

Pseudo-polar based estimation of large translations rotations and scalings in images

*Y. Keller^{1,2}, A. Averbuch¹, M. Isreali³

¹School of Computer Science

Tel Aviv University, Tel Aviv 69978, Israel

²Dept. of Electrical Engineering Systems

Tel-Aviv University Tel-Aviv 69978, Israel

³Faculty of Computer Science, Technion,

Haifa 32000, Israel

Phone: *972-55-694-464

Fax: *972-151-55-694-464

Email: keller@post.tau.ac.il

Submitted to the IEEE Transactions on Image Processing

August 2002

Abstract

One of the major challenges related to image registration is the estimation of large motions without prior knowledge. This paper presents a Fourier based approach that estimates large translation, scale and rotation motions. The algorithm uses the pseudo-polar transform to achieve substantial improved approximations of the polar and log-polar Fourier transforms of an image. Thus, rotation and scale changes are reduced to translations which are estimated using phase correlation. By utilizing the pseudo-polar grid we increase the performance (accuracy, speed, robustness) of the registration algorithms. Scales up to 4 and arbitrary rotation angles can be robustly recovered, compared to a maximum scaling of 2 recovered by the current state-of-the-art algorithms. The algorithm utilizes only 1D FFT calculations whose overall complexity is significantly lower than prior works. Experimental results demonstrate the applicability of these algorithms.

Keywords: Global motion estimation, Sub-pixel registration, Gradient methods, image alignment

EDICS Category={2-ANAL, 2-MOTD}

1 Introduction

Image registration plays a vital role in many image processing applications such as video compression [1, 2], video enhancement [3] and scene representation [4, 5, 6] to name a few. This problem was analyzed using various computational techniques, such as pixel domain Gradient methods [1, 4, 5], correlation techniques [20] and discrete Fourier (DFT) domain algorithms [10, 16]. Gradient methods based image registration algorithms are considered to be the state-of-the-art. They may fail unless the two images are misaligned by only a moderate motion. Fourier based schemes, which are able to estimate relatively large rotation, scaling and translation, are often used as bootstrap for more accurate gradient methods. The basic notion related to Fourier based schemes is the *shift property* [23] of the Fourier transform which allows robust estimation of translations using the *normalized phase-correlation* algorithm [10, 12, 13, 14, 15]. Hence, in order to account for rotations and scaling, the image is transformed into a polar or log-polar Fourier grid (referred to as the Fourier-Mellin transform). Rotations and scaling are reduced to translations in these representations and can be estimated using phase-correlation.

In this paper we propose to iteratively estimate the polar and log-polar DFT using the pseudo-polar FFT (PPFFT) [27, 28]. The resulting algorithm is able to robustly register images rotated by arbitrary angles and scaled up to a factor of 4. It should be noted that the maximum scale factor recovered in [16] and [21] was 2.0 and 1.8, respectively. In particular, the proposed algorithm does not result to interpolation in either spatial or Fourier domain. Only 1D FFT operations are used, making it much faster and especially suited for real-time applications.

The rest of paper is organized as follows: Prior results related to FFT based image registration are given in Section 2, while the pseudo-polar FFT, which is used as a basis for the proposed algorithm is presented in Section 3. The proposed algorithm, is presented in Section 4 and its accuracy is analyzed in Section 5. Experimental results are discussed in Section 6 and final conclusions are given in Section 7.

2 Previous related work

2.1 Translation estimation

The basis of the Fourier based motion estimation is the *shift property* [23] of the Fourier transform. Denote by

$$\mathfrak{F}\{f(x, y)\} \triangleq \hat{f}(\omega_x, \omega_y) \tag{2.1}$$

the Fourier transform of $f(x, y)$. Then,

$$\mathfrak{F}\{f(x + \Delta x, y + \Delta y)\} = \widehat{f}(\omega_x, \omega_y) e^{j(\omega_x \Delta x + \omega_y \Delta y)}. \quad (2.2)$$

Equation (2.2) can be used for the estimation of image translation [10, 15]. Assume the images $I_1(x, y)$ and $I_2(x, y)$ have some overlap that

$$I_1(x + \Delta x, y + \Delta y) = I_2(x, y). \quad (2.3)$$

Equation (2.3) is Fourier transform, so that

$$\widehat{I}_1(\omega_x, \omega_y) e^{j(\omega_x \Delta x + \omega_y \Delta y)} = \widehat{I}_2(\omega_x, \omega_y) \quad (2.4)$$

and

$$\frac{\widehat{I}_2(\omega_x, \omega_y)}{\widehat{I}_1(\omega_x, \omega_y)} = e^{j(\omega_x \Delta x + \omega_y \Delta y)}. \quad (2.5)$$

Thus, the translation parameters $(\Delta x, \Delta y)$ can be estimated in the spatial domain [10, 11] by taking the inverse FFT of Eq. (2.5):

$$Corr(x, y) \triangleq \mathfrak{F}^{-1}\left\{e^{j(\omega_x \Delta x + \omega_y \Delta y)}\right\} = \delta(x - \Delta x, y - \Delta y) \quad (2.6)$$

and by finding the position of the maximum value of the correlation function $Corr(x, y)$:

$$(x, y) = \arg\left\{\max_{(\tilde{x}, \tilde{y})}\{Corr(\tilde{x}, \tilde{y})\}\right\}. \quad (2.7)$$

In order to compensate for possible affine intensity changes Eq. (2.5) is reformulated as:

$$\widetilde{Corr}(\omega_x, \omega_y) \triangleq \frac{\widehat{f}_1(\omega_x, \omega_y) \widehat{f}_2^*(\omega_x, \omega_y)}{\left|\widehat{f}_1(\omega_x, \omega_y)\right| \left|\widehat{f}_2(\omega_x, \omega_y)\right|} = e^{j(\omega_x \Delta x + \omega_y \Delta y)} \quad (2.8)$$

where * denotes the complex conjugate.

This scheme was proven to robustly estimate large translations where the corresponding overlap between the images to be registered is down-to 30% of the smallest image size [15]. As no smoothness assumption is used, non-smooth and noisy functions (such as 2D DFT coefficients) can be accurately registered. Shekarforoush et-al [11] extended the phase-correlation based algorithm to subpixel accuracy by analyzing the shape of the peak value of $\widetilde{Corr}(\omega_x, \omega_y)$ in Eq. (2.8).

A different approach to phase correlation based translation estimation [9, 14, 15, 30] utilizes 2D linear regression in order to fit the phase values calculated in Eq. (2.8) to a two-dimensional linear function

$$\tan^{-1}\left(\widetilde{Corr}(\omega_x, \omega_y)\right) = \omega_x \Delta x + \omega_y \Delta y \quad \forall (\omega_x, \omega_y). \quad (2.9)$$

Solving Eq. (2.9) using linear regression might prove inaccurate [11] due to aliasing and phase wrapping (around 2π) of the spectra at low-frequencies. An iterative solution to phase unwrapping was suggested in [14] while Stone et-al [15] presented two approaches for modelling aliasing effects in order to improve image registration accuracy. The phase domain regression approach can be naturally extended to subpixel accuracy [30], since Eq. (2.9) holds for non-integral values of $(\Delta x, \Delta y)$.

2.2 Polar Fourier representations

The polar Fourier representation (Fourier-Mellin transform) is used to register images that are misaligned due to translation, rotation and scale [16, 21]. Let $I_1(x, y)$ be a translated, rotated and scaled replica of $I_2(x, y)$

$$I_1(x, y) = I_2(s \cdot x \cdot \cos\theta_0 + s \cdot y \cdot \sin\theta_0 + \Delta x, -s \cdot x \cdot \sin\theta_0 + s \cdot y \cdot \cos\theta_0 + \Delta y) \quad (2.10)$$

where θ_0 , s and $(\Delta x, \Delta y)$ are the rotation angle, scale factor and translation parameters, respectively. The DFT of Eq. (2.10) is

$$\widehat{I}_2(r, \theta) = e_1^{j(\omega_x \Delta x + \omega_y \Delta y)} \frac{1}{s^2} \widehat{I}_1\left(\frac{r}{|s|}, \theta + \theta_0\right). \quad (2.11)$$

Hence, M_1 and M_2 , the magnitudes of \widehat{I}_1 and \widehat{I}_2 , respectively, are related by rotation and scaling around the DC component

$$M_2(r, \theta) = \frac{1}{s^2} M_1\left(\frac{r}{|s|}, \theta + \theta_0\right). \quad (2.12)$$

Therefore, rotation and scale changes can be recovered first, regardless the translation parameters. Using a polar or log-polar DFT, rotation and scaling are reduced to translations, which can be robustly recovered using the phase-correlation procedure which will be described in Section 4.1. Using Eq. (2.12) to estimate the rotation angle θ results in an ambiguity by a factor of π [16] in the estimation of the rotation angle. This ambiguity can be resolved by applying both hypotheses θ and $\theta + \pi$ and recover the translational motion $(\Delta x, \Delta y)$ and the correlation peak of each hypothesis. The rotation hypothesis and translation values, which are related to the highest correlation peak, are chosen as the result.

Thus, the algorithm, which its flow is given in Fig. 1 and described below, is used to account for rotation and scaling:

1. The input images are transformed into a polar or log-polar Fourier grid.
2. Rotations and scaling in these representations are reduced to translations and can be estimated using phase-correlation (Eq. (2.8)).

3. The translation is estimated twice - for rotations of θ and $\theta + \pi$. The result with the highest correlation peak is chosen.

The estimation of the Fourier transform on a polar or log-polar grid can be conducted using either

Image domain warping [20] and then 2D FFT calculation [19];

or

Interpolation of the 2D DFT transform in the Fourier domain [16, 21].

The above interpolation based schemes induce significant errors to the approximation of the FFT on polar and log-polar grids, thus limiting the accuracy of the motion estimation [20].

