BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF MECHANICAL ENGINEERING

FAKULTA STROJNÍHO INŽENÝRSTVÍ

INSTITUTE OF MATHEMATICS

ÚSTAV MATEMATIKY

HIGH PRECISION SUB-PIXEL IMAGE REGISTRATION METHODS AND THEIR APPLICATIONS IN ASTROPHYSICS

HIGH PRECISION SUB-PIXEL IMAGE REGISTRATION METHODS AND THEIR APPLICATIONS IN ASTROPHYSICS

DOCTORAL THESIS

DIZERTAČNÍ PRÁCE

AUTHOR AUTOR PRÁCE	Ing. Zdeněk Hrazdíra
SUPERVISOR ŠKOLITEL	prof. RNDr. Miloslav Druckmüller, CSc.

BRNO 2022

Abstract

Many scientific observations and measurement techniques, that rely on data from images, include an image registration step. The results of such techniques thus often heavily rely on the precision of the image registration. This thesis describes a novel, robust and highly accurate sub-pixel image registration method (based on the standard phase correlation image registration method), and its applications in various fields.

Abstrakt

Mnoho vědeckých pozorování a měření, které pracují s obrazovými daty, obsahují krok, ve kterém je nutno dané obrazy registrovat (navzájem sesadit). Výsledky těchto metod tak často silně závisí na přesnosti registrace obrazů. Tato práce popisuje novou, robustní a vysoce přesnou metodu sub-pixelové registrace obrazů (založenou na registraci standardní fázovou korelací) a její aplikace v různých oborech.

Keywords

image registration, cross-correlation, phase correlation, Fourier transform, sub-pixel accuracy, optimization

Klíčová slova

registrace obrazů, křížová korelace, fázová korelace, Fourierova transformance, sub-pixelová přesnost, optimalizace

I declare that I have written the doctoral thesis topic *High precision sub-pixel image registration methods and their applications in astrophysics* on my own according to advice of my doctoral thesis supervisor prof. RNDr. Miloslav Druckmüller, CSc., and using the sources listed in references.

February 6, 2023 Ing. Zdeněk Hrazdíra

I would like to thank my supervisor prof. RNDr. Miloslav Druckmüller, CSc. for his guidance, support and especially for his great patience and goodwill in explaining some specific astronomical and mathematical topics.

Ing. Zdeněk Hrazdíra

Contents

Introduction

The need to effectively and reliably solve the image registration problem arises in many domains [Foroosh et al., 2002], ranging from medical and satellite imaging [Jenkinson and Smith, 2001, Mahmood and Lee, 2019, Do et al., 2019, Li et al., 2019] to optical flow [Lefébure and Cohen, 2001], experimental mechanics [Bing et al., 2006], 3D reconstruction [Gravel et al., 2012, Li et al., 2018, Casella et al., 2019], astrophysics [Chen et al., 2014, Zhou and Yu, 2018, Shapiro et al., 2013] and many more.

Intensity-based methods (methods based on a notion of correlation) are one of the most frequently used and widely-known techniques in this domain [Leng et al., 2019], mostly because of their relative computational efficiency, which is often achieved by a clever use of the convolution/correlation theorem (1.8) together with the Fast Fourier Transform algorithm [Brigham and Morrow, 1967]. Popular intensity-based image registration methods include the cross-correlation method (2.1.1) and the phase correlation method (2.1.2), upon which many of the state of the art methods are built [Debella-Gilo and Kääb, 2011, Heid and Kääb, 2012, Abdou, 1998, Foroosh et al., 2002, Michel and Rignot, 1999, Balci and Foroosh, 2006, Alba et al., 2015, Young and Driggers, 2006, Zhang et al., 2011].

Contrary to intensity-based image registratiom methods, feature-based methods do not produce a single image shift estimate for the whole registered image pair, but a set of varying image shift estimates at multiple automatically detected locations. Feature-based image registration methods are therefore well suited for registering images with non-uniform motion. Many feature detection and description methods exist, each with its own advantages and drawbacks, which include speed, accuracy, robustness, ease of use (number of parameters) and availability (license type). Popular and frequently used feature detectors/descriptors include AKAZE [Alcantarilla et al., 2011], BRISK [Leutenegger et al., 2011], KAZE [Alcantarilla et al., 2012], ORB [Rublee et al., 2011], SIFT [Lowe, 1999], SURF [Bay et al., 2008], BRIEF [Calonder et al., 2010], FREAK [Alahi et al., 2012], LATCH [Levi and Hassner, 2016] and DAISY [Tola et al., 2009].

The main results of this work include various novel astrophysical applications of state of the art intensity-based and feature-based image registration methods, along with a detailed description and evaluation of a novel, robust and highly accurate sub-pixel image registration method and some of its many potential applications. The novel Iterative Phase Correlation (IPC) method is an extension of the standard phase correlation method of image registration to the sub-pixel domain. The high sub-pixel accuracy of the IPC method is mainly suited for scientific measurements using high quality image data.

Mathematical foundations of the image registration methods described in this work are presented in Chapter 1. A general overview of the most frequently used image registration techniques can be found in Chapter 2, containing popular intensity-based methods in Section 2.1 and popular feature-based methods in Section 2.2. The novel IPC image registration method is described in detail in Chapter 3, including additional optional improvements and procedures like false correlations removal (3.6), parameter optimization (3.7) and efficient and reliable implementation in code (3.8). Rigorous sub-pixel accuracy measurements for various image sizes and noise levels and comparison with other image registration methods can be found in Section 3.9. Various novel astrophysical applications of the IPC algorithm and other image registration algorithms are presented in Chapter 4, including dissimilar image alignment (4.1), solar differential rotation speed measurement (4.2) and solar wind speed measurement (4.3).

Chapter 1

Mathematical background

1.1 The Fourier transform

Cross-correlation and phase correlation image registration methods (which are the most widely used and reliable intensity-based methods) both rely upon various properties of the Fourier transform to extract the underlying image shift between two translated images. The Fourier transform is thus a principal tool in intensity-based image registration, and will be defined and described in this section, along with its relevant properties. Since the main concern of this thesis is image data, only the two-dimensional Fourier transform will be described and used.

The two-dimensional Fourier transform of an absolutely integrable function $f(x, y) : \mathbb{R}^2 \to \mathbb{C}$, denoted $\mathcal{F}\{f\} = \hat{f}(\xi, \eta) : \mathbb{R}^2 \to \mathbb{C}$, is defined by

$$\hat{f}(\xi, \eta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i(\xi x + \eta y)} \,\mathrm{d}x \,\mathrm{d}y \tag{1.1}$$

for any pair of real numbers ξ and η. Conversely, the inverse two-dimensional Fourier transform of an absolutely integrable function $\hat{f}(\xi, \eta) : \mathbb{R}^2 \to \mathbb{C}$, denoted $\mathcal{F}^{-1}\left\{\hat{f}\right\} = f(x, y) : \mathbb{R}^2 \to \mathbb{C}$, is defined by

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{f}(\xi, \eta) e^{2\pi i(\xi x + \eta y)} \,\mathrm{d}\xi \,\mathrm{d}\eta \tag{1.2}$$

for any pair of real numbers x and y. Furthermore, if $f(x, y)$ is continuous and if both f and $\mathcal{F}\{f\}$ are absolutely integrable, then the original function f can be reconstructed from its two-dimensional Fourier transform $\mathcal{F}\{f\}$ via the inverse two-dimensional Fourier transform, mathematically expressed as

$$\mathcal{F}^{-1}\{\mathcal{F}\{f(x, y)\}\} = f(x, y). \tag{1.3}$$

The Fourier transform has multiple properties which are instrumental in cross-correlation and phase correlation methods. The first important Fourier transform property is the Fourier shift theorem. The Fourier shift theorem relates the Fourier transform of a function to the Fourier transform of its shifted version, namely

$$\mathcal{F}\{f(x - x_0, y - y_0)\}(\xi, \eta) = \mathcal{F}\{f(x, y)\}(\xi, \eta) e^{-i(\xi x_0 + \eta y_0)}, \tag{1.4}$$

where x_0, y_0 are real constants. The second important Fourier transform property is the Fourier scale theorem, which relates the Fourier transform of a function to the Fourier transform of its scaled version, namely

$$\mathcal{F}\left\{f(ax,by)\right\}(\xi,\eta)=\frac{1}{|ab|}\mathcal{F}\left\{f(x,y)\right\}\left(\frac{\xi}{a},\frac{\eta}{b}\right), \tag{1.5}$$

where a, b are real constants. The third important Fourier transform property is the Fourier rotation theorem, which relates the Fourier transform of a function to the Fourier transform of its rotated version, namely

$$\mathcal{F}\left\{f(x\cos\theta-y\sin\theta,x\sin\theta+y\cos\theta)\right\}(\xi,\eta)=\mathcal{F}\left\{f(x,y)\right\}(\xi\cos\theta-\eta\sin\theta,\xi\sin\theta+\eta\cos\theta), \tag{1.6}$$

where θ is a real constant.

1.2 Convolution

Most of the signal processing techniques that utilize the Fourier transform also apply various filters to frequency representations, and image registration techniques are no exception. Problem-specific filtering in the frequency domain can often drastically improve final results of any image processing algorithm, but is tightly coupled with convolution in the image domain (and vice versa). The two-dimensional convolution of $f(x,y):\mathbb{R}^2\to\mathbb{C}$ and $g(x,y):\mathbb{R}^2\to\mathbb{C}$, denoted $f*g$, is defined by

$$f(x,y)*g(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(s,t)g(x-s,y-t)\,\mathrm{d}s\,\mathrm{d}t \tag{1.7}$$

for any pair of real numbers x and y. Convolution in the image domain is equivalent to point-wise multiplication (denoted by $\odot$) in the frequency domain, as stated by the convolution theorem, namely

$$\mathcal{F}\left\{f*g\right\}=\mathcal{F}\left\{f\right\}\odot\mathcal{F}\left\{g\right\}, \tag{1.8}$$

where f, g are two absolutely integrable functions. The inverse relation also holds, namely

$$\mathcal{F}\left\{f\odot g\right\}=\mathcal{F}\left\{f\right\}*\mathcal{F}\left\{g\right\}. \tag{1.9}$$

Proof of the convolution theorem can be found in [Bracewell and Bracewell, 1986, p. 117].

1.3 The log-polar transform

Another mathematical tool instrumental in intensity-based image registration is the log-polar transform. The log-polar transform remaps the Cartesian coordinates (x,y) with center (x_c,y_c) into log-polar coordinates (ρ,ϕ), where

$$\rho=\ln\left(\sqrt{(x-x_c)^2+(y-y_c)^2}\right), \tag{1.10}$$

$$\phi=\arctan\left(\frac{y-y_c}{x-x_c}\right). \tag{1.11}$$

The log-polar transform enables the Fourier scale and rotation theorems to be utilized to generalize image registration to images that are not only shifted, but also rotated and/or scaled. The log-polar transform effectively converts the task of finding the scale s and rotational angle φ between images into the task of finding the shift in the x and y directions, respectively (see Section 3.5).

