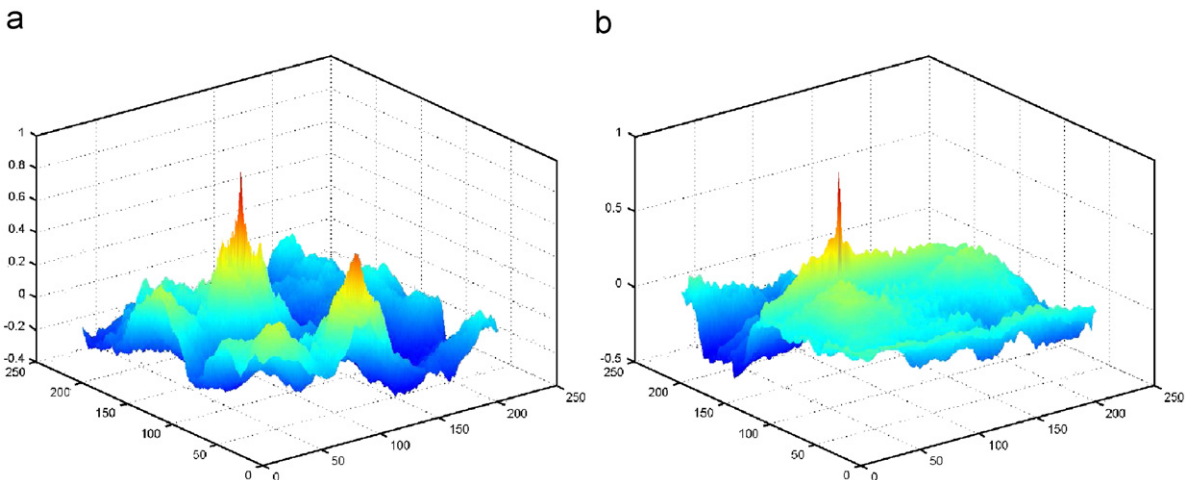


Fig. 3. Correlation coefficient functions corresponding to Building and Windmill, for translation $(-13, 32)$, for window size 64×64 and search area 264×264 .

is given by

$$P(\hat{s} \geq s) = P(1 - r \geq s) = P(r \leq 1 - s). \quad (36)$$

However, since the CDF $F_r(\cdot)$ is an unknown function, we are going to estimate it by evaluating its empirical counterpart. To this end, we adopt the following experimental setup.

Let $f(i, j)$, $w(i, j)$ be an image and a window of size $n \times n$, respectively, which we like to register. For the given image $f(i, j)$ we form the following set of its translated versions:

$$\begin{aligned} \mathcal{S}_p = \{f_p(i, j) = f(i - x_p, j - y_p), x_p, y_p \sim \mathcal{N}(0, 100), \\ p = 1, 2, \dots, P\}. \end{aligned} \quad (37)$$

Then, by keeping the size of the search area constant and using the proposed algorithm, we solve the image registration problem for each member of the set \mathcal{S}_p and keep the resulting computational cost. In this way, we attempt to record the dependency of the computational cost of the proposed algorithm on the location of the window inside the desired search area. In order to record the dependency of the computational cost on the random steps involved in the proposed algorithm, we repeat the above-mentioned procedure K times, thus obtaining a sequence of length $L = P \cdot K$, $r(\ell)$, with $\ell = 1, 2, \dots, L$, which contains the corresponding computational costs. Having obtained the sequence $r(\ell)$, we evaluate the desired ECDF by using the following relation:

$$\hat{F}_L(r) = \frac{1}{L} \sum_{\ell=1}^L u(r - r(\ell)) \quad (38)$$

where $u(\cdot)$ is the step function, and use it as an approximation of the true but unknown CDF. Moreover, in order to record the dependency of the computational cost on the window and search area size, we conducted the aforementioned procedure for different values of their sizes.

Specifically, in our experiments we considered three window cases with dimensions 128×128 , 64×64 and 32×32 and two search area cases. First, search areas of

sizes 228×228 , 164×164 and 132×132 were used, which correspond to $101 \times 101 = 10201$ searches (in an exhaustive search mode). Second, we used search areas of sizes 328×328 , 264×264 and 232×232 corresponding to $201 \times 201 = 40401$ searches. For each case, the length L of the sequence $r(\ell)$ was more than 10000. The value of the threshold, in the noise free case, was estimated to be equal to 0.95, by using the approach described in previous subsection. In order to evaluate the performance of the proposed algorithm under noisy conditions, the case of additive white Gaussian noise (20 dB) was also tested. In this case, an appropriate value for the threshold was found to be equal to 0.75.