3 The pseudo-polar FFT

The pseudo-polar Fourier transform evaluates the Fourier transform on a non-Cartesian point-set, which we call the pseudo-polar grid, using $O(N \log N)$ operations where $N = n^2$ and n is the number of pixels in each side of the square image. The rapid exact evaluation of the Fourier transform at these non-Cartesian grid points is possible using the fractional Fourier transform [29].

3.1 2D pseudo-polar grid

We define Θ to be the set of angles induced by lines with equally spaced slopes. More precisely, we define two sets of angles:

$$\begin{aligned} \Theta_2 &= \{ \arctan(2l/n) \mid l \in \mathbb{Z} \quad -n/2 \leq l \leq n/2 \} \\ \Theta_1 &= \{ \pi/2 - \arctan(2l/n) \mid l \in \mathbb{Z} \quad -n/2 \leq l \leq n/2 \}. \end{aligned} \quad (3.1)$$

If we take an arbitrary element from Θ_2 , i.e., $\theta_2^l = \arctan(2l/n)$ then the slope of the line corresponding to θ_2^l is $s = \tan \theta_2^l = 2l/n$. By inspecting two elements θ_2^l and θ_2^{l+1} we can see that the difference between the slopes, which correspond to the two angles, is

$$s_2 - s_1 = \tan \theta_2^{l+1} - \tan \theta_2^l = \frac{2(l+1)}{n} - \frac{2l}{n} = \frac{2}{n} \quad (3.2)$$

which means that our angles define a set of equally spaced slopes (see Figs. 2.a and 2.b).

We define the set of angles Θ to be

$$\Theta \triangleq \Theta_1 \cup \Theta_2. \quad (3.3)$$

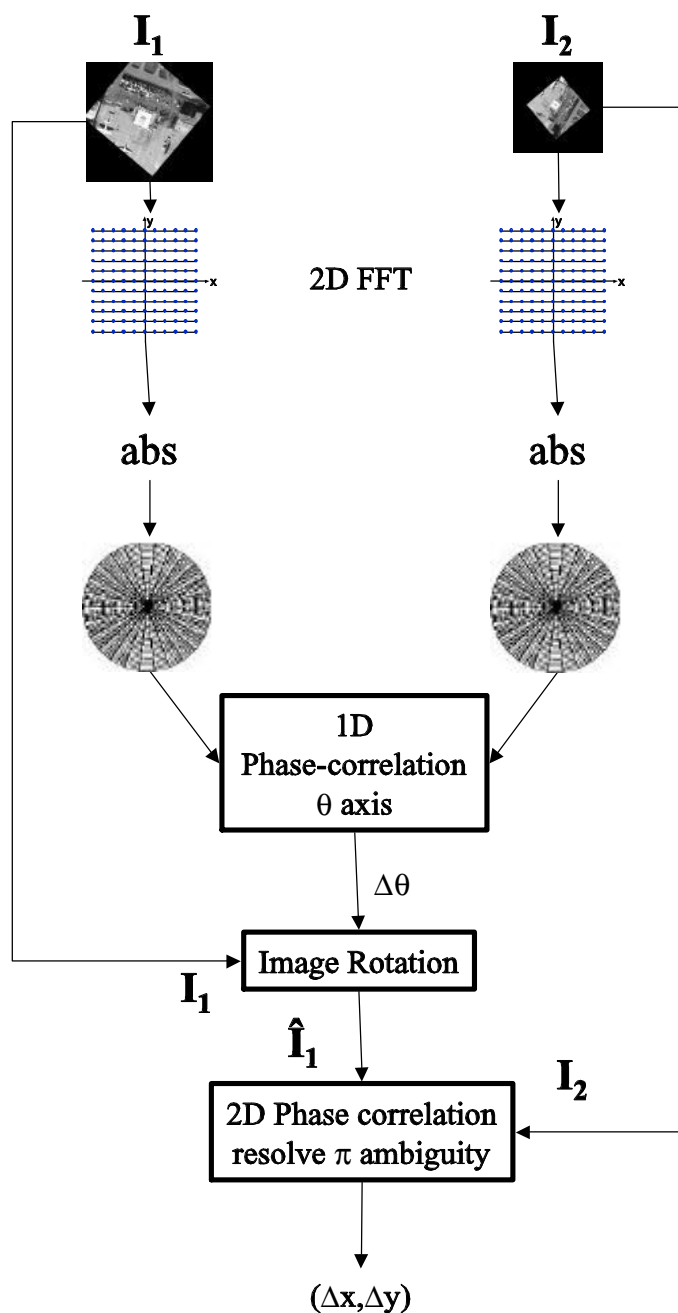


Figure 1: The flow of the state-of-the-art FFT based image registration: 1. The magnitudes of the polar DFT are approximated by interpolating the magnitudes of the 2D FFT. 2. The rotation $\Delta\theta$ is recovered using 1D phase-correlation on the θ axis. 3. One of the input images is rotated by angle $\Delta\theta$. 4. The translation is recovered and the θ ambiguity is resolved by applying a 2D phase-correlation twice: using I_1 rotated by both θ and $\theta + \pi$.

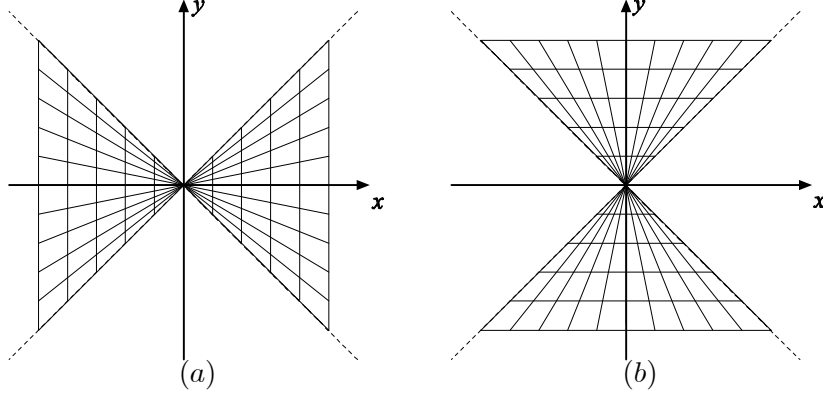


Figure 2: Slopes defined by Eq. (3.1): (a) Slopes that correspond to angles in Θ_2 , (b) Slopes that correspond to angles in Θ_1 .

Define the sets P_1 and P_2

$$P_1 \triangleq \left\{ \left(-\frac{2l}{n}k, k \right) \mid -n/2 \leq l \leq n/2 \quad -n \leq k \leq n \right\} \quad (3.4)$$

$$P_2 \triangleq \left\{ \left(k, -\frac{2l}{n}k \right) \mid -n/2 \leq l \leq n/2 \quad -n \leq k \leq n \right\} \quad (3.5)$$

and the set P as

$$P \triangleq P_1 \cup P_2. \quad (3.6)$$

The set P is called the pseudo-polar grid (see Figs. 3.a and 3.b). As can be seen from the figures, k serves as a “pseudo-radius” and l serves as a “pseudo-angle”.

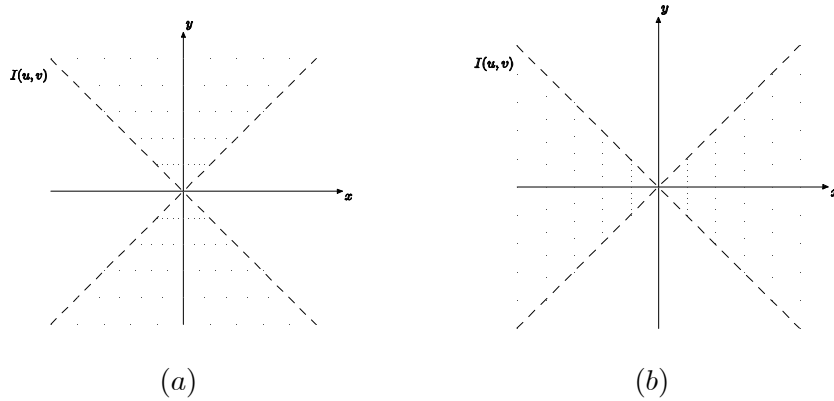


Figure 3: The sectors P_1 and P_2 : (a) The pseudo-polar sector P_1 , (b) The pseudo-polar sector P_2 .

3.2 2D pseudo-polar calculation

We define the pseudo-polar Fourier transform by resampling \hat{I} at the pseudo-polar grid P .

Definition 1 (Pseudo-polar Fourier transform) *The pseudo-polar Fourier transform PP_j ($j = 1, 2$) is a linear transformation, which is defined for $k, l \in \mathbb{Z}$, $-n \leq k \leq n$, $-n/2 \leq l \leq n/2$, as*

$$PPI_1(k, l) = \widehat{I}\left(-\frac{2l}{n}k, k\right) \quad (3.7)$$

$$PPI_2(k, l) = \widehat{I}\left(k, -\frac{2l}{n}k\right) \quad (3.8)$$

and

$$\widehat{I}(\xi_1, \xi_2) = \sum_{u=-n/2}^{n/2-1} \sum_{v=-n/2}^{n/2-1} I(u, v) e^{-\frac{2\pi i}{m}(\xi_1 u + \xi_2 v)}. \quad (3.9)$$

According to definition 1, for a given image I of size $n \times n$, the pseudo-polar Fourier transform PPI , is defined in definition 1, as

$$PPI(s, k, l) \triangleq \begin{cases} PPI_1(k, l) & s = 1 \\ PPI_2(k, l) & s = 2 \end{cases} = \begin{cases} \widehat{I}(-2lk/n, k) & s = 1 \\ \widehat{I}(k, -2lk/n) & s = 2 \end{cases} \quad (3.10)$$

where \widehat{I} is the trigonometric polynomial

$$\widehat{I}(\xi_1, \xi_2) = \sum_{u=-n/2}^{n/2-1} \sum_{v=-n/2}^{n/2-1} I(u, v) e^{-\frac{2\pi i}{m}(\xi_1 u + \xi_2 v)}. \quad (3.11)$$

We will show that for a given image I , we can rapidly resample \widehat{I} on the pseudo-polar grid points. The operator behind this resampling process is the *fractional Fourier transform*

Definition 2 (Fractional Fourier transform) *Given a vector $X = (X(j), -n/2 \leq j \leq n/2)$, and an arbitrary $\alpha \in \mathbb{R}$, the fractional Fourier transform is defined as*

$$(F_{n+1}^\alpha X)(\omega) = \sum_{u=-n/2}^{n/2} X(u) e^{-2\pi i \alpha \omega u / (n+1)} \quad \omega \in \mathbb{Z}, \quad -n/2 \leq \omega \leq n/2. \quad (3.12)$$

An important property of the fractional Fourier transform defined in 2 is that for a given vector $\{X(j)\}$ of $n + 1$ numbers, the sequence $(F_{n+1}^\alpha X)(\omega)$ can be computed using $O(n \log n)$ operations for any $\alpha \in \mathbb{R}$ (see [29]).