1.4 Cross-correlation

Cross-correlation (denoted by $\star$, sometimes also called the "sliding dot product") is one of the simplest and most widely used measures of similarity between two signals. The cross-correlation value $c(x_0, y_0)$ of two functions $f(x,y) : \mathbb{R}^2 \to \mathbb{C}$ and $g(x,y) : \mathbb{R}^2 \to \mathbb{C}$, for a given x-shift x_0 and y-shift y_0, is defined by

$$f(x,y) \star g(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \overline{f(x_0,y_0)} g(x_0 + x, y_0 + y) \, \mathrm{d}x \, \mathrm{d}y \tag{1.12}$$

for any pair of real numbers x_0 and y_0. The convolution equation (Eq. 1.7) is very similar to the cross-correlation equation (Eq. 1.12), which can indeed be rewritten to a convolution form as

$$f(x,y) \star g(x,y) = \overline{f(-x,-y)} * g(x,y). \tag{1.13}$$

Using the Fourier conjugate property

$$\mathcal{F}\left\{\overline{f(x,y)}\right\} = \overline{\mathcal{F}\left\{f(-x,-y)\right\}} \tag{1.14}$$

together with the convolution theorem (Eq. 1.8), cross-correlation in the frequency domain (also frequently called the cross-power spectrum, denoted by $C(\xi,\eta)$) can be expressed as

$$C(\xi,\eta) = \mathcal{F}\left\{f(x,y) \star g(x,y)\right\} = \overline{\mathcal{F}\left\{f(x,y)\right\}} \mathcal{F}\left\{g(x,y)\right\}. \tag{1.15}$$

Applying the inverse Fourier transform yields the convolution theorem version of the cross-correlation equation

$$c(x,y) = \mathcal{F}^{-1}\left\{C(\xi,\eta)\right\} = \mathcal{F}^{-1}\left\{\overline{\mathcal{F}\left\{f(x,y)\right\}} \mathcal{F}\left\{g(x,y)\right\}\right\}. \tag{1.16}$$

1.5 Phase correlation

Inspired by the Fourier shift theorem (Eq. 1.4), phase correlation is another measure of similarity between two signals, based on their relative displacement. Given a function $f_1(x,y)$ and its shifted version $f_2(x,y) = f_1(x - x_0, y - y_0)$, the cross-power spectrum (Eq. 1.15) can be expressed as

$$C(\xi,\eta) = \mathcal{F}\left\{f_1(x,y)\right\} \overline{\mathcal{F}\left\{f_2(x,y)\right\}} = \mathcal{F}\left\{f_1(x,y)\right\} \overline{\mathcal{F}\left\{f_1(x - x_0, y - y_0)\right\}}, \tag{1.17}$$

which after normalization (eliminating magnitude information, leaving phase information only, hence the name "phase correlation") becomes the normalized cross-power spectrum

$$P(\xi,\eta) = \frac{\mathcal{F}\left\{f_1(x,y)\right\} \overline{\mathcal{F}\left\{f_1(x - x_0, y - y_0)\right\}}}{\left|\mathcal{F}\left\{f_1(x,y)\right\} \mathcal{F}\left\{f_1(x - x_0, y - y_0)\right\}\right|}. \tag{1.18}$$

Applying the Fourier shift theorem, the normalized cross-power spectrum can be expressed as

$$P(\xi,\eta) = \frac{\mathcal{F}\left\{f_1(x,y)\right\}\overline{\mathcal{F}\left\{f_1(x,y)\right\}}\left(\overline{e^{-i(\xi x_0+\eta y_0)}}\right)}{\left|\mathcal{F}\left\{f_1(x,y)\right\}\mathcal{F}\left\{f_1(x,y)\right\}\right|} = e^{i(\xi x_0+\eta y_0)}. \tag{1.19}$$

The normalized cross-power spectrum of a function and its shifted version is thus a complex exponential, which is coincidentally the Fourier transform of a shifted Dirac distribution [Bracewell and Bracewell, 1986] (denoted by $\delta(x,y)$), namely

$$e^{i(\xi x_0+\eta y_0)} = \mathcal{F}\left\{\delta(x+x_0,y+y_0)\right\}. \tag{1.20}$$

Combining equations 1.19 and 1.20 and using the inverse Fourier transform yields the phase correlation (denoted $p(x,y)$) of a function and its shifted version, which is the Dirac distribution shifted by the corresponding underlying shift (x_0,y_0), and can be calculated as

$$\delta(x+x_0,y+y_0) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{f_1(x,y)\right\}\overline{\mathcal{F}\left\{f_1(x-x_0,y-y_0)\right\}}}{\left|\mathcal{F}\left\{f_1(x,y)\right\}\mathcal{F}\left\{f_1(x-x_0,y-y_0)\right\}\right|}\right\}. \tag{1.21}$$

Equation 1.21 can be generalized for any two functions $f(x,y)$ and $g(x,y)$, yielding the phase correlation equation

$$p(x,y) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{f(x,y)\right\}\overline{\mathcal{F}\left\{g(x,y)\right\}}}{\left|\mathcal{F}\left\{f(x,y)\right\}\mathcal{F}\left\{g(x,y)\right\}\right|}\right\}, \tag{1.22}$$

where $p(x,y)$ converges to a shifted Dirac distribution as $g(x,y)$ converges to a shifted version of $f(x,y)$.

Chapter 2

Image registration methods

2.1 Intensity-based methods

2.1.1 The cross-correlation method

The cross-correlation function (defined in Eq. 1.16) describes a measure of similarity between two input signals, based on their relative displacement. Thus, for each pair of sufficiently similar regions in the two correlated signals, the cross-correlation function will contain a local maximum - a distinct “peak” at the location of the corresponding relative displacement. The value of the cross-correlation function in each of these local maxima depends on the extent of similarity, and therefore may vary significantly across all local maxima.

To measure the relative x and y shift (denoted Δx and Δy) between images $I_1(x, y)$ and $I_2(x, y)$ via the cross-correlation image registration method, firstly, the Fourier transform of the first image (denoted $\mathcal{F}\left\{I_1(x, y)\right\}$) and the complex conjugate of the Fourier transform of the second image (denoted $\overline{\mathcal{F}\left\{I_2(x, y)\right\}}$) are computed. These Fourier spectra are then multiplied element-wise (denoted by $\odot$), and an inverse Fourier transform of the result is calculated. This results in the discrete cross-correlation landscape $c(x, y)$.

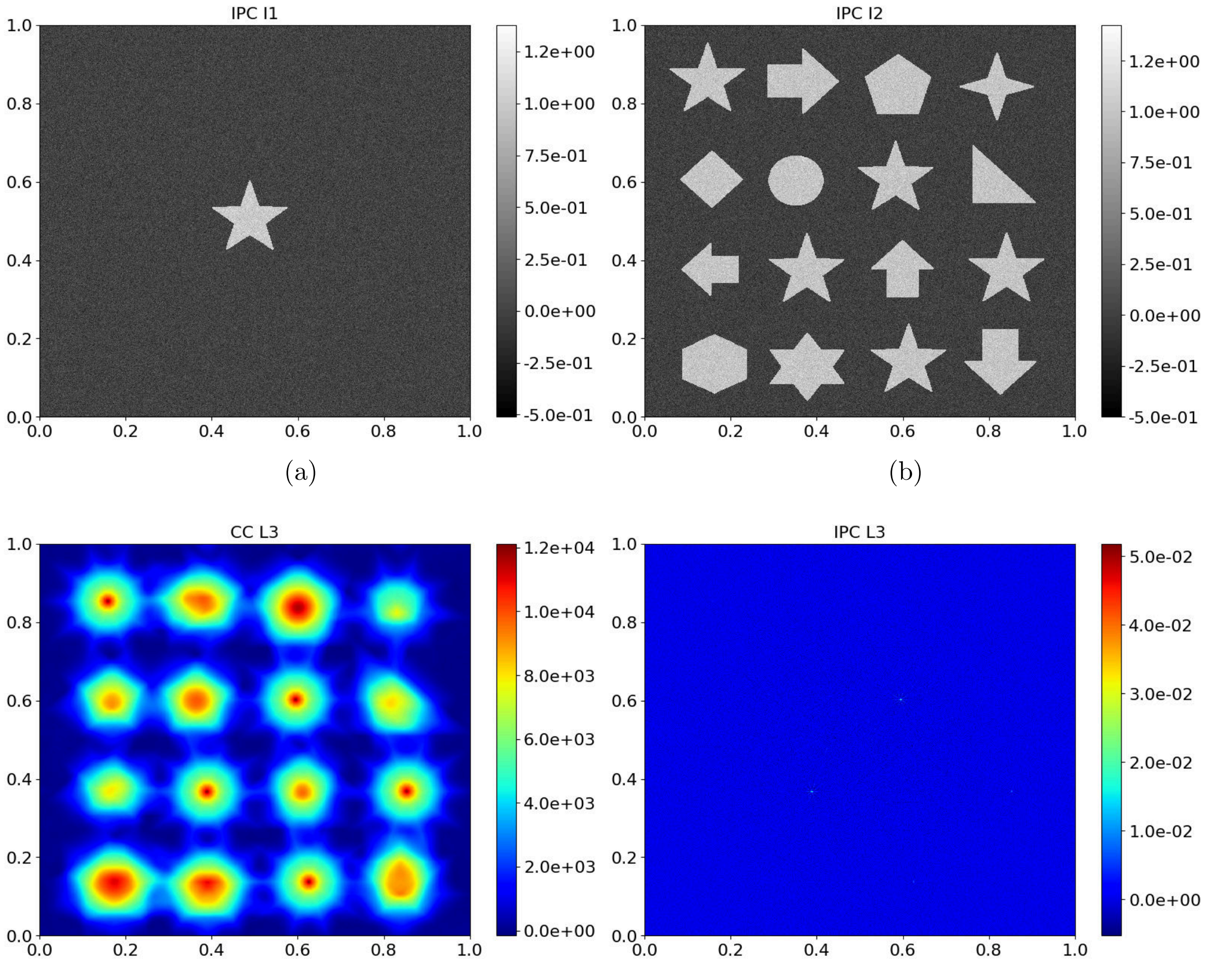
To produce the estimate of the image shift $[\Delta x, \Delta y]$, the location of the maximum correlation value (maximum element in $c(x, y)$) is found. The $[x, y]$ coordinates of the maximum correlation location are then the corresponding cross-correlation image registration shift estimates, mathematically expressed as

$$[\Delta x, \Delta y] = \underset{x,y}{\arg\max} \left(\mathcal{F}^{-1} \left\{ \mathcal{F}\left\{I_1(x, y)\right\} \odot \overline{\mathcal{F}\left\{I_2(x, y)\right\}} \right\} \right). \tag{2.1}$$

2.1.2 The phase correlation method

Similarly to the cross-correlation function, the phase correlation function (defined in Eq. 1.22) also describes the similarity between two input signals, based on their relative displacement. The process of obtaining the correlation landscape with phase correlation is also very similar, the only difference being that the cross power spectrum (the result of multiplying the Fourier transforms together) is normalized before the inverse Fourier transform is applied, mathematically expressed as

$$[\Delta x, \Delta y] = \underset{x,y}{\arg\max} \left(\mathcal{F}^{-1} \left\{ \frac{\mathcal{F}\left\{I_1(x, y)\right\} \odot \overline{\mathcal{F}\left\{I_2(x, y)\right\}}}{\left| \mathcal{F}\left\{I_1(x, y)\right\} \odot \overline{\mathcal{F}\left\{I_2(x, y)\right\}} \right|} \right\} \right). \tag{2.2}$$



(a) (b)

(c) (d)

Figure 2.1: Differences between cross-correlation and phase correlation image registration landscapes on a pair of artificially noised ($\sigma = 0.1$) 1024×1024 images. First input image $I_1(x, y)$ (a), second input image $I_2(x, y)$ (b), the corresponding cross-correlation landscape $c(x, y)$ (c) and the corresponding phase correlation landscape $p(x, y)$ (d).