The results we obtained from the experiments concerning image Building are shown in Figs. 4 and 5 and Tables 4 and 5, while those concerning image Windmill are shown in Figs. 6 and 7 and Tables 6 and 7, respectively. Specifically, in the figures, the estimated probabilities (using the ECDFs) that the speedups are more than or equal to a value s are plotted, for different window sizes, while in the tables, the probabilities that the proposed algorithm offers speedups more than 50%, 80%, 90% and 95% are contained. The maximum speedup achieved for the case of the smaller search areas (where the maximum number of searches is 10201) is 98.029%, while the maximum speedup for the other case (where the maximum number of searches is 40401) is 99.007%.

The computational savings that the algorithm offers are significant, especially for window sizes 128×128 and 64×64 . The case of window size 32×32 is more difficult, however, there are important savings as well, since as we can see from the corresponding tables the estimated probability that the speedup is more than 50% is very close to 1. In case of noise the results are approximately the same with the noise-free case, for all the window and search area sizes. Note also that the proposed algorithm seems to have similar behavior for both images, although they have totally different structures.

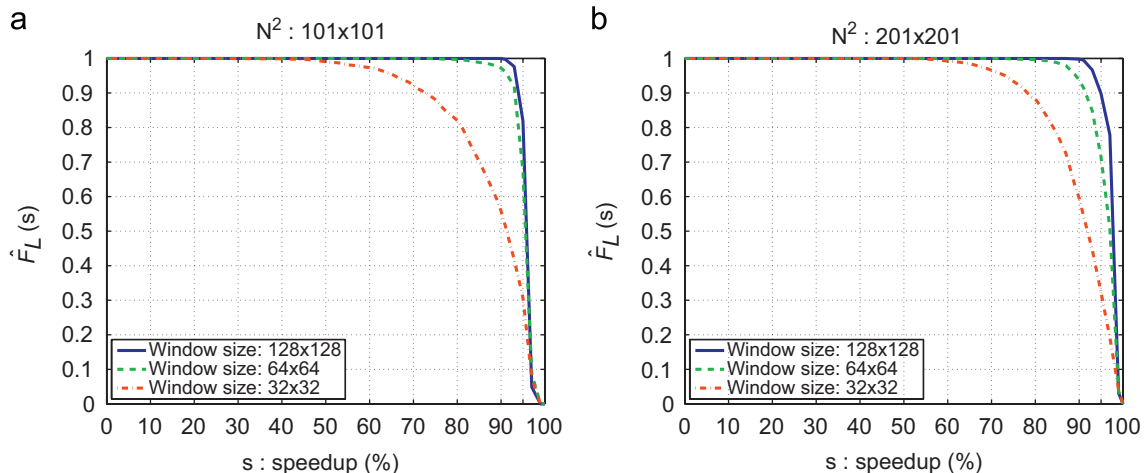


Fig. 4. Image Building, threshold: 0.95. Estimated probability (using the ECDF) that speedup is more than or equal to a value s .