Definition 2 is usually referred to as the unaliased fractional Fourier transform. It differs from the usual definition of the fractional Fourier transform given by [29]. According to the definition given in [29], for a vector $X = (X(j), 0 \leq j \leq n)$, and an arbitrary $\alpha \in \mathbb{R}$, the fractional Fourier transform is defined as

$$(F_n^\alpha X)(k) = \sum_{j=0}^{n-1} X(j) e^{-2\pi i j k \alpha} \quad 0 \leq k < n. \quad (3.13)$$

3.2.1 Algorithm description for the rapid computation of the 2D pseudo-polar Fourier transform (pseudo-code)

Notation

F_n^{-1} 1D inverse DFT.

F_m^α Fractional Fourier transform with factor α . The operator accepts a sequence of length n , pads it symmetrically to length $m = 2n + 1$, applies to it the fractional Fourier transform with factor α , and returns the $n + 1$ central elements.

F_2 2D DFT.

Using the above notation we define the operator $G_{k,n}$ to be

$$G_{k,n} = F_m^\alpha \circ F_n^{-1} \quad \alpha = 2k/n. \quad (3.14)$$

Input: Image I of size $n \times n$.

Output: Two arrays Res_1, Res_2 of size $m \times (n + 1)$ ($m = 2n + 1$, m values along the x-axis, and $n + 1$ values along the y-axis; or in a matrix notation - $n + 1$ rows and m columns).

Process:

1. Pad I symmetrically along the y-axis to length $2n + 1$. Denote the result \tilde{I}_1 .
2. $\hat{I}_d = F_2(\tilde{I}_1)$.
3. For each k ($-n \leq k \leq n$) we take the k^{th} row from \hat{I}_d and compute

$$W_k = G_{k,n}(\hat{I}_d(\cdot, k)), \quad W_k \in \mathbb{C}^{n+1}.$$

4. $Res_1(k, l) = W_k(-l)$
5. Pad I symmetrically along the x-axis to length $2n + 1$. Denote the result \tilde{I}_2 .
6. $\hat{I}_d = F_2(\tilde{I}_2)$.
7. For each k ($-n \leq k \leq n$), we take the k^{th} column from \hat{I}_d and compute

$$W_k = G_{k,n}(\hat{I}_d(k, \cdot)), \quad W_k \in \mathbb{C}^{n+1}.$$

8. $Res_2(k, l) = W_k(-l)$

In words, algorithm 3.2.1 computes Res_1 as:

1. Pad both ends of the y-direction of the image I and compute the 2D DFT of the padded image. The results are placed in \hat{I}_d .
2. Apply the 1D inverse Fourier transform on each row k in \hat{I}_d . Resample each row k in the resulted array using 1D fractional Fourier transform with $\alpha = 2k/n$.
3. Flip each row around its center.

4 The proposed registration algorithm

This section presents the proposed image registration algorithms, which use the pseudo-polar FFT (PPFFT) to reduce rotations and scalings to translations in the polar and log-polar domains, respectively. These translations are recovered using the phase-correlation scheme presented in Section 2.1. Thus, the phase-correlation algorithm is applied on three domains: spatial, polar DFT's magnitude and log-polar DFT's magnitude. Table 1 summarizes the properties of the translation estimation in each domain. Detailed implementation issues are given in section 4.1.

| Estimated motion | Registration domain | Domain properties | Translation estimation | Described in section |
|------------------|-------------------------|----------------------------|-----------------------------------|----------------------|
| translation | spatial | non-cyclic | \vec{r} and $\vec{\theta}$ axes | 4.1.1 |
| rotation | polar DFT magnitude | cyclic $\vec{\theta}$ axis | $\vec{\theta}$ axis | 4.1.2 |
| rotation+scaling | log-polar DFT magnitude | cyclic $\vec{\theta}$ axis | \vec{r} and $\vec{\theta}$ axes | 4.1.3 |

Table 1: Properties of the translation estimation algorithms in various domains: spatial, polar DFT's magnitude and log-polar DFT's magnitude.

Section 4.2 introduces the rotation and translation estimation algorithm illustrated in Fig. 5, while the simultaneous estimation of scaling, rotation and translation is presented in section 4.3 and Fig. 6.

4.1 Translation estimation implementation

Translation is estimated using Eq. (2.8) where several implementation issues have to be considered according to the type of the domain on which the motion is conducted. The evaluation of Eq. (2.8) requires the input signals to be zero padded in order to have the same size. Furthermore, implementing Eq. (2.8) using the DFT estimates the *cyclic correlation* [24] which differs from the regular correlation needed for translation estimation. This wrap-around problem is illustrated in

Fig. 4 and it is solved by an appropriate zero padding. Section 4.1.1 presents the padding of the spatial domain translation where the cyclic wrap-around effect is avoided, while in sections 4.1.2 and 4.1.3 the $\vec{\theta}$ axis is cyclic over the interval $[0, 2\pi]$.

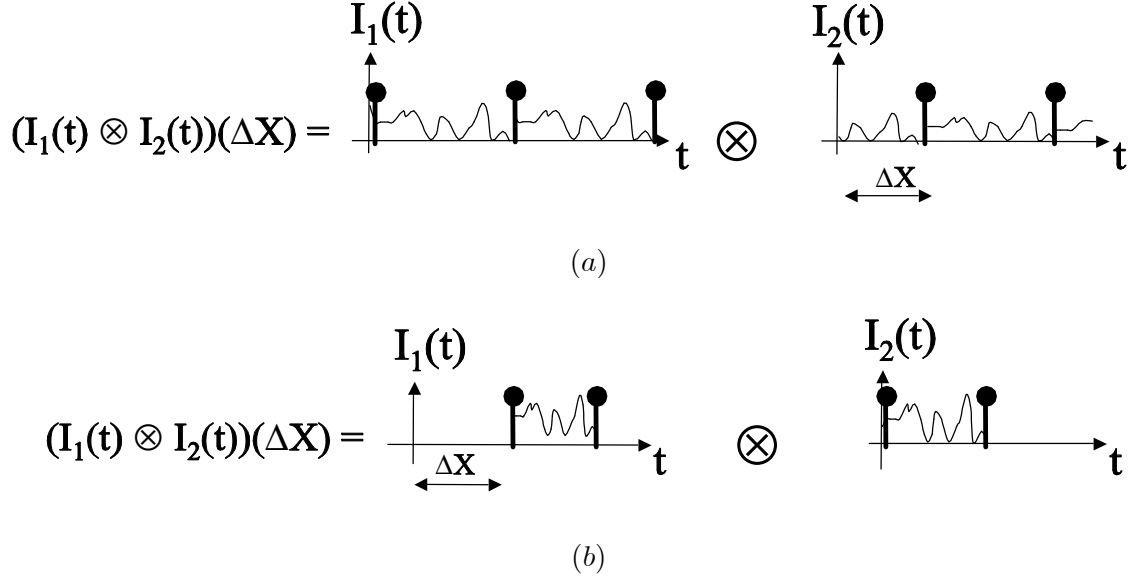


Figure 4: Illustration why fast correlation estimation which uses DFT requires zero padding. Figures (a) and (b) demonstrate the calculation of 1D correlation for a translation of Δx . (a) Without zero padding: non-corresponding parts of the input signals are aligned in the computation, thus, degrading its accuracy. (b) With zero padding: only corresponding parts are aligned within the correlation computation.

4.1.1 Spatial domain translation estimation

Denote by $length(I)$ the length of the 1D signal I . Then, in order to avoid the cyclic wrap-around problem in the DFT computation, both signals are zero-padded such that:

$$\begin{aligned} length(\tilde{I}_1) &= \max(length(I_1), length(I_2)) + \Delta X_{\max} \\ length(\tilde{I}_2) &= length(\tilde{I}_1) \end{aligned} \quad (4.1)$$

where

- \tilde{I}_1 and \tilde{I}_2 are the zero padded versions of I_1 and I_2 , respectively.
- ΔX_{\max} is the maximum translation value for which the correlation function is evaluated.

In the 2D case the input signals $I_1(i, j)$ and $I_2(i, j)$ are zero padded in both axes. This is done according to Eq. (4.1) using the maximal translations $(\Delta X_{\max}, \Delta Y_{\max})$ in both axes.

4.1.2 Translation estimation in the polar domain

In the polar domain the estimation of the regular and cyclic correlations coincides. This is due to the fact that the angular axis $\vec{\theta}$ is cyclic over the interval $[0, 2\pi]$. Hence, no zero padding is needed to avoid the alignment of non-corresponding samples. Zero padding is conducted using Eq. (4.1) with $\Delta X_{\max} = 0$.

4.1.3 Translation estimation in the log-polar domain

In the log-polar domain a distinction is made between the angular axis $\vec{\theta}$ and the radial axis \vec{r} . The motion along the angular axis $\vec{\theta}$ is estimated according to section 4.1.2, while the motion along the radial axis \vec{r} is estimated according to section 4.1.1.