The process of normalization in the frequency domain affects the resulting correlation landscape greatly. A typical example of the difference caused by frequency domain normalization can be seen in Fig. 2.1 (c-d). The cross-correlation landscape $c(x, y)$ (Fig. 2.1 (c)) contains strong and relatively flat local maxima corresponding to each shape in $I_2(x, y)$, even if the shape is vastly different. The correct local maxima corresponding to the five "star" shapes cannot be clearly separated from the other (false) local maxima corresponding to the rest of the shapes.

On the contrary, the phase correlation landscape $p(x, y)$ (Fig. 2.1 (d)) contains only very sharp local maxima corresponding to the five correct star shapes. Additionally, the local maxima in $p(x, y)$ are very sharp and localized (mostly just a couple of pixels wide), which can sometimes make them hard for a human to identify, compared to the very flat cross-correlation local maxima. The sharpness of the phase correlation local maxima also generally holds for a relatively wide range of noise amounts. The extreme sharpness of the phase correlation local maxima gives the phase correlation method great accuracy and resolution advantage over the cross-correlation method.

2.1.3 Sub-pixel correlation extensions

The resulting image shift estimate obtained by the standard cross-correlation and phase correlation methods is inherently restricted to integer values, since it directly corresponds to the location of maximal correlation. Multiple newer image registration techniques aim to improve the registration accuracy of the phase correlation method by extending it to the sub-pixel domain [Alba et al., 2015]. These correlation extensions employ many various ideas, e.g. image upsampling, quadratic/Gaussian/sinc fitting in correlation space, linear fitting in frequency space, local correlation center of mass (centroid), upsampling in the frequency space and more.

Image upsampling

The simplest sub-pixel extension of any image registration method is image upsampling (described in Debella-Gilo and Kääb [2011]). In this method, both input images are upsampled by an upsampling factor $U > 1$ (usually via bilinear or bicubic interpolation) prior to being registered. After registration, the integer image shift estimates are scaled by $1/U$ to produce the sub-pixel image shift estimates. This method, however, is seldom used in practice, since the computational overhead of bilinear or bicubic interpolation of both input images and the subsequent upsampled DFT computations is often too large for common applications (for example, to obtain 0.1px sub-pixel accuracy, one has to compute two $10\times$ two-dimensional interpolations).

Correlation function fitting

Correlation function fitting approaches produce sub-pixel image shift estimates by fitting a neighborhood of the maximum value location of the discrete correlation function $c(x, y)$ by a continuous function, followed by an exact analytical calculation of its maximum location.

Abdou [1998] and Argyriou and Vlachos [2006] explored correlation fitting with a bivariate quadratic function of the form

$$c_Q(x, y) = a_0x^2 + a_1y^2 + a_2xy + a_3x + a_4y + a_5, \tag{2.3}$$

whose maximum location can be found by setting both partial derivatives $\partial_x c_Q(x, y)$ and $\partial_y c_Q(x, y)$ to zero, resulting in a sub-pixel image shift estimate of the form

$$\Delta x = \frac{a_2a_3 - 2a_4a_0}{4a_0a_1 - a_2^2}, \tag{2.4}$$

$$\Delta y = -\frac{a_2\Delta x + a_3}{2a_0}. \tag{2.5}$$

Since the bivariate quadratic function $c_Q(x, y)$ is determined by six coefficients $a_0, \dots, a_5$, six discrete integer-valued fitting points need to be selected. Usually, the point corresponding to the maximum correlation value and its four nearest neighbors are used for this purpose. The sixth point can be arbitrarily selected from any of the four cornering neighbors of the maximum correlation point. However, the arbitrary choice introduces some bias, which can be circumvented by calculating average $a_0, \dots, a_5$ of all four choices of the sixth cornering neighbor fitting point. Alternatively, least squares function fitting can be used to fit any number of neighboring points.

Takita et al. [2003] explored correlation fitting with a shifted sinc function of the form

$$c_S(x,y) = \frac{\alpha}{N_x N_y} \frac{\sin\left(\pi(x+\Delta x)\right)}{\sin\left(\frac{\pi}{N_x}\left(x+\Delta x\right)\right)} \frac{\sin\left(\pi(y+\Delta y)\right)}{\sin\left(\frac{\pi}{N_y}\left(y+\Delta y\right)\right)}, \tag{2.6}$$

where optimal parameters α, Δx and Δy are found via least squares function fitting (using gradient descent methods) with 3×3, 5×5, 7×7 and 9×9 fitting points neighboring the correlation peak.

Local correlation center of mass

Michel and Rignot [1999] used the idea of a local correlation center of mass (centroid) in a neighborhood of the correlation peak to obtain image shift estimates with sub-pixel accuracy. The sub-pixel centroid of a maximum correlation location neighborhood $\hat{C}(x,y)$ can be calculated by

$$\Delta x = \frac{\sum_x \sum_y x\hat{C}(x,y)}{\sum_x \sum_y \hat{C}(x,y)}, \Delta y = \frac{\sum_x \sum_y y\hat{C}(x,y)}{\sum_x \sum_y \hat{C}(x,y)}. \tag{2.7}$$

The size of the neighborhood $\hat{C}(x,y)$ is routinely set to 5×5 px. Furthermore, a weighted centroid calculation can be used to give more weight to pixels near the integer maximum correlation location, potentially improving method robustness.

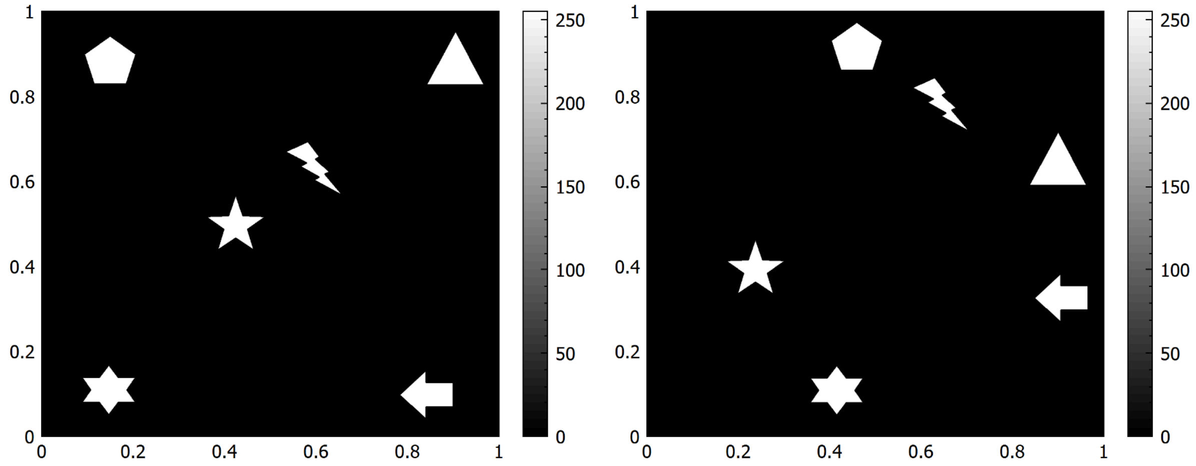
2.2 Feature-based methods

For some types of image registration tasks, using intensity-based (correlation) image registration algorithms yields very poor results. Such tasks mostly include registering image pairs, that contain multiple distinct regions, which are shifted by a different amount and/or in a different direction, or registering images with non-stationary regions and a stationary background. In these cases, intensity-based image registration methods have difficulty estimating the correct image shift, since there is no single correct underlying image shift for the whole image. The correlation landscapes for such image pairs often include multiple similarly strong peaks, and for this reason, the resulting image shift estimates become unreliable and tend to oscillate between multiple values.

The only possible solution (using intensity-based methods) in such case is to partition the registered images into equally-sized sub-regions and estimate the image shift individually for each sub-region. However, this approach only works, if the non-uniform overall image shift becomes sufficiently close to being uniform in each sub-region. It might be challenging to estimate the correct shape and number (and therefore size) of sub-regions in the partition, so that each sub-region sufficiently fulfills the image shift uniformity condition. Furthermore, for small sub-regions, the resolution might become insufficient to obtain a reliable image shift estimate.

2.2.1 Feature-based image registration

The feature-based image registration process consists of three main steps - feature detection, feature description and feature matching. Firstly, both images are scanned and



(a)

(b)

Figure 2.2: An example of an image pair with non-uniform motion.

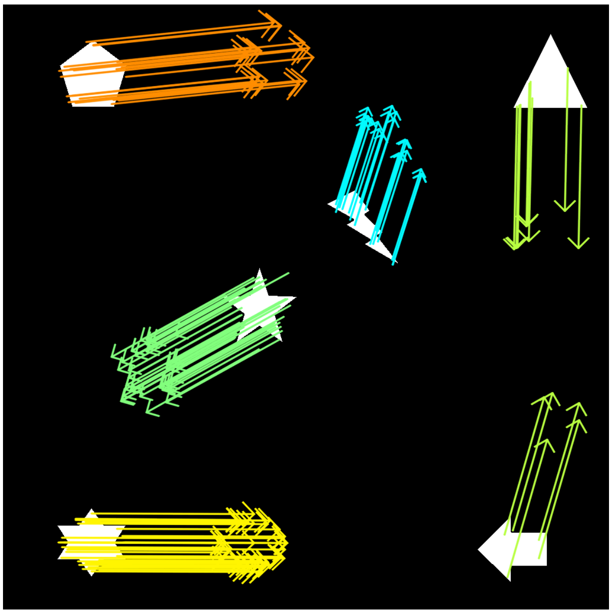
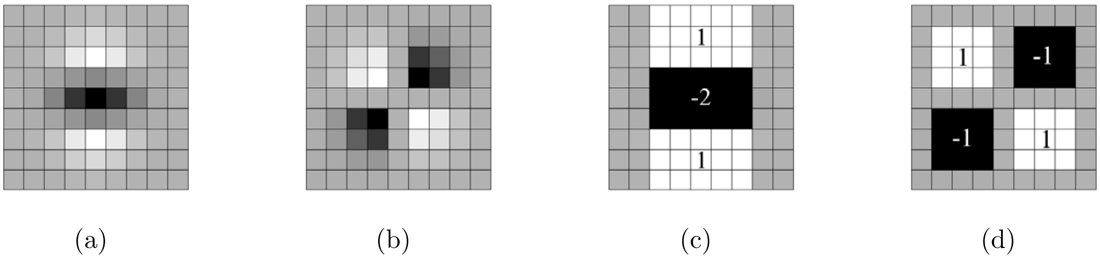


Figure 2.3: Feature-based image registration on a pair of images from Fig. 2.2. Arrows indicate the measured shift for each detected keypoint.