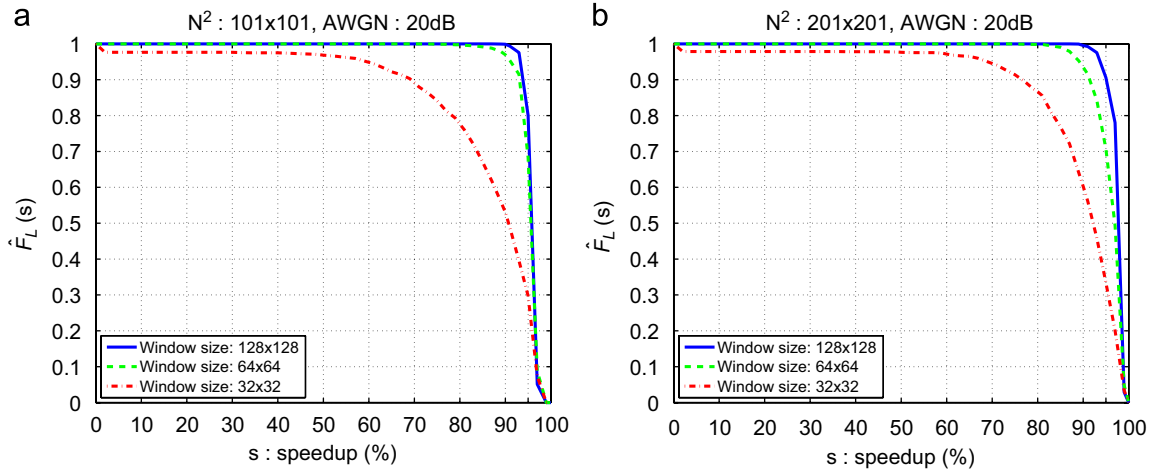


Fig. 5. Image Building, threshold: 0.75. Estimated probability (using the ECDF) that speedup is more than or equal to a value s .

Table 4

Image: Building (512×512), threshold: 0.95

Search area	228×228	164×164	132×132	328×328	264×264	232×232
Window	128×128	64×64	32×32	128×128	64×64	32×32
$\hat{F}_L(0.5)$	1	1	0.9905	1	1	1
$\hat{F}_L(0.8)$	1	0.9970	0.8205	1	0.9970	0.8825
$\hat{F}_L(0.9)$	1	0.9720	0.5585	0.9980	0.9375	0.5960
$\hat{F}_L(0.95)$	0.8190	0.6740	0.3075	0.8980	0.7140	0.3225

Table 5

Image: Building (512×512), AWGN: 20 dB, threshold: 0.75

Search area	228×228	164×164	132×132	328×328	264×264	232×232
Window	128×128	64×64	32×32	128×128	64×64	32×32
$\hat{F}_L(0.5)$	1	1	0.9685	1	1	0.9765
$\hat{F}_L(0.8)$	1	0.9975	0.7760	1	0.9985	0.8665
$\hat{F}_L(0.9)$	0.9990	0.9695	0.5305	0.9955	0.9350	0.6030
$\hat{F}_L(0.95)$	0.8010	0.6835	0.2975	0.9060	0.7070	0.3345

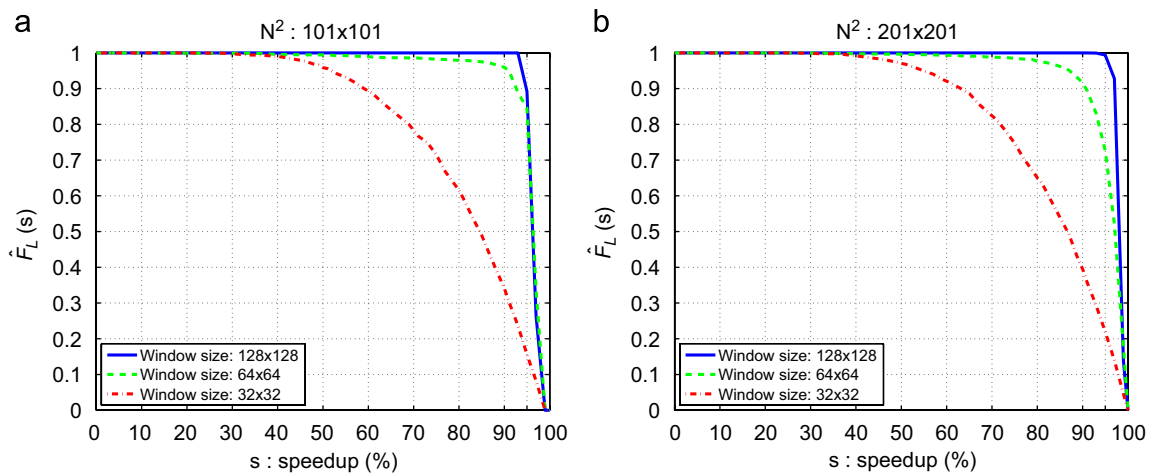


Fig. 6. Image Windmill, threshold: 0.95. Estimated probability (using the ECDF) that speedup is more than or equal to a value s .