4.2 Simultaneous estimation of rotation and translation

Let $I_1(i, j)$ and $I_2(i, j)$ be the input images to be registered. $I_1^{(n)}(i, j)$ denotes the fact that I_1 evolves throughout the registration process, while $I_2(i, j)$ remains unchanged. We assume that initially $I_1^{(0)}(i, j) = I_1(i, j)$ ($n = 0$). The rotation and translation estimation algorithm operates as follows:

1. Let (m_1, l_1) and (m_2, l_2) be the sizes of $I_1(i, j)$ and $I_2(i, j)$, respectively. Then, at iteration $n = 0$, $I_1(i, j)$ and $I_2(i, j)$ are zero padded such that

$$m_1 = l_1 = m_2 = l_2 = 2^k, k \in \mathbb{Z}. \quad (4.2)$$

2. The magnitudes $M_1^{PP}(\theta_i, r_j)$ and $M_2^{PP}(\theta_i, r_j)$ of the PPFFTs of $I_1^{(n)}(i, j)$ and $I_2(i, j)$ are calculated, respectively.
3. The polar DFTs ,magnitudes $\widehat{M}_1^{Polar}(\theta_i, r_j)$ and $\widehat{M}_2^{Polar}(\theta_i, r_j)$ of $I_1^{(n)}(i, j)$ and $I_2(i, j)$ are substituted by $M_1^{PP}(\theta_i, r_j)$ and $M_2^{PP}(\theta_i, r_j)$ respectively.
4. The translation along the $\vec{\theta}$ axis of $M_1^{PP}(\theta_i, r_j)$ and $M_2^{PP}(\theta_i, r_j)$ is estimated using the procedure described in section 4.1.2. The result is denoted by $\Delta\theta_n$.
5. Let θ_n be the accumulated rotation angle estimated at iteration n

$$\theta_n \triangleq \sum_{i=0}^n \Delta\theta_i = \theta_{n-1} + \Delta\theta_n.$$

Then, the input image $I_1(i, j)$ is rotated by θ_n (around the center of the image) using the FFT based image rotation algorithm described in [7]. This rotation scheme is accurate and fast since only 1D FFT operations are used

$$I_1^{(n+1)}(\theta, r) = I_1^{(0)}(\theta + \theta_n, r), \quad n = 1, \dots$$

The rotation can be conducted around any pixel. We recommend to use the central pixel of $I_1(i, j)$ such that the bounding rectangular of the rotated image will be as small as possible.

6. Steps 2-5 are reiterated until the angular refitment term $\Delta\theta_n$ is smaller than a predefined threshold ε_θ , i.e. $|\Delta\theta_n| < \varepsilon_\theta$, or a predefined number of iterations n_{\max} is reached.
7. The rotated images $I_1^{(n)}(i, j)$ and $I_2(i, j)$ are used as inputs to the spatial domain translation estimation algorithm presented in section 4.1.1. The rotation ambiguity is resolved by the procedure presented in section 2.2.

4.3 Simultaneous estimation of rotation, translation and scaling

Let $I_1(i, j)$ and $I_2(i, j)$ be the input images to be registered. $I_1^{(n)}(i, j)$ denotes the fact that I_1 evolves throughout the registration process, while $I_2(i, j)$ remains unchanged. We assume that initially $I_1^{(0)}(i, j) = I_1(i, j)$ ($n = 0$). The scaling, rotation and translation estimation algorithm operates as follows:

1. Let (m_1, l_1) and (m_2, l_2) be the sizes of $I_1^{(n)}(i, j)$ and $I_2(i, j)$, respectively. Then, at iteration $n = 0$, $I_1^{(0)}(i, j)$ and $I_2(i, j)$ are zero padded such that

$$m_1 = l_1 = m_2 = l_2 = 2^k, \quad k \in \mathbb{Z}. \quad (4.3)$$

The padding is needed to abide by the limitations of the PPFFT described in section 3.

2. The magnitudes $M_1^{PP}(\theta, r)$ and $M_2^{PP}(\theta, r)$ of the PPFFTs of $I_1^{(n)}(i, j)$ and $I_2(i, j)$ are calculated, respectively.
3. Let

$$\text{base}^{\text{length}(M_2^{PP}(\theta_i, r_j))} = \text{length}(M_2^{PP}(\theta_i, r_j)) \quad (4.4)$$

be the logarithmic base used to define the log domain in the \vec{r} axis. The log-polar DFTs of $I_1^{(n)}(i, j)$ and $I_2(i, j)$ are approximated by $M_1^{PP}(\theta, r)$ and $M_2^{PP}(\theta, r)$, respectively, using the following procedure:

$$\begin{aligned} \widehat{M}_1^{\text{Log-Polar}}(i, j) &= M_1^{PP}(i, \lfloor \text{base}^j \rfloor) \\ \widehat{M}_2^{\text{Log-Polar}}(i, j) &= M_2^{PP}(i, \lfloor \text{base}^j \rfloor) \end{aligned} \quad (4.5)$$

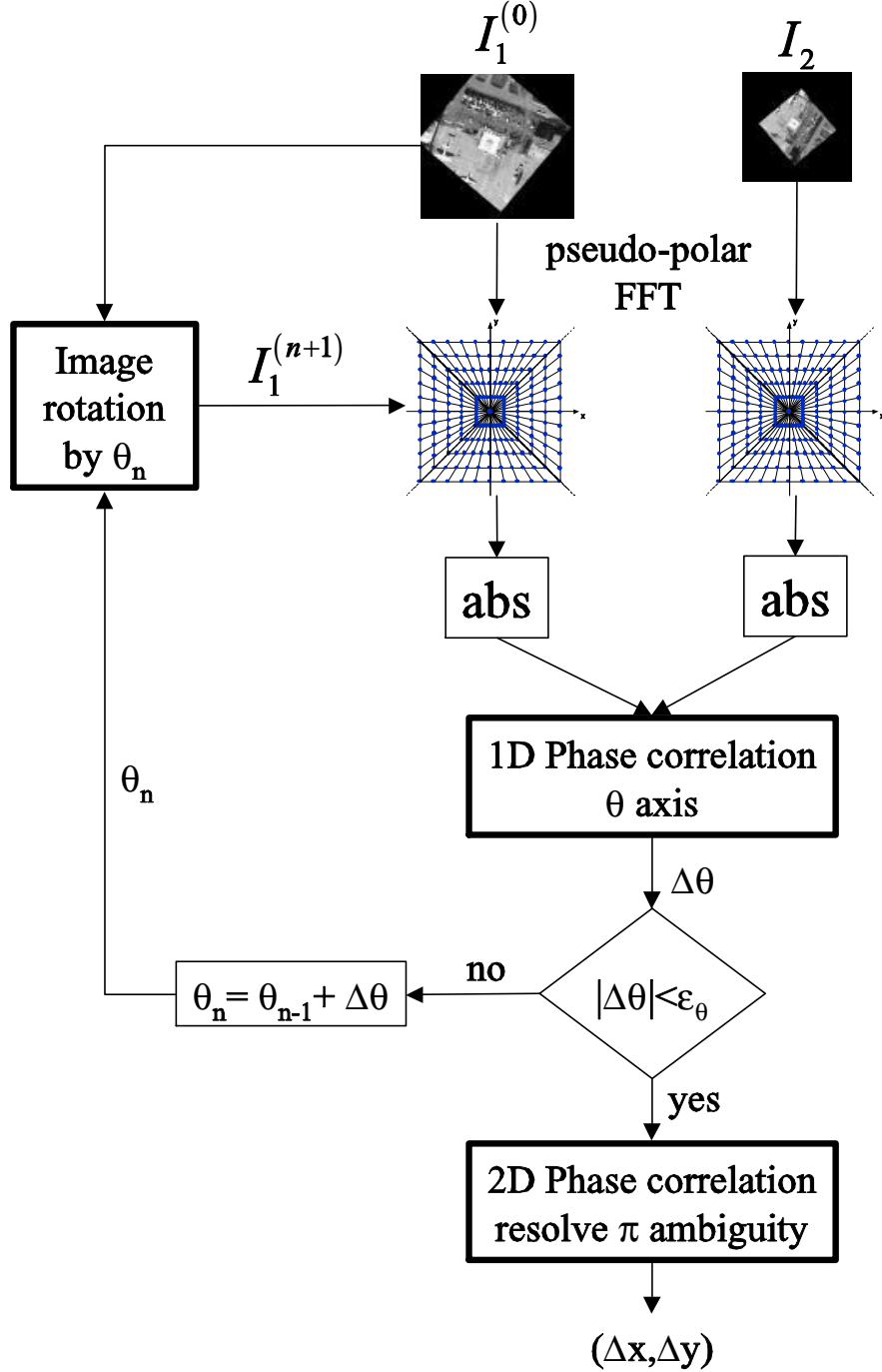


Figure 5: The flow of the pseudo-polar FFT based rotation and translation image registration algorithm: 1. The magnitude of the pseudo-polar FFT is calculated after the input images were zero padded to be of the same size. 2. The rotation $\Delta\theta$ is recovered using 1D phase-correlation on the θ axis in the pseudo-polar domain. 3. One of the input images is rotated by the accumulated rotation angle θ_n . 4. Steps 1-3 are reiterated until a stoppage condition is met. 5. The translation is recovered and the θ ambiguity is resolved by applying the 2D phase-correlation twice: using I_1 rotated by both θ and $\theta + \pi$.

where $\lfloor x \rfloor$ denotes the integral part of x ($x > 0$). The polar axis is approximated using the same procedure as in section 4.2, while the radial axis is approximated using nearest-neighbor interpolation.