(a) (b) (c) (d)

Figure 2.4: Left to right: Gaussian second order partial derivatives (discretised and cropped) in the y and xy directions, and their approximations using box filters. The grey regions are equal to zero. [Bay et al., 2008]

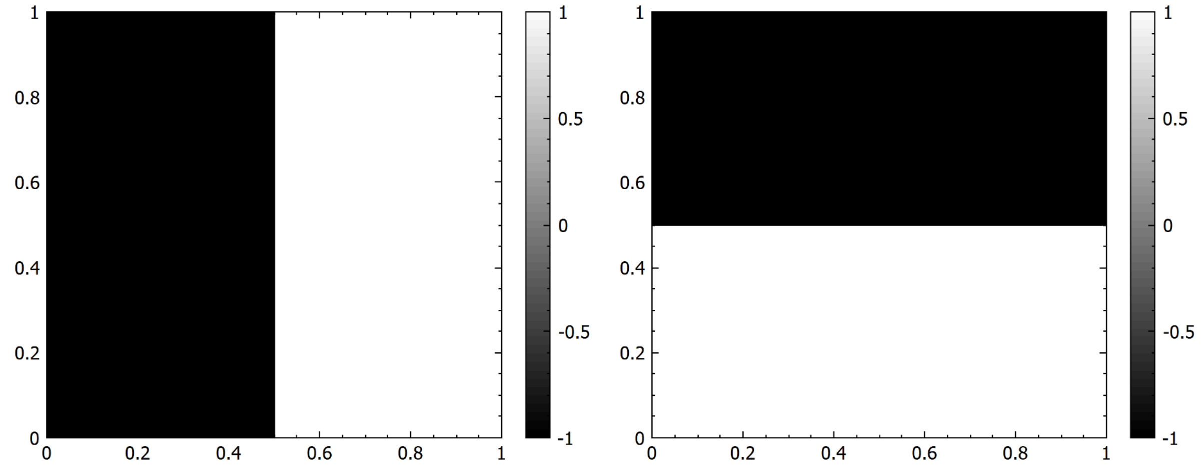
points of interest (keypoints) are detected in each image (feature detection). Secondly, feature descriptors are computed for each of the detected keypoints, carrying information about the local neighborhood of the keypoint (feature description). Finally, the descriptors of the keypoints detected in the first image are matched with the descriptors of the keypoints detected in the second image, yielding a set of matching keypoint pairs. The resulting image shift estimate for the given keypoint pair is then simply the difference between the corresponding keypoint locations.

2.2.2 Feature detection

Among the state of the art feature detector/descriptor pairs, the Speeded up robust features (SURF) feature detector and descriptor [Bay et al., 2008] seems to be often giving the most reliable results. The SURF algorithm is based on the well-known and frequently used Scale-invariant feature transform (SIFT) feature detection/description method [Lowe, 1999]. The SURF method uses the determinant of the Hessian matrix to find points of interest. In order for the method to be scale invariant, this process is done at multiple different scales (denoted s), where higher scales (lower resolutions) are achieved by repeated blurring and sub-sampling of the original image - a process known as an image pyramid [Adelson et al., 1984]. Points with maximal Hessian determinant values are chosen, using a threshold determinant value, which is set before the start of the feature detection process. Gaussian second order partial derivatives are used to compute the Hessian matrix for each pixel, where the (discretised and cropped) Gaussian smoothing (Fig. 2.4 (a), (b)) is further approximated/simplified by box filters (Fig. 2.4 (c), (d)). For example, for scale $s = 1.2$, a 9×9 box filter is used as an approximation of a Gaussian with $\sigma = 1.2$, as can be seen in Fig. 2.4. Furthermore, to improve keypoint localisation and uniqueness in both space and scale, a 3×3×3 non-maximum suppression (in the x, y and s directions) is applied to the Hessian determinant values. This eliminates potential keypoints (points where the determinant of the Hessian matrix is greater than the threshold value), which are not in a local Hessian determinant maximum.

2.2.3 Feature description

To describe the local neighborhood of each detected keypoint, firstly, the SURF method extracts a square region with size $20s$ centered around the keypoint. Secondly, the square region is further regularly subdivided into 16 (4×4) smaller sub-regions, and for each sub-



(a)

(b)

Figure 2.5: Haar wavelet filters in the x (a) and y (b) directions used to compute the SURF descriptor vector components d_x and d_y.

region, Haar wavelet responses (see Fig. 2.5) in horizontal and vertical directions (denoted d_x and d_y) are computed for a set of 25 (5×5) regularly spaced sample points. The size of the Haar wavelet window is $2s$. These Haar wavelet responses are weighted with a Gaussian ($\sigma = 3.3s$) centered around the keypoint, to increase the robustness towards geometric deformations and noise. Thirdly, the wavelet responses (and their magnitudes) are summed up over all of the 25 sample points, yielding a four-dimensional descriptor vector

$$\mathbf{v} = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y| \right)$$

for each sub-region. Finally, all of the sub-region descriptor vectors are concatenated and form the final keypoint descriptor, which with 16 total sub-regions sums up to a 64-dimensional descriptor.

2.2.4 Feature matching

After the descriptors are computed for all of the detected keypoints in both images, the descriptors from the first image are matched with the descriptors from the second image. This matching is done simply by a brute-force method, which means, that for a given keypoint descriptor in the first image, a distance metric is computed for all of the keypoint descriptors in the second image, and the keypoint in the first image is then paired with the keypoint with the lowest distance (highest similarity) in the second image. The distance metric used for the descriptor matching is the L^2 metric.

To further increase the robustness of the keypoint descriptor matching process, unreliable matches are detected and discarded. The detection of unreliable matches is done by Lowe's ratio test [Lowe, 2004]. To perform Lowe's ratio test, during the brute-force matching, the second best match (with the second lowest distance) is also recorded. The ratio test assumes, that if the match is reliable, the descriptor distance to the best match should be substantially lower than the distance to the second best match, and thus the ratio between these distances should be low. Lowe [2004] demonstrated on a 40 000 keypoint database, that this assumption is indeed true for the vast majority of cases. Matches with descriptor distance ratio higher than a threshold value (e.g. 0.5) are deemed unreliable, and are discarded. In other words, this means, that for a given keypoint in the first image, a match with a keypoint in the second image (which minimizes the L^2 descriptor distance) is deemed reliable only if the best distance is at least twice lower (assuming a ratio threshold of 0.5) than the second best descriptor distance.

Chapter 3

The Iterative phase correlation method

The Iterative Phase Correlation (IPC) image registration method is a sub-pixel extension of the standard phase correlation algorithm. The registration accuracy improvement is obtained mainly by the means of cross-power spectrum filtering, correlation interpolation and iterative sub-pixel centroid refinement, all of which will be discussed in detail in the sections to come. It is important to mention, that exactly the same extension can also be applied to the cross-correlation algorithm. However, the standard phase correlation approach frequently outperforms the standard cross-correlation approach (in the sense of accuracy, resolution and brightness invariance), and thus is the preferred candidate for further attempts of accuracy improvement.

3.1 Image windowing

Before the first and most computationally expensive phase correlation step of computing the discrete Fourier transforms (DFTs) of both input images, it often proves beneficial to multiply both images by a window function $W(x, y)$ (or simply just window), to reduce the effects of spectral leakage. The term "spectral leakage" originates from the fact, that the DFT power of a given underlying image frequency often "leaks" to the adjacent frequency bins. The effects of spectral leakage are present in all DFTs of aperiodic images, due to the periodic input assumption of the discrete Fourier transform.

The simplest approach to mitigate such negative DFT effects is setting all pixels on the image boundary to the same value (usually zero), or letting the image pixels unmodified but increasing the image size by adding black (zero) borders on all sides, which effectively makes the image periodic in both x and y directions. This approach, however, introduces large gradients near the image boundary for most real-world images. The resulting DFT power spectrum is then often corrupted with large values at high frequencies, which arise as a direct consequence of the large image boundary gradients. This simple constant boundary approach thus resolves one negative DFT effect, but introduces another, which often affects the resulting DFT spectrum even more severely. As previously mentioned, the approach, which then usually works the best, is the multiplication of the whole image by a window function $W(x, y)$, which modifies the image minimally near the center, and tends smoothly towards zero near the image borders. Such window then does not introduce any significant high frequency noise to the resulting DFT, while mitigating the spectral

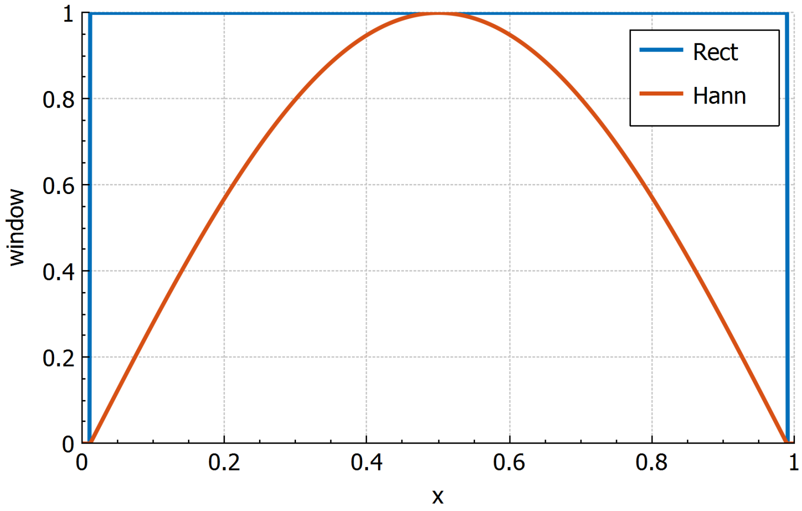


Figure 3.1: Common image window functions - a rectangular window (blue) and a Hann window (orange).

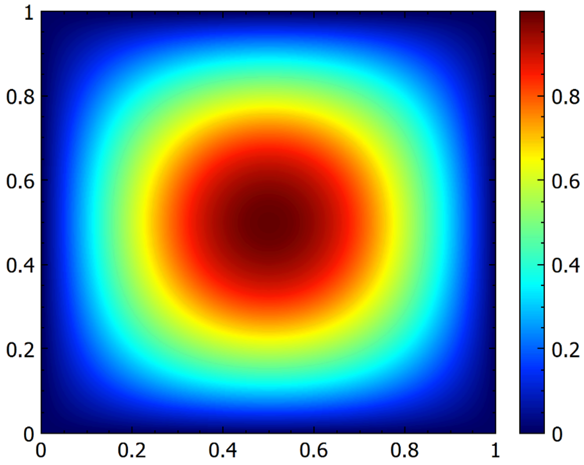


Figure 3.2: The two-dimensional Hann window function.

leakage effect.