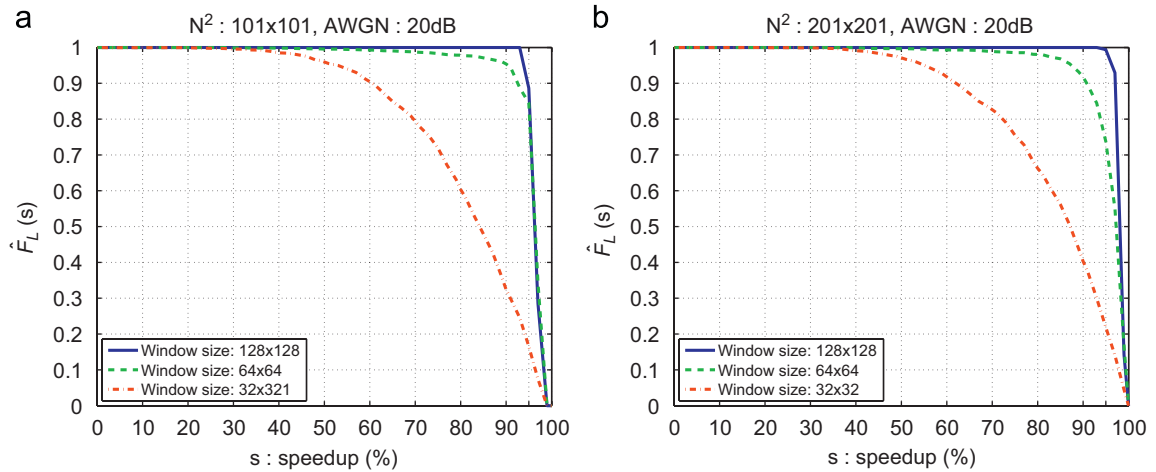


Fig. 7. Image Windmill, threshold: 0.75. Estimated probability (using the ECDF) that speedup is more than or equal to a value s .

Table 6

Image: Windmill (512×512), threshold: 0.95

Search area	228×228	164×164	132×132	328×328	264×264	232×232
Window	128×128	64×64	32×32	128×128	64×64	32×32
$\hat{F}_L(0.5)$	1	0.9945	0.9585	1	0.9960	0.9705
$\hat{F}_L(0.8)$	1	0.9790	0.6160	1	0.9765	0.6505
$\hat{F}_L(0.9)$	1	0.9595	0.3415	1	0.9165	0.3955
$\hat{F}_L(0.95)$	0.8925	0.8450	0.1560	0.9940	0.7260	0.2195

Table 7

Image: Windmill (512×512), AWGN: 20 dB, threshold: 0.75

Search area	228×228	164×164	132×132	328×328	264×264	232×232
Window	128×128	64×64	32×32	128×128	64×64	32×32
$\hat{F}_L(0.5)$	1	0.9955	0.9595	1	0.9965	0.9710
$\hat{F}_L(0.8)$	1	0.9785	0.6035	1	0.9805	0.6625
$\hat{F}_L(0.9)$	1	0.9540	0.3245	1	0.9175	0.4030
$\hat{F}_L(0.95)$	0.8865	0.8450	0.1655	0.9940	0.7375	0.2190

4.3. Evaluation of the proposed technique based on real data

In this last subsection we validate the speedup values obtained in the previous section by the proposed fast registration scheme from its application on real image pairs. To this end, we used the image sequences found in [3]. They have been obtained by moving the camera right and down and capturing (from different viewpoints of a toy scene) a total of 47 different images, contained in sequences HorizL, HorizR, VertL and VertR. The displacements between successive images were approximately 1.5–2.5 pixels (the camera movements were about 0.5 in).

We applied the proposed fast scheme for the registration of all possible combinations of image pairs resulting from the above-mentioned sequences (a total of 1081 image pairs), and we calculated the resulting computational costs. In this experiment we used a window of size

128×128 , search areas of size 328×328 (corresponding to $201 \times 201 = 40401$ searches), and a threshold equal to 0.8. This threshold value was estimated from the evaluation of the correlation coefficient function of images horizL0 and horizL1, shown in Fig. 8(b). The obtained ECDF is depicted (dashed line) in Fig. 9(a). This ECDF is very close to the ECDF obtained by following the procedure described in Section 4.2. Indeed, using the same setup and the procedure described in Section 4.2 we generated 10 000 translated versions of image horizL0 and the resulting ECDF is also depicted (solid line) in Fig. 9(a). The two ECDFs are close enough each other, especially for values of s lower than or equal to 95%, as we can see clearly in Fig. 9(b), where part of Fig. 9(a) is shown. As we can see, the estimated probabilities that the proposed fast scheme will offer a speed up more than 95%, are 0.9940 and 1, respectively. Finally, we must stress at this point