4. The relative translation between $\widehat{M}_1^{Log-Polar}(i,j)$ and $\widehat{M}_2^{Log-Polar}(i,j)$ is recovered by a 2D phase correlation on the $\vec{\theta}$ and \vec{r} axes, which was described in section 4.1.3.
5. Let $\Delta\theta_n$ and Δr_n be the rotation angle and the scaling value estimated at iteration n , respectively. Then, the input image $I_1(x, y)$ is rotated (around the center of the image) [7] and then scaled using DFT domain zero padding

$$I_1^{(n+1)}(\theta, r) = I_1^{(0)}(\theta + \theta_n, r \cdot r_n) \quad (4.6)$$

where

$$\begin{aligned} \theta_n &= \sum_{i=0}^n \Delta\theta_i = \theta_{n-1} + \Delta\theta_n \\ r_n &= \prod_{i=0}^n \Delta r_i = r_{n-1} \cdot \Delta r_n. \end{aligned} \quad (4.7)$$

6. Steps 1-3 are reiterated until either the updated estimates are smaller than predefined thresholds $|\Delta\theta_n| < \varepsilon_\theta$ and $\Delta r_n < \varepsilon_r$ or a predefined number of iterations n_{\max} is reached.
7. The rotation and scale parameters θ_n and r_n , respectively, are used as inputs to the spatial domain registration algorithm presented in section 4.2 step 7.

4.4 Complexity estimation

The overall complexity of the proposed algorithm is dominated by the complexity of the 2D FFT:

- The pseudo-polar FFT is computed using $O(n^2 \log n)$ operations (Section 3).
- The fast image rotation algorithm [7] uses $O(n^2 \log n)$ operations.
- $O(n^2 \log n)$ operations are needed for image zooming using FFT domain zero padding.

5 Accuracy analysis of the polar and log-polar DFT estimation using the pseudo-polar FFT

This section provides an analysis of the performance of the registration algorithm by comparing it to prior state-of-the-art algorithms. By examining the proposed algorithms' flows presented in

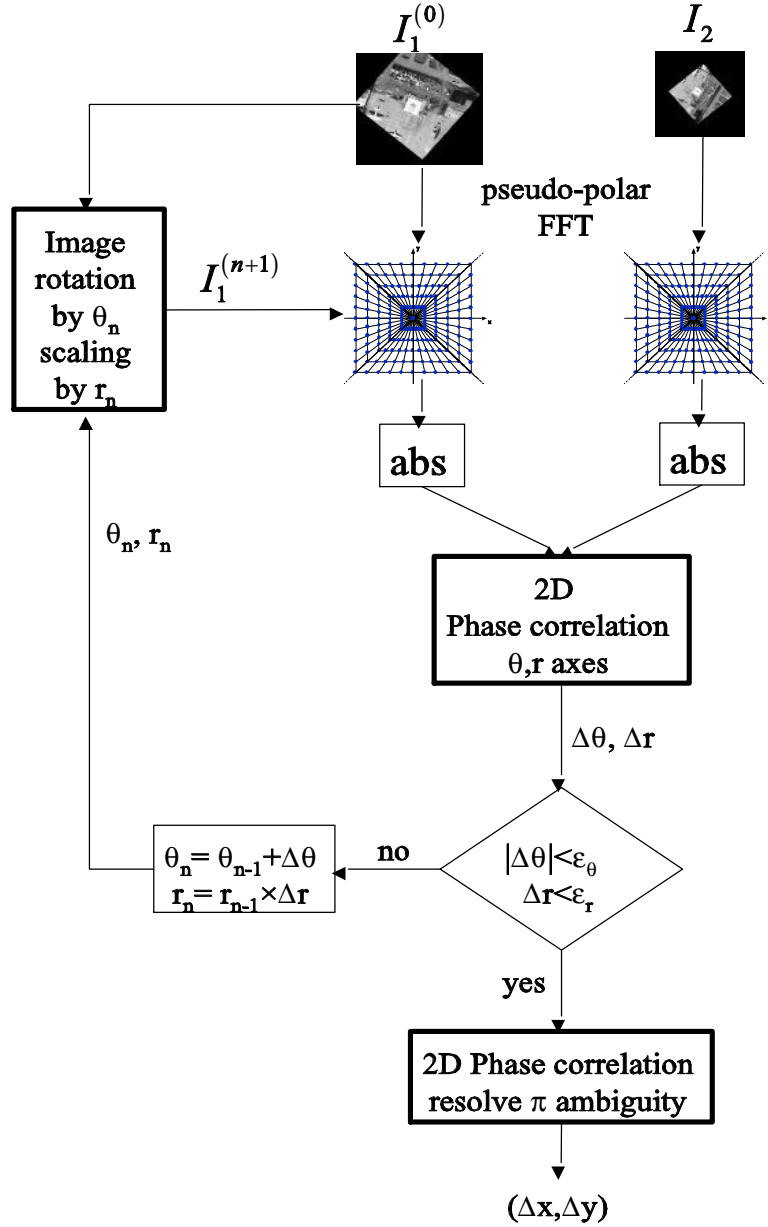


Figure 6: The flow of the pseudo-polar FFT based algorithm for scaling, rotation and translation estimation of image registration: 1. The magnitude of the pseudo-polar FFT is calculated after the input images were zero padded to be of the same size. 2. The rotation $\Delta\theta$ and scaling Δr are recovered using a 2D phase-correlation on θ and r axes in the pseudo-polar domain. 3. One of the input images is scaled and rotated by an accumulated rotation θ_n and scale Δr . 4. Steps 1-3 are reiterated until a stoppage criteria is met. 5. The translation is recovered and the $\pm\pi$ ambiguity in the value of θ is resolved by applying the 2D phase-correlation twice: using I_1 rotated by both θ and $\theta + \pi$.

sections 2 and 4, it follows that the values of the DFTs' magnitudes on polar and log-polar grids are estimated using the PFFFT grid points. In prior works [13, 16], the DFT's magnitudes on polar and log-polar grids were interpolated on a cartesian grid of the 2D DFT. The interpolation accuracy depends on the distance between the interpolated grid points (polar, log-polar) and the source grid used for the interpolation (pseudo-polar, cartesian). We show that the accumulated distance between the representations in the pseudo-polar and polar/log-polar grids is smaller than the accumulated distance between the cartesian and polar/log-polar grids. Hence, the registration accuracy of rotations and scalings is improved.

The angular accuracy of the PFFFT is analyzed in Section 5.1, while the radial accuracy analysis is given in Section 5.2. Accumulated distance calculation for several image sizes is given in Section 5.3, where it is compared to the accumulated distances of the cartesian grid used by [13, 16].

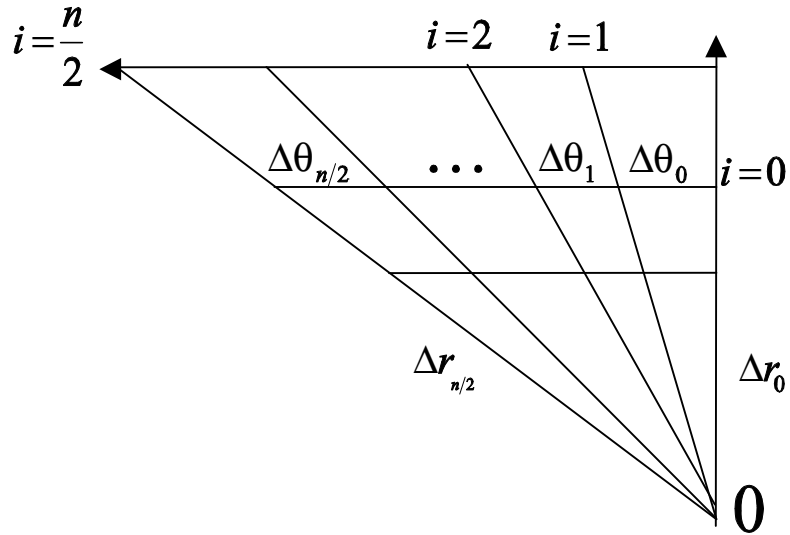


Figure 7: An illustration of the non-uniformity of the angular spacing of the pseudo-polar grid. $\Delta\theta_{PP}$, the angular spacing of the pseudo-polar grid becomes finer as the angle increases. The spacing varies smoothly as a function of the ray index.

5.1 Angular accuracy analysis

The polar grid shown in Fig. 7 is characterized by having a constant angular spacing

$$\Delta\theta_{Polar} = \frac{2\pi}{n} \quad (5.1)$$

where n is the size of the polar $\vec{\theta}$ axis. Following Eq. (3.2) and the notation used in section 3.2, $\Delta\theta_{PP}$, the PFFFT's angular change is given by

$$\begin{aligned}\Delta\theta_{PP}(i) &\triangleq \theta_2(i+1) - \theta_2(i) \\ &= \arctan\left(\frac{2(i+1)}{n}\right) - \arctan\left(\frac{2i}{n}\right), \quad i = 0, \dots, \frac{n}{2}\end{aligned}\quad (5.2)$$

where n is the size of a square image and $i = 0, \dots, \frac{n}{2}$ is the index of a ray.

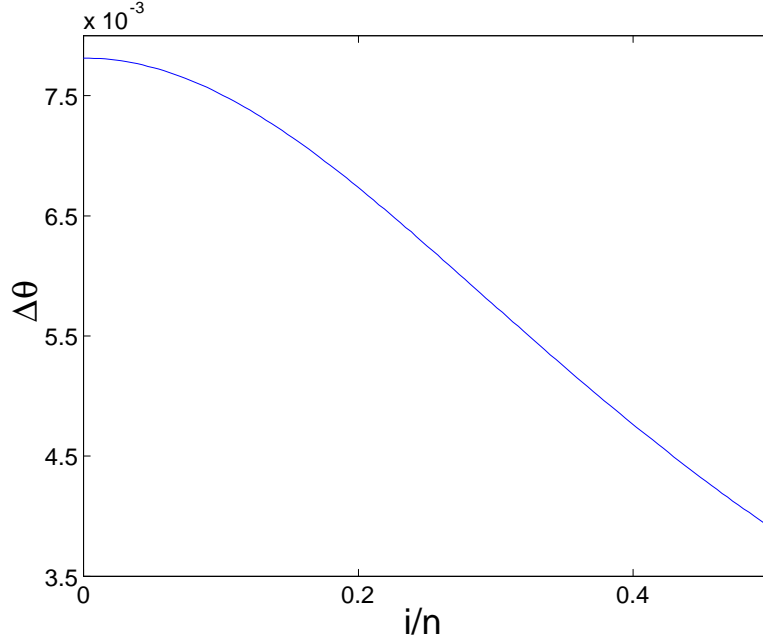


Figure 8: Illustration of the non-uniformity of $\Delta\theta_{PP}$, the angular spacing of the pseudo-polar grid.