It is very important to mention, that due to the Fourier transform convolution theorem property, multiplying the entire image $I(x, y)$ point-wise with a window function $W(x, y)$ is equivalent to performing a convolution of the DFT of $I(x, y)$ with the DFT of $W(x, y)$, namely

$$\mathcal{F}\left\{I\left(x,y\right)\odot W\left(x,y\right)\right\} = \mathcal{F}\left\{I(x,y)\right\} * \mathcal{F}\left\{W(x,y)\right\}. \tag{3.1}$$

Therefore, a good window function $W(x, y)$ should not only be smooth and tend towards zero near the image borders, but also have most of the power concentrated near the center (origin) of the DFT power spectrum, to minimize the introduced parasitic convolution effects on the DFT. For example, since the Fourier transform of a rectangular window is a sinc function, the resulting DFT of the original (not windowed) image often contains “ripples” (also called “ringing” effects), which arise due to the convolution of the image DFT with a sinc function.

There are many types of possible window functions, most popular of which are the

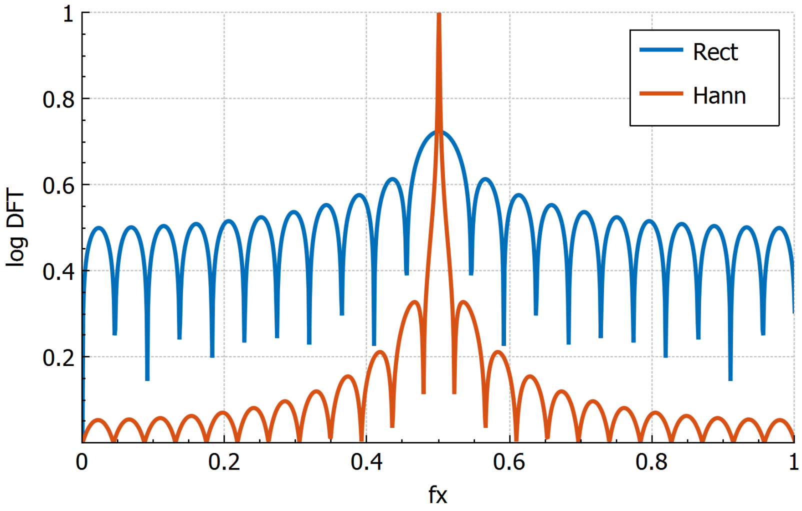
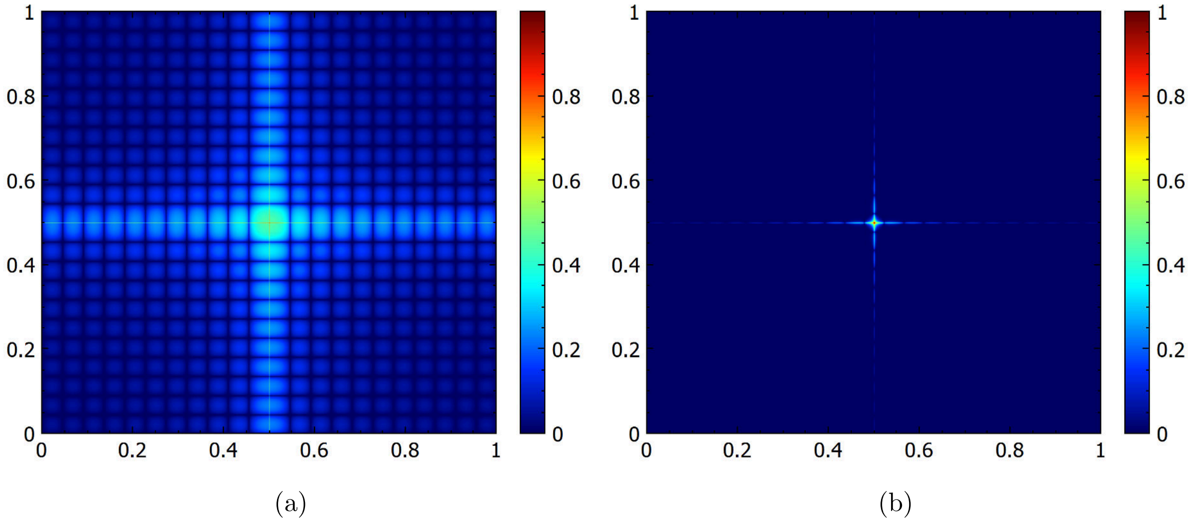


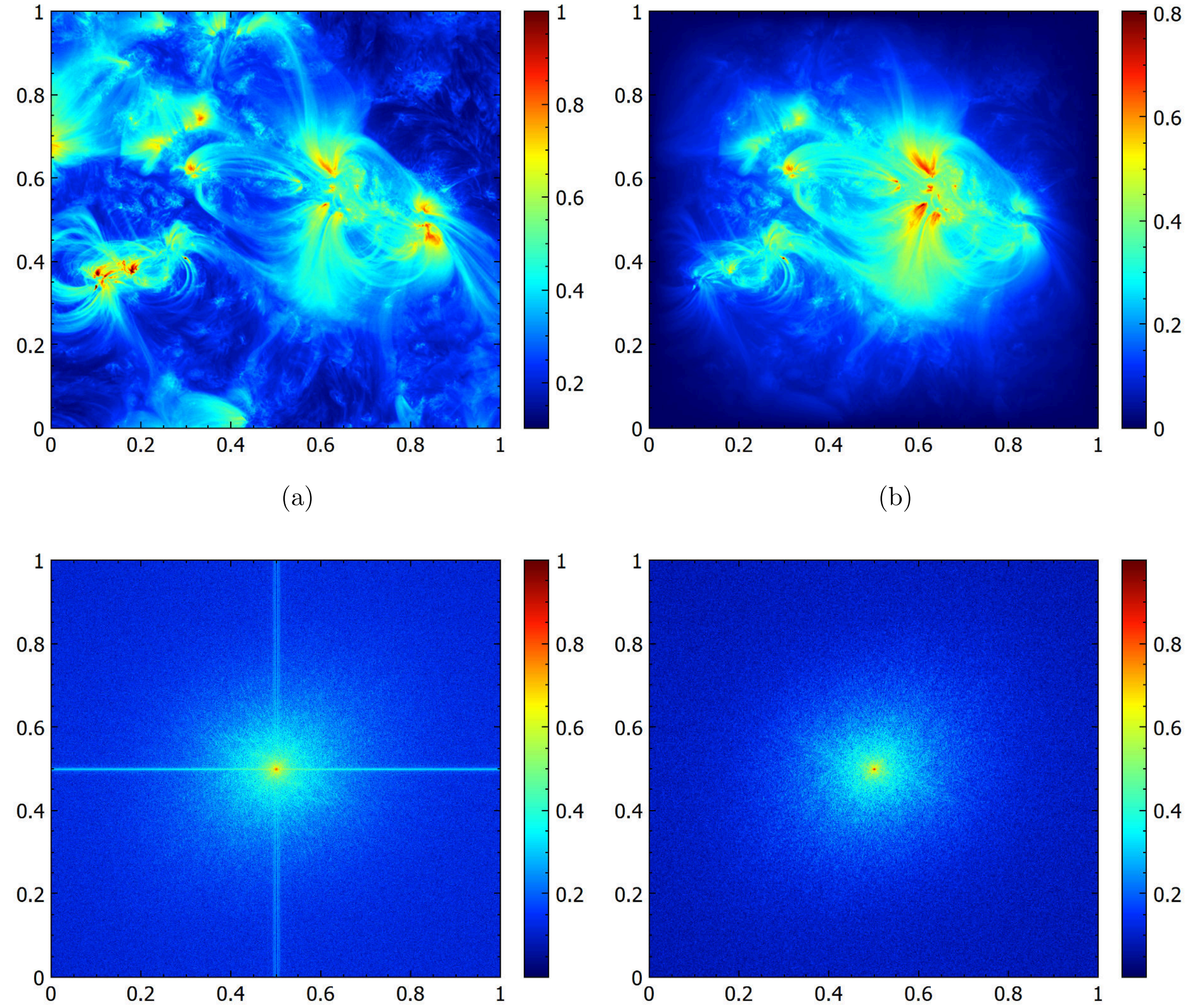
Figure 3.3: The log power spectrum ($\log |\mathcal{F}\{W(x)\}|$) of a rectangular window (blue) and a Hann window (orange).



(a)

(b)

Figure 3.4: The two-dimensional log power spectrum ($\log |\mathcal{F}\{W(x, y)\}|$) of a rectangular window (a) and a Hann window (b). Note the clearly visible effects of spectral leakage in (a).



(a)

(b)

(c)

(d)

Figure 3.5: The effect of image windowing on the log power spectrum - image $I_1(x, y)$ without an applied window (with an implicit rectangular window) (a), image $I_2(x, y)$ with a Hann window applied (b), the log power spectrum $I_1(x, y)$ (c), the log power spectrum of $I_2(x, y)$ (d). Note the dramatic suppression of the spectral leakage effect from (c) to (d).

Gaussian window, the Blackman window, the Hann window and the Hamming window. The case of not applying any window prior to computing the two image DFTs is equivalent to applying a simple rectangular window.

All window functions (including the rectangular window) introduce a compromise between the width of the main lobe and the attenuation of the side lobes in the frequency domain. The rectangular window function results in the sharpest main lobe of all windows, but has poor attenuation of the side lobes. In contrast, the Blackman window function results in a wider main lobe, but with significantly better side lobe attenuation. According multiple performed image registration experiments, the Hann window seems like a good compromise and often yields the best results of all the mentioned windows, when used with the IPC method.

3.2 Cross-power spectrum filtering

The first substantial deviation of the IPC method from the standard phase correlation algorithm is the addition of adjustable cross-power spectrum filtering. The filtering is applied to the normalized cross power spectrum $P(x, y)$ (see Eq. 1.19) of the input images. The motivation behind the use of a cross-power spectrum filter is twofold. Firstly, as is usual in signal processing, higher spatial frequencies often mostly contain the information about noise, and thus should almost always be attenuated. Secondly, lower spatial frequencies mostly contain the information about the overall brightness of the image, which should not be important to the image registration algorithm. Furthermore, the overall brightness of the two input images could differ not only due to the image shift itself, but also due to some external factors, such as different image calibration or different exposure times. These external factors should not play a big role in the resulting image shift estimate, and thus the lower spatial frequencies should also be attenuated.