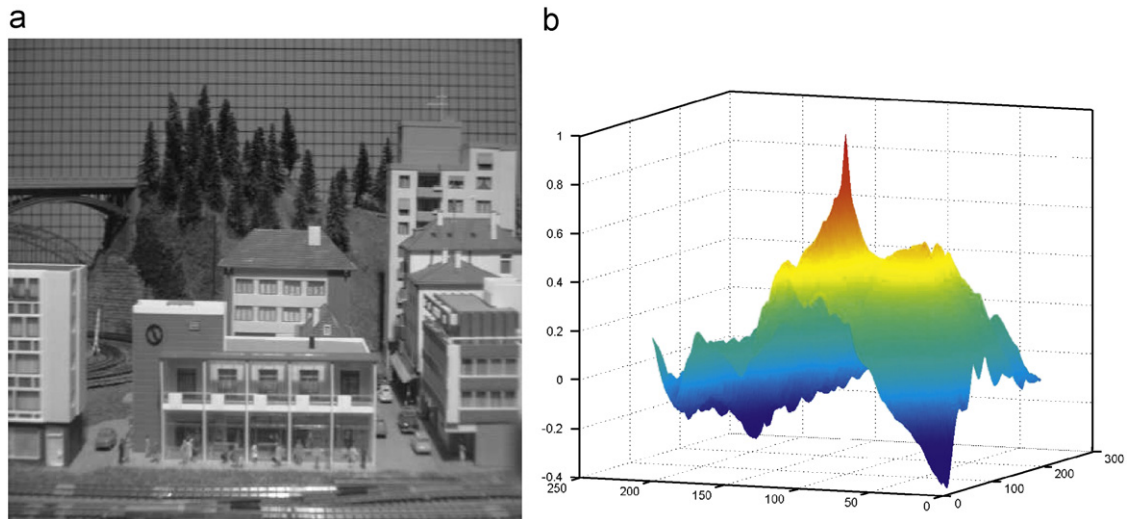


Fig. 8. Image horizL0 and the correlation coefficient function for window size 128×128 and search area 328×328 of images horizL0 and horizL1.

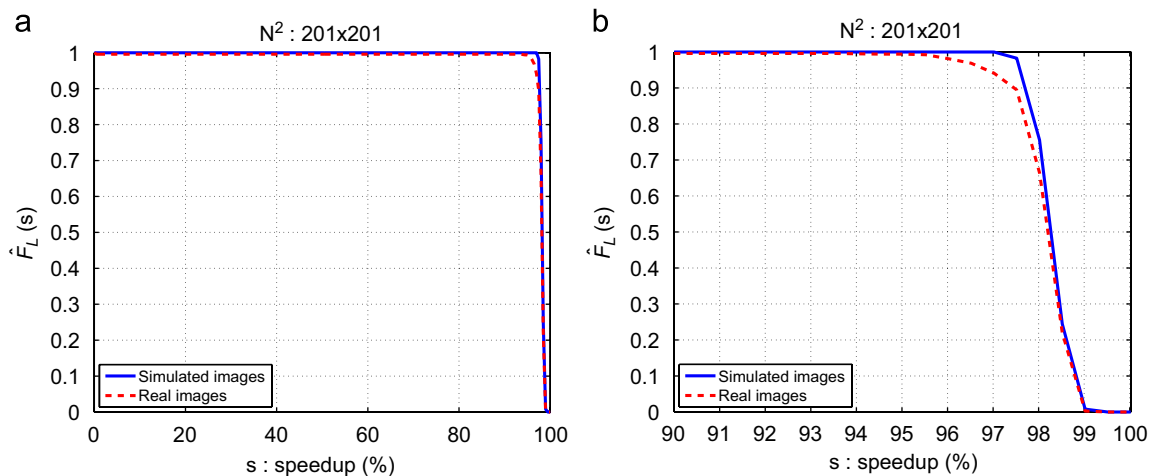


Fig. 9. (a) Estimated probabilities (using the ECDFs) that speedup is more than or equal to a value s , for simulated and real images. (b) Zoom in (a).

that in all the above registration problems the proposed scheme has converged to the same solution provided by the exhaustive search.