In order to analyze the non-equally spaced pseudo-polar angular axis, we expand $\Delta\theta_{PP}$, the difference between two angles using a second order Taylor series expansion

$$\arctan(x+h) - \arctan(x) = \frac{1}{1+x^2}h + \frac{1}{2} \frac{2x}{(1+x^2)^2}h^2 + O(h)^3. \quad (5.3)$$

By substituting Eq. (5.2) into Eq. (5.3) ($x = \frac{2i}{n}, h = \frac{2}{n}$) we get

$$\begin{aligned}\widetilde{\Delta\theta_{PP}}(h, i) &\triangleq \frac{1}{1 + \left(\frac{2i}{n}\right)^2} \left(\frac{2}{n}\right) + \frac{\frac{2i}{n}}{\left(1 + \left(\frac{2i}{n}\right)^2\right)^2} \left(\frac{2}{n}\right)^2 \\ &= \frac{2n}{n^2 + 4i^2} + \frac{8i}{(n^2 + 4i^2)^2 n} \\ &= \frac{2n}{n^2 + 4i^2} + O\left(\frac{1}{n^5}\right).\end{aligned}\quad (5.4)$$

For common image sizes ($n \approx 300 - 500$), $O\left(\frac{1}{n^5}\right) = O(10^{-13})$ and the error term in Eq. (5.4) is negligible. Then,

$$\widetilde{\Delta\theta_{PP}} \approx \frac{2n}{n^2 + 4i^2} \quad (5.5)$$

and a constant angular spacing is achieved in low angles $2i \ll n$. The spacing which is illustrated in Fig. 8 changes smoothly as a function of i the initial ray index.

5.2 Radial accuracy analysis

Following Fig. 7, Δr_{PP} the radial step-size in the pseudo-polar grid is constant along the ray. As Δr_{PP} changes from ray to ray, for the j^{th} ray we have

$$\Delta r_{PP}(j) = \frac{\sqrt{\left(\frac{n}{2}\right)^2 + j^2}}{\left(\frac{n}{2}\right)} = \sqrt{1 + 4\left(\frac{j}{n}\right)^2}, \quad j \in 0, \dots, \frac{n}{2}. \quad (5.6)$$

Figure 9 shows the Δr_{PP} as a function of the ray index. For the polar grid we have $\Delta r(j) = 1$ for all $j \in 0, \dots, \frac{n}{2}$. We conclude that the radial spacing smoothly changes as a function of the ray

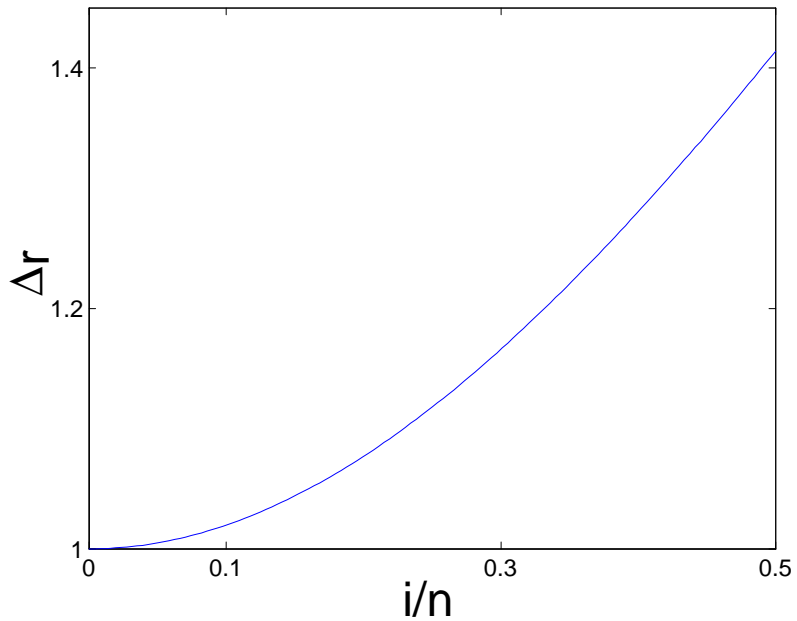


Figure 9: The pseudo-polar grid spacing Δr as it changes from ray to ray.

angle. The finest spacing $\Delta r_{PP} = 1$ is achieved for low angles $\theta \rightarrow \frac{\pi}{2}k, k \in \mathbb{Z}$, while the lowest resolution spacing is on the main diagonal with angles θ such that $\theta \rightarrow \frac{\pi}{4} + \frac{\pi}{2}k, k \in \mathbb{Z}$.

5.3 Accuracy analysis of the polar and log-polar DFT interpolation using the pseudo-polar FFT

We compare the accuracy of approximating the polar and log-polar DFT magnitudes using the PPFST and FFT evaluated on cartesian grids. Denote by ε_P and ε_{LP} , the approximation errors of the polar and log-polar DFTs' magnitudes, respectively

$$\begin{aligned}\varepsilon_{P/LP}(\theta_i, r_j) &= \left(\widehat{M}_1^{P/LP}(\theta_i, r_j) - M_1^{P/LP}(\theta_i, r_j) \right)^2 \\ &\approx \frac{\partial^2}{\partial\theta\partial r} M_1^{Polar} \varepsilon_{Grid}^2(\theta_i, r_j)\end{aligned}\quad (5.7)$$

where $M_1^{P/LP}(\theta_i, r_j)$ and $\widehat{M}_1^{P/LP}(\theta_i, r_j)$ are the magnitude and approximated magnitude of the polar/log-polar DFT at a spatial frequency (θ_i, r_j) . The interpolation error $\varepsilon_{P/LP}$, defined in Eq. (5.7), depends on the divergence of the pseudo-polar grid from the polar and log-polar pointset. Thus, the 1st order interpolation error (bilinear interpolation) can be estimated by

$$\varepsilon_{P/LP}(\theta_i, r_j) \approx \frac{\partial^2}{\partial\theta\partial r} M_1^{Polar} \varepsilon_{Grid}^2(\theta_i, r_j) \quad (5.8)$$

where $\varepsilon_{Grid}^2(\theta_i, r_j)$, which is the distance between the real polar point and the closest non-polar pixel, is given by

$$\varepsilon_{Grid}^2(\theta_i, r_j) = \left((X_{Pseudo-Polar}(i, j) - X_{Polar}(i, j))^2 + (Y_{Pseudo-Polar}(i, j) - Y_{Polar}(i, j))^2 \right)^2 \quad (5.9)$$

where

$$\{X_{Pseudo-Polar}(i, j), X_{Pseudo-Polar}(i, j)\} = \{\cos(\theta_i) \Delta r(j), \cos(\theta_i) \Delta r(j)\} \quad (5.10)$$

$$\{X_{Polar}(i, j), X_{Polar}(i, j)\} = \left\{ \cos\left(\frac{2\pi i}{n}\right) \cdot j, \cos\left(\frac{2\pi i}{n}\right) \cdot j \right\} \quad (5.11)$$

and the pseudo-polar coordinates θ_i and $\Delta r(j)$ are defined in Eqs. (3.1) and (5.6), respectively.

The 2nd order error (cubic interpolation) is given by

$$\varepsilon_{P/LP}(\theta_i, r_j) \sim \varepsilon_{Grid}^3(\theta_i, r_j). \quad (5.12)$$

In order to evaluate the grid resampling accuracy of 1st and 2nd order interpolations, the approximation errors are estimated by

$$\begin{aligned}\varepsilon_{Grid}^1 &= \sum_{i,j} \varepsilon_{Grid}^2(\theta_i, r_j) |\nabla^2 M(\theta_i, r_j)| \\ \varepsilon_{Grid}^2 &= \sum_{i,j} |\varepsilon_{Grid}|^3(\theta_i, r_j) |\nabla^3 M(\theta_i, r_j)|\end{aligned}\quad (5.13)$$

where $M(\theta_i, r_j)$ is the magnitude of the 2D FFT. We can look at Eq. (5.13) as a weighted sum of grid errors where the weights are the magnitudes' derivatives. This weighted representation is significant since most of the energy of natural image's spectra is concentrated in low frequencies.

The testbed for evaluating Eq. (5.13) was the *lena* image whose log-magnitude is presented in Fig. 10. The energy concentration in low frequencies is evident. The error estimations ε_{Grid}^1 and ε_{Grid}^2 were computed in the following way: *lena*'s DFT magnitudes were calculated on grids of sizes 64×64 , 128×128 and 256×256 . The derivatives of these magnitudes were computed and used to numerically evaluate Eq. (5.13). The results are presented in Table 2.

| Interpolation accuracy improvement | | | |
|------------------------------------|-------------------|------------------|------------------|
| Interpolation mode | Grid size [pixel] | | |
| | 64×64 | 128×128 | 256×256 |
| Bilinear in polar grid | 18.3% | 15.5% | 13.1% |
| Bilinear in log-polar grid | 56.2% | 53.4% | 49.3% |
| Cubic in polar grid | 26.5% | 18.2% | 10.5% |
| Cubic in log-polar grid | 67.2% | 57.7% | 63.0% |

Table 2: The reduction in the approximation error when the pseudo-polar based algorithm is used. The error's decrease is computed relatively to the error of the polar/log-polar grid approximation algorithms which are based on cartesian DFT. The proposed pseudo-polar based algorithm improves the approximation accuracy. We get better results with higher order approximations. We can see that using bigger grids in most cases does not increase the gain.