The ideal cross-power spectrum filter for the IPC algorithm is then a band-pass filter $B(x, y)$ with adjustable parameters affecting the amount of attenuation of both lower and higher spatial frequencies. The band-pass filter can be implemented with sharp (rectangular) or smooth (Gaussian) transitions. The sharp rectangular adjustable band-pass filter $B_R(x, y)$ (see Fig. 3.6 and 3.7 (a)) can be defined as

$$B_R(x,y) = \begin{cases} 1, & \text{if } f_L \leq \sqrt{\frac{\left(x-\frac{w}{2}\right)^2}{2\left(\frac{w}{2}\right)^2} + \frac{\left(y-\frac{h}{2}\right)^2}{2\left(\frac{h}{2}\right)^2}} \leq f_H \vee f_L \geq f_H \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

where w and h denote the width and height of the image and $f_L, f_H \in [0, 1]$ denote the lower and upper normalized spatial frequency thresholds, satisfying the inequality

$$0 \leq f_L < f_H \leq 1.$$

Similarly to the rectangular window, the rectangular band-pass filter is very simple and quick to compute, but does not usually perform well (see Fig. 3.9), again due to the convolution theorem property (this time the inverse version of it). Multiplying the normalized cross-power spectrum point-wise with a band-pass function $B(x, y)$ is equivalent to performing a convolution of the resulting phase correlation landscape $p(x, y)$ with the inverse DFT of $B(x, y)$, namely

$$\mathcal{F}^{-1}\left\{P\left(x,y\right) \odot B\left(x,y\right)\right\} = p\left(x,y\right) * \mathcal{F}^{-1}\left\{B(x,y)\right\}. \tag{3.3}$$

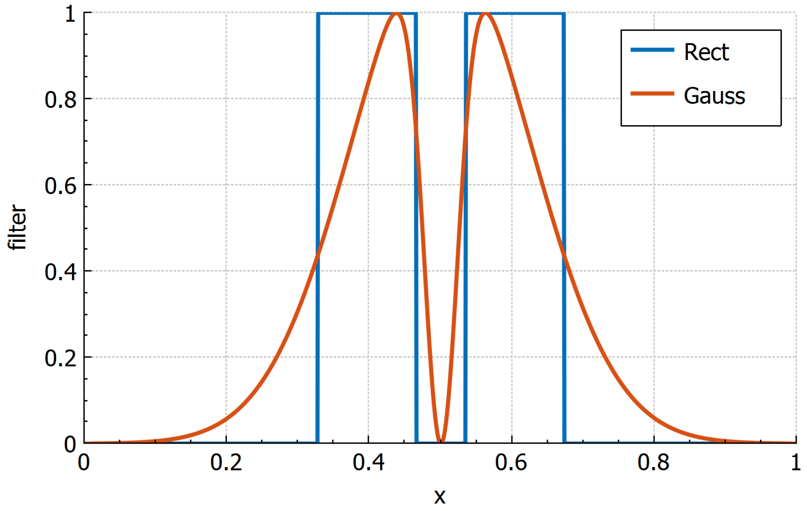


Figure 3.6: Frequency domain band-pass filter functions - a rectangular band-pass filter ($f_L = 0.05,\ f_H = 0.25$) and a Gaussian band-pass filter ($\sigma_L = 0.05,\ \sigma_H = 0.25$).

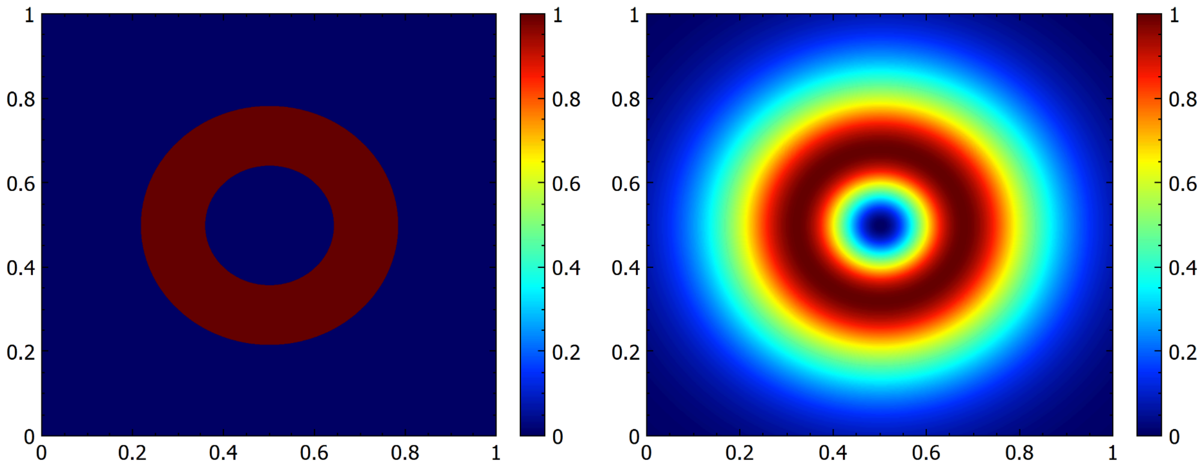
Thus, a good band-pass filter should have an inverse DFT, that has the majority of power concentrated in a single lobe at the origin, so that it does not "corrupt" the resulting phase correlation landscape (see Fig. 3.8).

The smooth version of the adjustable band-pass filter - the Gaussian band-pass filter $B_G(x, y)$ (see Fig. 3.6 and 3.7 (b)) can be constructed by combining multiple Gaussians as

$$B_G(x,y) = \begin{cases} \exp\left(-\frac{1}{2\sigma_H^2}\left(\frac{\left(x-\frac{w}{2}\right)^2}{\left(\frac{w}{2}\right)^2} + \frac{\left(y-\frac{h}{2}\right)^2}{\left(\frac{h}{2}\right)^2}\right)\right), & \text{for } \sigma_L \leq 0 \land 0 < \sigma_H < 1 \\ 1 - \exp\left(-\frac{1}{2\sigma_L^2}\left(\frac{\left(x-\frac{w}{2}\right)^2}{\left(\frac{w}{2}\right)^2} + \frac{\left(y-\frac{h}{2}\right)^2}{\left(\frac{h}{2}\right)^2}\right)\right), & \text{for } \sigma_L > 0 \land \sigma_H \geq 1 \\ \exp\left(-\frac{1}{2\sigma_H^2}\left(\frac{\left(x-\frac{w}{2}\right)^2}{\left(\frac{w}{2}\right)^2} + \frac{\left(y-\frac{h}{2}\right)^2}{\left(\frac{h}{2}\right)^2}\right)\right) & \\ \times\left(1 - \exp\left(-\frac{1}{2\sigma_L^2}\left(\frac{\left(x-\frac{w}{2}\right)^2}{\left(\frac{w}{2}\right)^2} + \frac{\left(y-\frac{h}{2}\right)^2}{\left(\frac{h}{2}\right)^2}\right)\right)\right), & \text{for } \sigma_L > 0 \land 0 < \sigma_H < 1 \\ 1, & \text{otherwise} \end{cases} \tag{3.4}$$

where $\sigma_L, \sigma_H \in \mathbb{R}$ denote the adjustable parameters controlling the widths of the low pass and high pass Gaussians, which in turn affect the amount of low frequency and high frequency attenuation. The band-pass filter values are always normalized to the $[0, 1]$ value range. With decreasing σ_H, the low pass Gaussian shrinks, and thus more high frequency attenuation is applied to the cross power spectrum, while with increasing σ_L, the high pass Gaussian widens, and thus more low frequency attenuation is applied to the cross power spectrum.

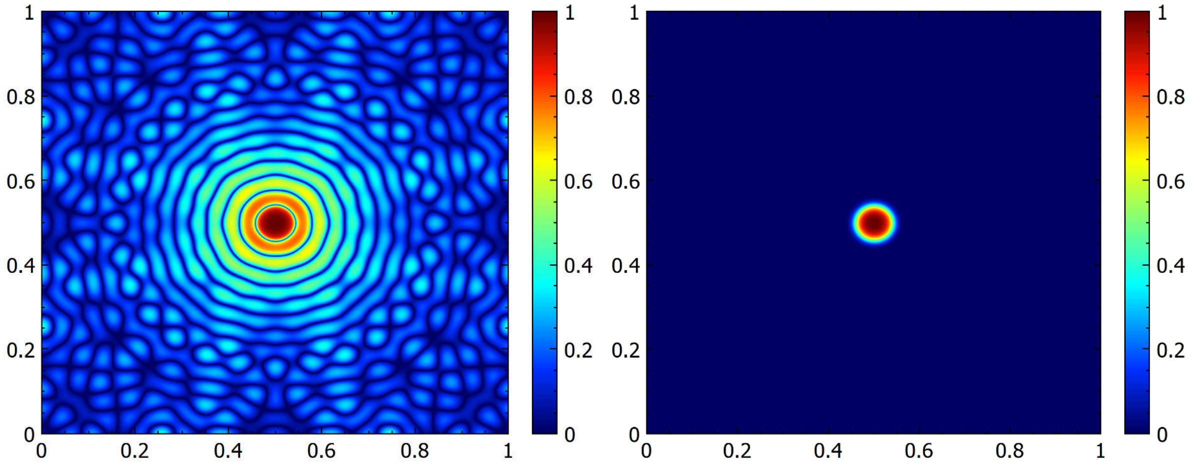
The filtering process becomes very easy to control with rectangular and Gaussian band-pass filters $B_R(x, y)$ and $B_G(x, y)$, such as those defined in Equations 3.2 and 3.4. For $f_L = 0$ or $\sigma_L \leq 0$, no low frequency attenuation is performed. Similarly, for $f_H = 0$ or $\sigma_H \geq 1$, no high frequency attenuation is performed. The entire filtering process can be skipped with $f_L = 0 \land f_H = 1$ or $\sigma_L \leq 0 \land \sigma_H \geq 1$. In general, f_L, f_H or σ_L, σ_H parameters precisely and uniquely control the amounts of high and low spatial frequency



(a)

(b)

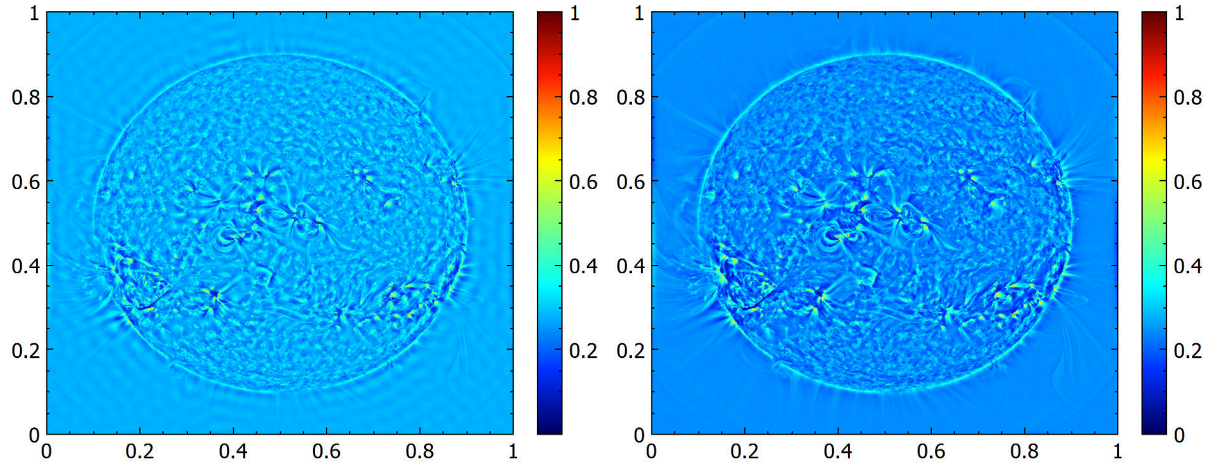
Figure 3.7: Two-dimensional cross-power spectrum filter functions: rectangular band-pass filter ($f_L = 0.2$, $f_H = 0.4$) (a), Gaussian band-pass filter ($\sigma_L = 0.2$, $\sigma_H = 0.4$) (b).