5. Conclusion

A new technique for subpixel image registration is proposed in this paper. It is based on the maximization of the correlation coefficient. An easily computed closed-form solution is derived, which does not require the reconstruction of the images intensities, as the intensity interpolation-based methods do. It provides registration of high accuracy and is robust to photometric distortions, as well as to additive white Gaussian noise. Moreover, an efficient spatial domain algorithm is proposed which with high probability reduces significantly the computational cost of the image registration problem. This algorithm properly combined with the proposed similarity measure

results in a fast spatial domain technique for subpixel image registration.

Acknowledgments

This work was supported by the general Secretariat for Research and Technology of the Greek Government, as part of the project “XROMA”, PENED 01. An initial version of this work was supported by the European Social Fund (ESF), Operational Program for Educational and Vocational Training II (EPEAEK II) and particularly the Program HERAKLITOS.

References

- [1] R.J. Althof, M.G.J. Wind, J.T. Dobbins, A rapid and automatic image registration algorithm with subpixel accuracy, *IEEE Trans. Med. Imaging* 16 (3) (June 1997) 308–316.

- [2] [Online]. Available: (http://lcavwww.epfl.ch/reproducible_research/VandewalleSV05/).
- [3] [Online]. Available: (<http://vasc.ri.cmu.edu/idb/images/motion/cil/>).
- [4] [Online]. Available: (http://robots.stanford.edu/cs223b04/algo_tracking.pdf).
- [5] [Online]. Available: (<http://www.intel.com/technology/computing/opencv/>).
- [6] D.I. Barnea, H.F. Silverman, A class of algorithms for fast digital image registration, *IEEE Trans. Comput.* C-21 (February 1972) 179–186.
- [7] K. Berberidis, I. Karybali, A new efficient cross-correlation based image registration technique with improved performance, in: *Proceedings of the 11th European Signal Processing Conference (EUSIPCO 2002)*, vol. I, September 3–6, Toulouse, France, pp. 13–16.
- [8] L. Brown, A survey of image registration techniques, *ACM Comput. Surv.* 24 (4) (1992) 325–376.
- [9] Q. Chen, M. Defrise, F. Deconinck, Symmetric phase-only matched filtering of Fourier–Mellin transform for image registration and recognition, *IEEE Trans. Pattern Recognition Mach. Intell.* 16 (1994) 1156–1168.
- [10] X. Dai, S. Khorram, A feature-based image registration algorithm using improved chain-code representation combined with invariant moments, *IEEE Trans. Geosci. Remote Sensing* 37 (September 1999) 2351–2362.
- [11] E. De Castro, C. Morandi, Registration of translated and rotated images using finite Fourier transforms, *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-9 (May 1987) 700–703.
- [12] J. De Moigne, I. Zavorin, Use of wavelets for image registration, in: *Proceedings of the SPIE Aerospace 2000, Wavelet Applications VIII*, April 2000.
- [13] H. Foroosh (Shekarforoush), J.B. Zerubia, M. Berthod, Extension of phase correlation to subpixel registration, *IEEE Trans. Image Process.* 11 (3) (March 2002) 188–200.
- [14] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 2002.
- [15] V. Govindu, C. Shekhar, Alignment using distributions of local geometric properties, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (October 1999) 1031–1043.
- [16] M. Guizar-Sicairos, S.T. Thurman, J.R. Fienup, Efficient subpixel image registration algorithms, *Opt. Lett.* 33 (2) (January 2008) 156–158.
- [17] I.G. Karybali, E.Z. Psarakis, K. Berberidis, G.D. Evangelidis, Efficient image registration with subpixel accuracy, in: *Proceedings of the 14th European Signal Processing Conference (EUSIPCO 2006)*, September 4–8, Florence, Italy.
- [18] S.M. Kay, *Fundamentals of Statistical Signal Processing: Vol. II, Detection Theory*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1998.