6 Experimental results

The registration algorithm was tested on the 128×128 *Lena* and *Airfield* images, which were rotated, scaled and translated to create the test pairs shown in Figs. 11 and 12 similar to [16, 20]. The images were registered using the rotation and rotation/scale algorithms presented in sections 4.2 and 4.3, respectively. The results are graphically presented by overlaying the edges of one of the images on the other using the estimated motion parameters, while the corresponding numerical results are presented in tables 3 and 4. The translation was estimated using the phase-correlation algorithm discussed in sections 2.1 and 4.1.1 whose accuracy is limited to integral translation values. We note that the choice of integral scaling factors for the experiments is of no special significance

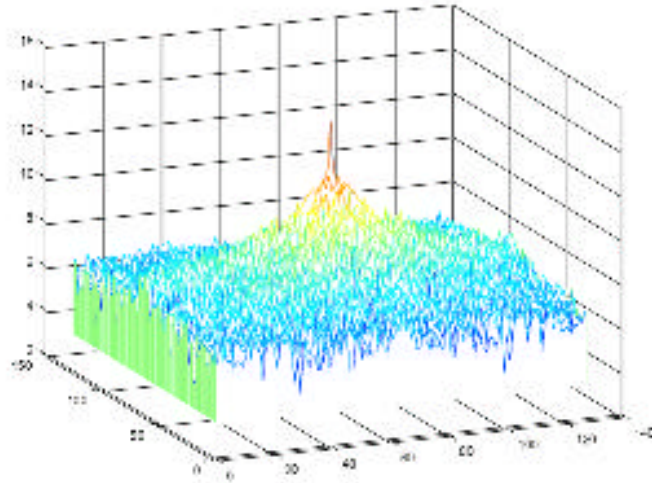


Figure 10: The log of the magnitude of the DFT of lena. This is a typical example of the energy distribution of the DFT magnitude of natural images, where most of the image energy is concentrated in low frequencies.

for the performance of the proposed algorithm, which was able to register any scaling factor within its dynamic range that is discussed later.

| Images | Correct ($Scale, \theta, \Delta x, \Delta y$) | Registration mode | Computed ($Scale, \theta, \Delta x, \Delta y$) |
|--------|--|----------------------------|---|
| a | (0.5, -100° , 19.0, 28.0) | scale,rotation,translation | (0.489, -100.15° , 19, 28) |
| b | (1, 130° , 10.5, 9.8) | rotation,translation | (1, 129.72° , 7, 71) |
| b | (1, 130° , 10.5, 9.8) | scale,rotation,translation | diverged |
| c | (1, 130° , 10.5, 9.8) | rotation,translation | (1, 129.72° , 10, 10) |

Table 3: Registration results of the image sets presented in Figure 11 using the pseudo-polar based algorithm. The images are rotated and scaled by small scaling factors. The results demonstrate the algorithm’s robustness even when noise is present and there is a partial alignment.

Table 3 and Fig. 11 present the registration results of the *Airfield* image which contains man made objects characterized by sharp edges surrounded by smooth regions. The algorithm was able to robustly register the images in Fig. 11a having a scaling factor of 2 and a large rotation of 100° . This test was repeated for various rotation angles resulting in a similar accurate registrations. Figures 11b and 11c present a situation of partial alignment under large rotation. The registration

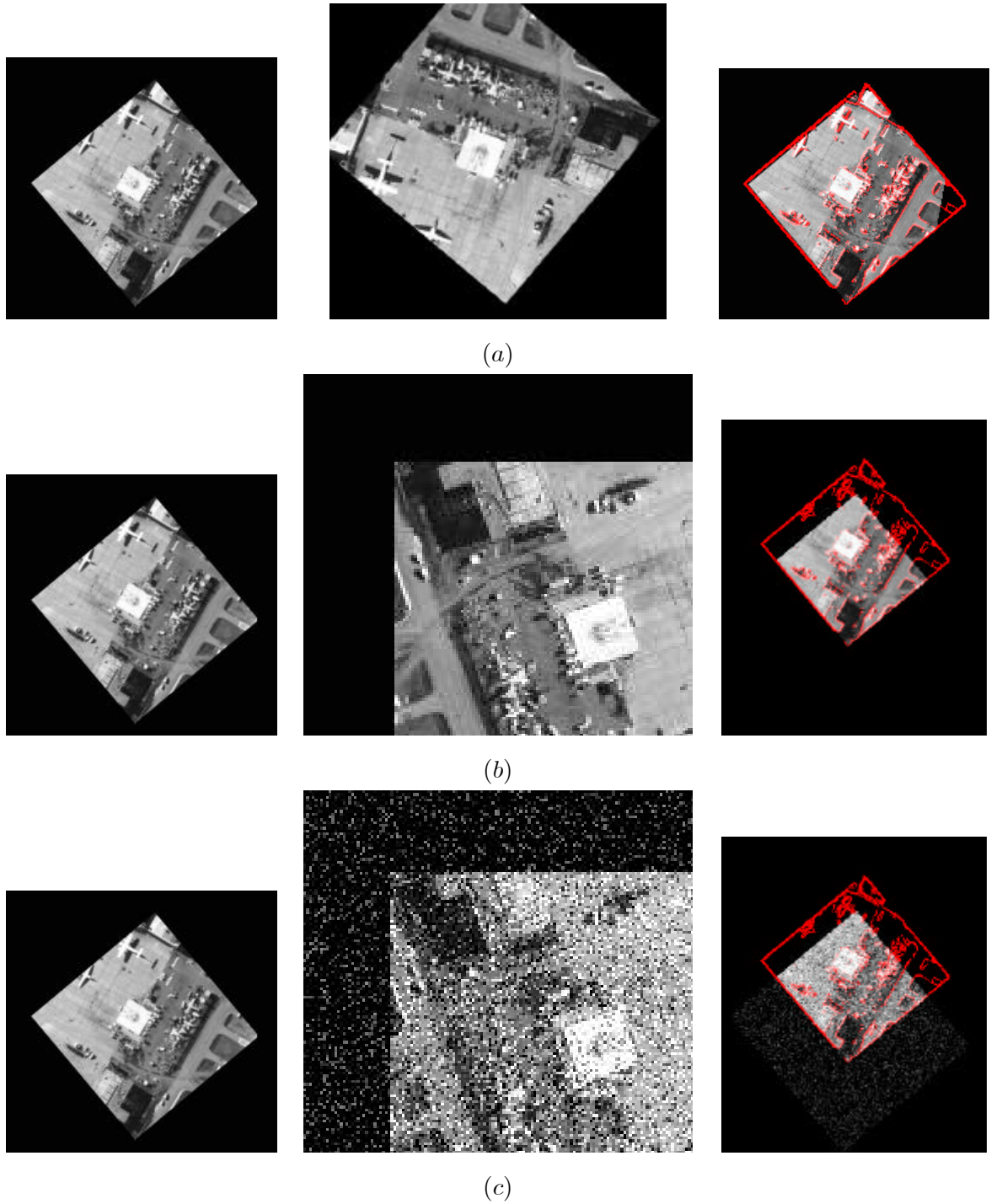


Figure 11: Registration results of *Airfield* image. In each row, the left and center images were registered. The edges of the images in the center column are overlaid on the left images to illustrate the registration accuracy. The results are shown in the right column. (a) Scale = 2, Angle = 100° . (b) Registration of images with partial non-overlapping areas: Scale=1, Angle = 130° . (c) Robustness was tested by adding significant noise to the images used in (b).

algorithm was able to register the images since the DFT’s magnitude is the Periodogram [23] and thus it is an approximation of the image’s power spectrum [23]. These statistical properties are invariant to partial alignment when the non-corresponding image parts have similar statistical properties. The robustness property was examined using the images in Fig. 11c, where significant noise was added to one of the images and registration was conducted using rotation and rotation/scale registration modes. In both cases the translation is evaluated afterward. When the rotation mode is used, we were able to register images in both noisy and noise-free scenarios, while the algorithm diverged when registering the images using the rotation/scale mode. The numerical results in table 3 show that when the algorithm converges it achieves high accuracy. No consistent misalignments (“local minima”) were observed. The scale/rotation mode was found to be less stable resulting in the algorithm’s failure to estimate the motion in Fig. 11b using this mode. When the algorithm converged, convergence was achieved within 3-4 iterations for all image sets while consuming 5-7 seconds of computational time on a 1.5MHz P4 Win2000 computer, where the algorithm was implemented using non-optimized C++.

The images in Fig. 12 were used to test the algorithm for registration of images with large scaling factors. Prior algorithms [13, 16] reported to recover scalings up to a factor of 2. The proposed algorithm was successfully tested for scalings factors up to 4. The graphical registration results shown in Fig. 12 are accurate, where the numerical results are presented in table 4. Figures 12a, 12c and 12d demonstrate the algorithm’s ability to recover large image scalings. We were not able to consistently recover scalings larger than 4. Computation times were similar to the results in Table 3 and Fig. 11. We conclude that the proposed algorithm is able to register images with scalings up to 4 with good accuracy. Within this range any rotation angle can be estimated while translations are estimated.

| Image sequence | Registration mode | Correct ($Scale, \theta$) | Computed ($Scale, \theta$) |
|----------------|----------------------------|-----------------------------|------------------------------|
| a | scale,rotation,translation | (4.0, 105°) | (4.0, 103.78°) |
| b | rotation,translation | (1, 150°) | (1, 150.11°) |
| b | scale,rotation,translation | (1, 150°) | <i>diverged</i> |
| c | scale,rotation,translation | (3, 130°) | (3.01, 129.64°) |
| d | scale,rotation,translation | (4, -135°) | (3.91, -135.35°) |

Table 4: Registration results of the *Lena* images presented in Figure 12 using the pseudo-polar based algorithm.