(a)

(b)

Figure 3.8: Examples of contrast-enhanced inverse Fourier transforms of a rectangular band-pass filter (a) and a Gaussian band-pass filter (b) with equivalent parameters ($f_L = \sigma_L, f_H = \sigma_H$).



(a)

(b)

Figure 3.9: An example of a band-pass filtered image using a rectangular filter (a) and a Gaussian filter (b) with equivalent parameters ($f_L = \sigma_L, f_H = \sigma_H$).

attenuation of the corresponding normalized cross power spectrum filter type.

3.3 Correlation sub-regions and upsampling

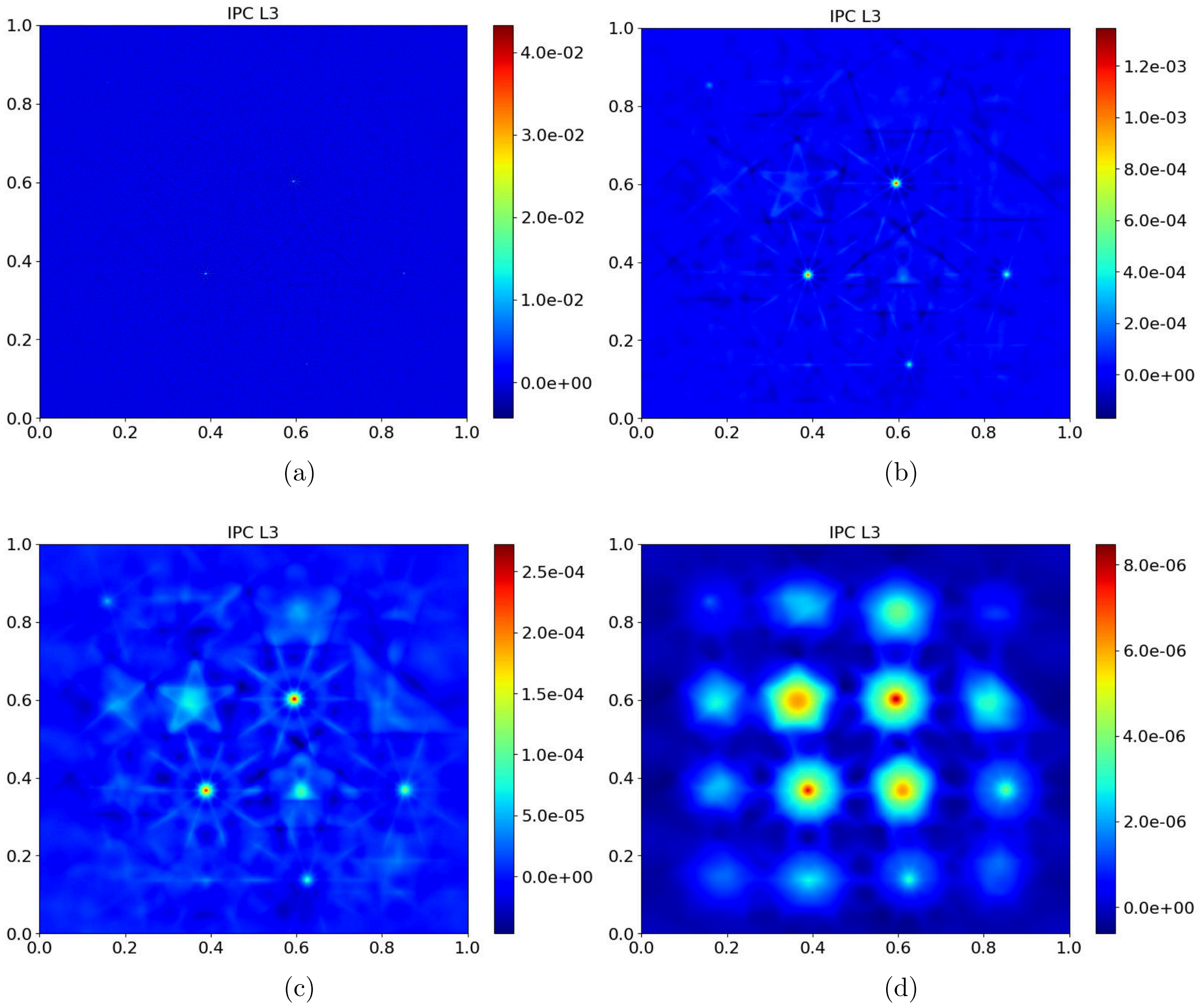
3.3.1 The L^3 region

After the inverse DFT of the band-pass filtered normalized cross power spectrum of the two windowed input images I_1 and I_2 is computed, the correlation function

$$L^3 = \mathcal{F}^{-1}\left\{ B \odot \frac{\mathcal{F}\left\{I_1 \odot W\right\} \odot \overline{\mathcal{F}\left\{I_2 \odot W\right\}}}{\left|\mathcal{F}\left\{I_1 \odot W\right\} \odot \overline{\mathcal{F}\left\{I_2 \odot W\right\}}\right| + \xi} \right\}, \tag{3.5}$$

denoted as L^3, is obtained.[1] The additional parameter ξ in the normalized cross-power spectrum denominator effectively prevents division by zero errors. Additionally, with larger values of ξ, the resulting correlation landscape becomes gradually more and more similar to the standard cross-correlation landscape (see Fig. 3.10).

[1] The L^n naming convention of the phase correlation sub-regions was inspired by the commonly used types of computer processor unit (CPU) memory caches, namely the L3/L2/L1 cache hierarchy [Smith, 1982]. The purpose of these processor caches is to hold copies of data from the main memory, which are likely to be accessed by the CPU next, directly in the CPU cores themselves. The presence of these smaller memory caches close to the CPU cores increases the average execution speed and power efficiency, since data do not need to be accessed from the main memory as frequently. While the L3 cache has the largest memory capacity, it is also the slowest. The L2 and L1 caches are even closer to the CPU cores and therefore are significantly faster, but usually hold proportionally less data. Analogically to the main computer memory, the phase correlation landscape L^3 holds all the raw correlation data, while the L^2, L^{2U} and L^1 sub-regions (described in 3.3.3, 3.3.4) represent significantly smaller portions of L^3 localized near the point of maximum phase correlation, and are accessed frequently during the iterative refinement process (see 3.4).



(a)

(b)

(c)

(d)

Figure 3.10: Examples of the L^3 region obtained with Eq. 3.5 for $\xi = 0$ (a), $\xi = 1$ (b), $\xi = 10$ (c), $\xi = 1000$ (d).

3.3.2 The L^2 sub-region

Since upsampling[2] the entire correlation landscape L^3 is unnecessary and computationally expensive, a sub-region of L^3, centered around the point of maximal correlation $L^3_{\max}$, is extracted (denoted as L^2). The size of this sub-region (denoted $|L^2|$) is a parameter and can be modified by the user. However, since the sub-region should be centered around a specific point, odd sub-region sizes are preferred, so that the specified point can correspond precisely to the central pixel.

3.3.3 The L^{2U} sub-region

The extracted correlation sub-region L^2 is upsampled with a two-dimensional interpolation method, yielding an upsampled version of L^2, denoted as L^{2U}. The type of the interpolation method and the size of L^{2U} (or equivalently, the correlation upsampling factor U) are additional adjustable parameters, which can be modified according to the characteristics of the input images. Most commonly used two-dimensional interpolation methods include the nearest neighbor interpolation, bilinear interpolation and bicubic interpolation.

3.3.4 The L^1 sub-region

Finally, another correlation sub-region centered around the maximal correlation point is extracted, this time from the already upsampled sub-region L^{2U}. The size of this sub-region is another modifiable parameter, however, specifying it as a ratio of $\left|L^{2U}\right|$ adds more clarity to the meaning of this parameter. With the L^1 size ratio denoted as L^1_r and satisfying $0 < L^1_r < 1$, the size of the final extracted L^1 sub-region becomes

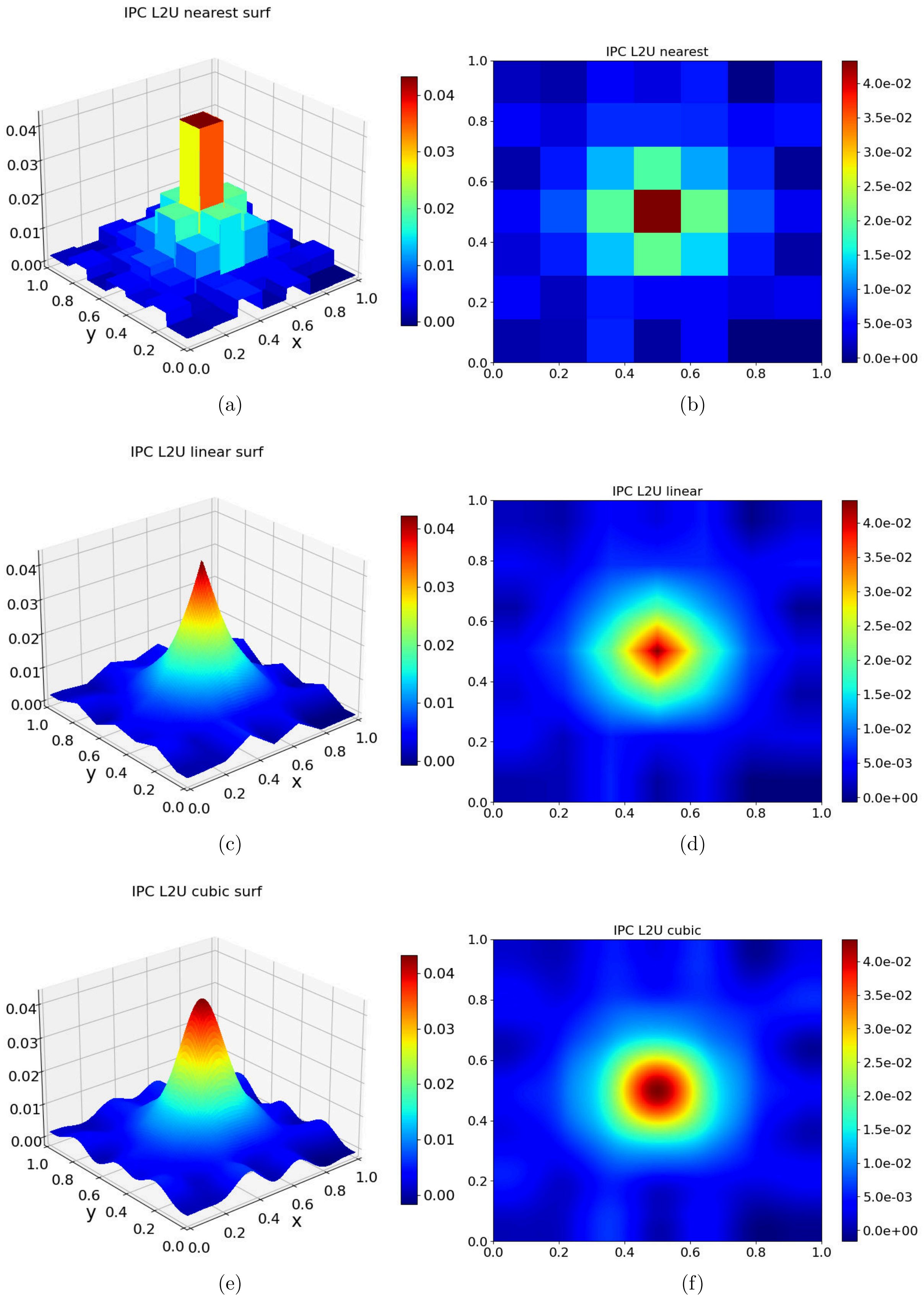
$$\left|L^1\right| = \begin{cases} \max\left(\left\lfloor L^1_r \cdot \left|L^{2U}\right|\right\rfloor, 3\right), & \text{if } \left\lfloor L^1_r \cdot \left|L^{2U}\right|\right\rfloor \text{ odd} \\ \max\left(\left\lfloor L^1_r \cdot \left|L^{2U}\right|\right\rfloor + 1, 3\right), & \text{otherwise} \end{cases} \tag{3.6}$$

where the degenerate cases of too small sub-region sizes are solved simply by clamping the result from the bottom by introducing a maximum function with the value of 3, which is the lowest pixel size, that is odd and can still yield sub-pixel improvements during the iterative refinement process.