- [19] D. Keren, S. Peleg, R. Brada, Image sequence enhancement using sub-pixel displacement, in: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, June 1988, pp. 742–746.
- [20] S.P. Kim, W.Y. Su, Subpixel accuracy image registration by spectrum cancellation, in: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, April 1993, pp. 153–156.
- [21] J.P. Lewis, Fast normalized cross-correlation, *Vision Interface*, 1995.
- [22] H. Li, B.S. Manjunath, S.K. Mitra, A contour-based approach to multisensor image registration, *IEEE Trans. Image Process.* 4 (March 1995) 320–334.
- [23] B.D. Lucas T. Kanade, An iterative image registration technique with an application to stereo vision, in: *Proceedings of the Imaging Understanding Workshop*, 1981, pp. 255–258.
- [24] L. Lucchese, G.M. Cortelazzo, A noise-robust frequency domain technique for estimating planar roto-translations, *IEEE Trans. Signal Process.* 48 (6) (June 2000) 1769–1786.
- [25] B. Marcel, M. Briot, R. Murrieta, Calcul de translation et rotation par la transformation de Fourier, *Traitement du Signal* 14 (2) (1997) 135–149.
- [26] S. Periaswamy, H. Farid, Elastic registration in the presence of intensity variations, *IEEE Trans. Med. Imaging* 22 (7) (July 2003) 865–874.
- [27] J.P.W. Pluim, J.B.A. Maintz, M.A. Viergever, Mutual-information-based registration of medical images: a survey, *IEEE Trans. Med. Imaging* 22 (8) (August 2003) 986–1004.
- [28] W.K. Pratt, *Digital Image Processing*, second ed., Wiley, New York, 1991.
- [29] E.Z. Psarakis, G.D. Evangelidis, An enhanced correlation-based method for stereo correspondence with subpixel accuracy, in: *Proceedings of the 10th IEEE International Conference on Computer Vision*, October 2005, Beijing, China.
- [30] S.S. Rao, *Engineering Optimization, Theory and Practice*, Wiley, New York, 1996.
- [31] B.S. Reddy, B.N. Chatterji, An FFT-based technique for translation, rotation and scale-invariant image registration, *IEEE Trans. Image Process.* 5 (1996) 1266–1271.
- [32] H. Shekarforoush, M. Berthod, J. Zerubia, Subpixel image registration by estimating the polyphase decomposition of cross power spectrum, *Comput. Vision Pattern Recognition* (June 1994) 532–537.
- [33] H.S. Stone, M.T. Orchard, E.-C. Chang, S.A. Matrucci, A fast direct Fourier-based algorithm for subpixel registration of images, *IEEE Trans. Geosci. Remote Sensing* 39 (10) (October 2001) 2235–2243.
- [34] H.S. Stone, R. Wolpov, Blind cross-spectral image registration using prefiltering and Fourier-based translation detection, *IEEE Trans. Geosci. Remote Sensing* 40 (3) (March 2002) 637–650.
- [35] P. Thévenaz, U.E. Ruttimann, M. Unser, A pyramidal approach to subpixel registration based on intensity, *IEEE Trans. Image Process.* 7 (January 1998) 27–41.
- [36] Q. Tian, M.N. Huhns, Algorithms for subpixel registration, *Comput. Vision Graphics Image Process.* 35 (1986) 220–233.
- [37] R. Tsai, T. Huang, Multiframe image restoration and registration, *Adv. Comput. Vision Image Process.* 1 (1984) 317–339.
- [38] P. Vandewalle, S. Susstrunk, M. Vetterli, A frequency domain approach to registration of aliased images with application to super-resolution, *EURASIP J. Appl. Signal Process.* 2006 (2006) 14, Article ID 71459, doi:10.1155/ASP/2006/71459.
- [39] H.L. Van Trees, *Detection, Estimation, and Modulation Theory*, Wiley, New York, 1971.
- [40] P. Viola, W.M. Wells, Alignment by maximization of mutual information, in: *Proceedings of the 5th International Conference on Computer Vision*, June 1995, pp. 16–23.
- [41] W.M. Wells, P. Viola, et al., Multi-modal volume registration by maximization of mutual information, *Med. Image Anal.* 1 (1996) 35–51.
- [42] B. Zitová, J. Flusser, *Image registration methods: a survey*, *Elsevier Image Vision Comput.* 21 (2003) 977–1000.