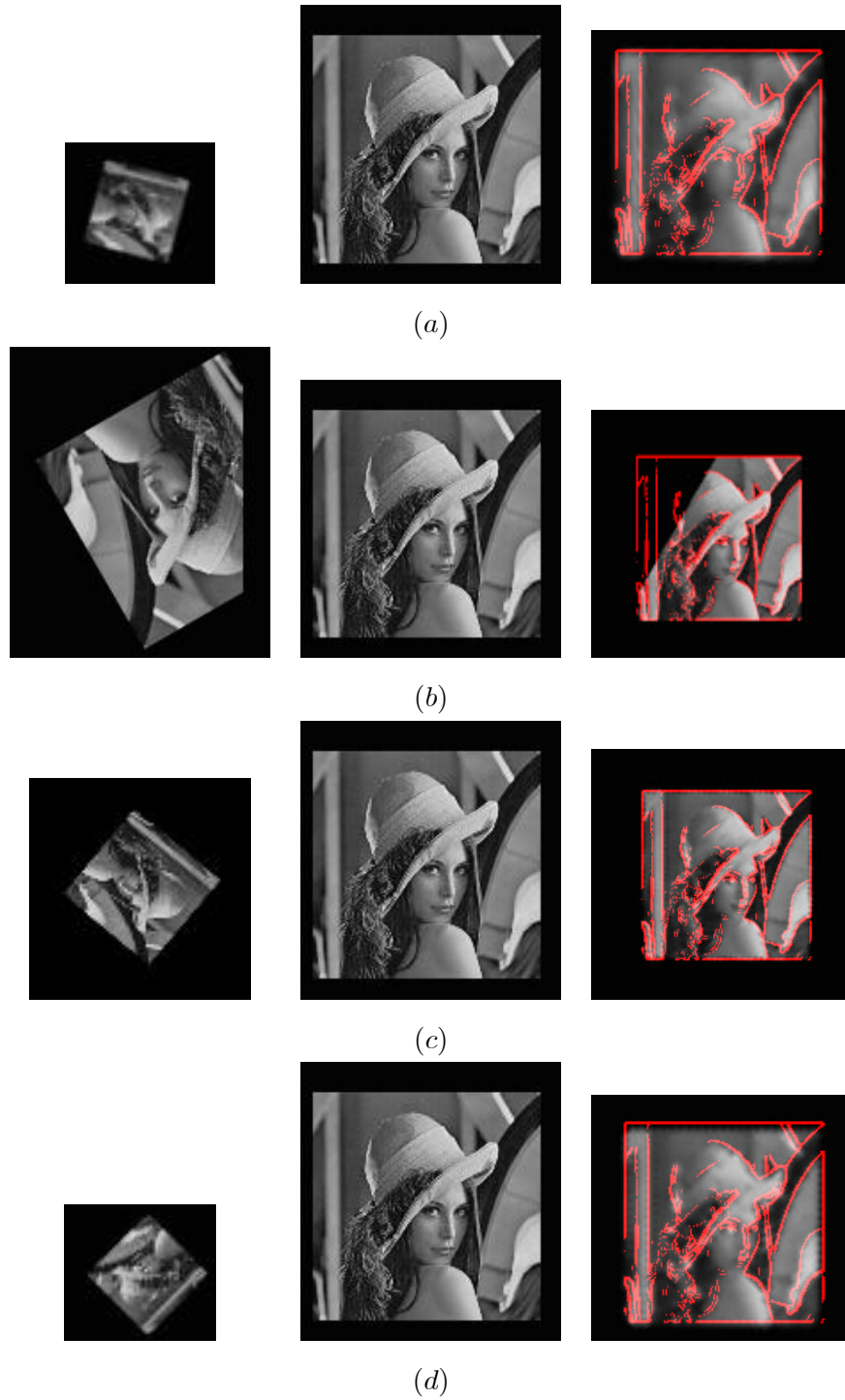


Figure 12: Registration results of the *Lena* image when large scalings factors are present. In each row, the left and center images were registered. The edges of the center image are overlaid on the left image to illustrate the registration accuracy. The output is given in the right column. (a) Scale = 4, Angle = 105°. (b) Registration of images with partial non overlapping areas: Scale=1, Angle = 150°. (c) Scale = 3, Angle = 150°. (d) Scale = 4, Angle = -135°.

7 Conclusions

In this paper we proposed a FFT based image registration algorithm, which was shown to be able to recover large rotations and scaling factors. This algorithm which uses the pseudo-polar FFT [27] enhance the current state-of-the-art image registration algorithms. The overall complexity is dominated by FFT operations and is $O(n^2 \log n)$. The algorithm can achieve close to real-time performance by using machine specific optimized FFT implementations which are widely available.

References

- [1] F. Dufaux and J. Konrad, "Efficient, Robust, and Fast Global Motion Estimation for Video Coding", IEEE Transactions on Image Processing, vol. 9, no. 3, pp. 497-501, March 2000.
- [2] A. Tekalp, "Digital Video Processing", Prentice Hall, 1995.
- [3] M. Irani and S. Peleg, "Motion Analysis for Image Enhancement: Resolution, Occlusion and Transparency". Journal of Visual Communication and Image Representation, Vol. 4, No. 4, pp.324-335, December 1993.
- [4] S .Man and R. W. Picard, "Virtual Bellows: constructing high quality stills from video", Proc. IEEE Int. Conf. Image Processing Austin, TX, Nov. 13-16, 1994.
- [5] R. Szeliski, "Image Mosaicking for Tele-Reality Applications",CRL 94/2, May 1994.
- [6] M. Irani, P. Anandan, and S. Hsu. "Mosaic based representations of video sequences and their applications", International Conference on Computer Vision, pages 605-611, 1995.
- [7] M. Unser, P. Thvenaz, and L. Yaroslavsky, "Convolution-based interpolation for fast, high-quality rotation of images" IEEE Trans. Image Processing, vol. 4, pp. 1371-1381, Oct. 1995.
- [8] S. Kruger and A. Calway, "Image Registration Using Multiresolution Frequency Domain Correlation". In British Machine Vision Conference, pages 316-325. British Machine Vision Association, September 1998
- [9] D. J. Fleet, "Disparity from local weighted phase-correlation". In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, pages 48-56, San Antonio 1994.
- [10] C. D. Kuglin and D. C. Hines. "The phase correlation image alignment method", IEEE Conference on Cybernetics and Society, pp. 163-165, September 1975.

- [11] H. Foroosh; J.B. Zerubia, M. Berthod, “Extension of phase correlation to subpixel registration”, *IEEE Trans. on Image Processing* vol. 11 no. 3 , March 2002, pp. 188 -200, March 2002.
- [12] M. Peyman, “Projection-based, Frequency-Domain Estimation of Superimposed Translational Motions”, *Journal of the Optical Society of America: A, Optics and Image Science*, vol. 13, no. 11, pp. 2151-2162, Nov. 1996.
- [13] M. Peyman, “Two Dimensional Matched Filtering for Motion Estimation” , *IEEE Transactions on Image Processing*, vol. 8, no. 3, March 1999, pp.438-443.
- [14] Y. Chou and H. Hang, “A New Motion Estimation Method using Frequency Components”, *Journal of Visual Communication and Image Representation*, Vol. 8, No. 1, pp. 83-96, 1997.
- [15] H. Stone, M. Orchard and E. Chang, “Subpixel Registration of Images”, *33rd Asilomar Conference on Signals, Systems, and Computers*, October 1999
- [16] S. Reddy and B. N. Chatterji. “An FFT-based technique for translation, rotation, and scale-invariant image registration” . *IEEE Trans. on Image Processing*, 3(8):1266–1270, August 1996.
- [17] A. Taberero, J. Portilla,R. Navarro, “Duality of Log-Polar Image Representations in the Space and Spatial-frequency Domains”, *IEEE Trans. on Signal Processing*, vol. 47, No. 9, pp. 2469-2479, September 1999.
- [18] L. Lucchese, “A Frequency Domain Technique Based on Energy Radial Projections for Robust Estimation of Global 2D Affine Transformations”, *Computer Vision and Image Understanding* Vol. 82, No. 1, pp. 72–116, Apr 2001.
- [19] S. Derrode, F. Ghorbel , “Robust and Efficient Fourier-Mellin Transform Approximations for Gray-Level Image Reconstruction and Complete Invariant Description”, *Computer Vision and Image Understanding*, Vol. 83, No. 1, pp. 57-78, July 2001.
- [20] G. Wolberg and S. Zokai, “Robust Image Registration Using Log-Polar Transform”, *Proc. IEEE Intl. Conference on Image Processing*, Vancouver, Canada, September 2000.
- [21] Q. Chen, M. Defrise, and F. Deconinck. “Symmetric phase-only matched filtering of Fourier-Mellin transforms for image registration and recognition” . *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(12):1156–1168, December 1994.
- [22] S.A.Kruger and A.D.Calway, “Image registration using multiresolution frequency domain correlation”, *Proc. British Machine Vision Conf*, 1998.

- [23] B. Porat, “A Course in Digital Signal Processing”, John Wiley Pub.,1997.
- [24] A.V. Oppenheim and R. W. Schaffer, “Discrete-Time Signal Processing”, New Jersey: Prentice-Hall, 1998.
- [25] A. Averbuch, Y. Keller, “A unified approach to FFT based image registration”, submitted to the IEEE Trans. On Image Processing.
- [26] J. Ruanaidh and T. Pun, “Rotation, scale and translation invariant digital image watermarking”, in Proceedings IEEE International Conference on Image Processing 1997 (ICIP 97), Santa Barbara, CA, USA, vol. 1, pp. 536–539, October 1997.
- [27] A. Averbuch, D.L. Donoho, R.R Coifman, M. Israeli, and Y. Shkolnisky, “Fast slant stack: A notion of Radon transform for data in cartesian grid which is rapidly computable, algebraically exact, geometrically faithful and invertible”, to appear in SIAM Scientific Computing.
- [28] A. Averbuch, D. Donoho, Y. Shkolnisky, “The 2D discrete Radon transform”, submitted to Fourier Analysis and Applications, 2002.
- [29] P.N. Swartztrauber and D. H. Bailey, “The fractional Fourier transform and applications”, SIAM Review, 33(3):389-404, September 1991.
- [30] A. Averbuch, Y. Keller, “FFT based image registration”, IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP2002), Orlando, USA, May 2002.