The L^1 sub-region represents the closest neighborhood of the maximum phase correlation point $L^3_{\max}$ of the original phase correlation landscape L^3, and will be the only region that will be updated in each iteration during the iterative refinement process. The purpose of this region is to represent the phase correlation peak as closely as possible, all the while omitting all other irrelevant correlation variations from the rest of L^3. Since the horizontal cross sections of the phase correlation peak are always of a circular shape, filtering of the correlation values that do not contribute to the main correlation peak can be done by multiplying the entire L^1 region point-wise with a binary circular mask M, defined as

$$M(x, y) = \begin{cases} 1, & \text{if } \left(x - \frac{\left|L^1\right|}{2}\right)^2 + \left(y - \frac{\left|L^1\right|}{2}\right)^2 \leq \left(\frac{\left|L^1\right|}{2}\right)^2 \\ 0, & \text{otherwise.} \end{cases} \tag{3.7}$$

[2] Here, upsampling is meant as increasing the sampling rate (increasing the resolution), which is usually done by inserting zeros (zero-padding), or by an appropriate interpolation method.



(a)

(b)

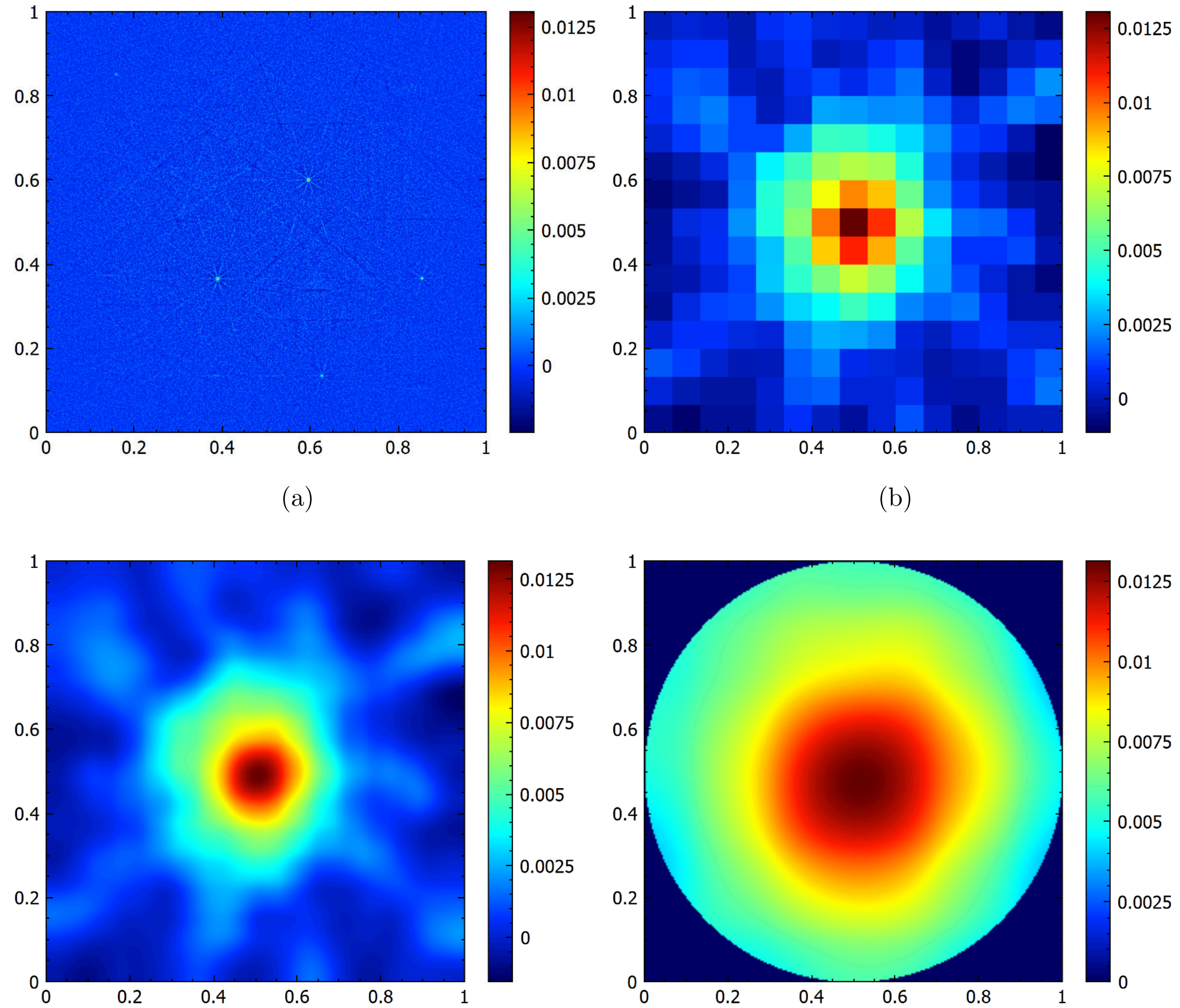
(c)

(d)

(e)

(f)

Figure 3.11: Examples of the L^{2U} sub-region obtained with nearest neighbor (a)/(b), linear (c)/(d), and cubic (e)/(f) interpolation.



(a)

(b)

(c)

(d)

Figure 3.12: An example of the IPC sub-region hierarchy - the L^3 region (a), the corresponding L^2 sub-region (b), the corresponding L^{2U} sub-region (c), the corresponding L^1 sub-region (d).

Many other sub-pixel extensions of the phase correlation method work mainly with a rectangular neighborhood (e.g. 5×5) around the point of maximum phase correlation, which introduces substantial amounts of irrelevant correlation values, especially near the corners of the rectangular region. These irrelevant correlation values can often substantially skew the estimate of the main peak center towards incorrect values and thus degrade the resulting sub-pixel image shift estimation accuracy.

3.4 Iterative refinement

After the L^3, L^2, L^{2U} and L^1 regions are computed, the process of iterative sub-pixel refinement is started. The aim of this process is to accurately estimate the sub-pixel part of the image shift and thus improve the final registration accuracy. Since zero image shift corresponds to the central pixel of L^3, the pixel level accuracy image shift estimate $[\Delta x, \Delta y]$ can be computed simply by

$$[\Delta x, \Delta y] = L^3_{\text{max}}\left[x, y\right] - L^3_{\text{mid}}\left[x, y\right], \tag{3.8}$$

where $L^3_{\text{max}}\left[x, y\right]$ is the point of maximum phase correlation and $L^3_{\text{mid}}\left[x, y\right]$ is the central pixel of L^3. This is the standard phase correlation result, to which the final sub-pixel image shift estimate obtained by the iterative refinement will be added.

In each iterative refinement iteration (denoted with index i), firstly, the current circular upsampled sub-region $L^1\left(i\right)$ is computed as a sub-region of L^{2U}, centered around the current upsampled maximum phase correlation location $L^{2U}_{\text{max}}\left(i\right)\left[x, y\right]$. Secondly, the sub-pixel centroid location $L^1_{\text{cen}}\left(i\right)\left[x, y\right]$ of the current circular upsampled sub-region $L^1\left(i\right)$ is calculated as

$$L^1_{\text{cen}}\left[x, y\right] = \left[\frac{\sum_x \sum_y x L^1(x, y)}{\sum_x \sum_y L^1(x, y)}, \frac{\sum_x \sum_y y L^1(x, y)}{\sum_x \sum_y L^1(x, y)}\right]. \tag{3.9}$$

Thirdly, the current upsampled maximum phase correlation location $L^{2U}_{\text{max}}\left(i\right)\left[x, y\right]$ in the L^{2U} sub-region is calculated by adding the integer part of the current relative $L^1\left(i\right)$ centroid location (relative to the L^1 origin $L^1_{\text{mid}}\left[x, y\right]$) to its value from the previous iteration as

$$L^{2U}_{\text{max}}\left(i\right)\left[x, y\right] = L^{2U}_{\text{max}}\left(i-1\right)\left[x, y\right] + \left\lfloor L^1_{\text{cen}}\left(i\right)\left[x, y\right] - L^1_{\text{mid}}\left[x, y\right]\right\rfloor. \tag{3.10}$$

The entire iterative centroid refinement process is concisely summarized by Algorithm 1.

This iterative process is repeated until the circular upsampled sub-region centroid location is less than 0.5 pixels away from the origin in both x and y directions, or if the maximum number of iterations (denoted N_{max}) is reached. The threshold distance of 0.5 pixels reflects the fact, that the L^1 sub-region can be moved only by an integer number of (upsampled) pixels in each iteration. Therefore, if this distance is less than 0.5 pixels in both directions, the algorithm converged and the iterative refinement process is terminated.

On the other hand, if more than N_{max} refinement iterations are computed and the current centroid distance is still larger than the threshold distance, the algorithm diverged. In this case, reducing the size of L^1 and running the entire iterative refinement process again (which can be done automatically) can yield better results. This is caused by the fact, that smaller peak neighborhoods are likely to contain lower amounts of correlation

Algorithm 1: Iterative sub-pixel centroid refinement process

```
- calculate the $L^2$ sub-region of $L^3$ centered around $L^3_{\text{max}}$;
- calculate the $L^{2U}$ upsampled version of $L^2$;
while $|L^1| \geq 3$ do
    - reset $L^{2U}_{\text{max}}$ to $L^{2U}_{\text{mid}}$;
    - calculate the current $|L^1|$ (Eq. 3.6);
    for $i = 0$ to $N_{max}$ do
        - calculate the current $L^1$ centered around the previous $L^{2U}_{\text{max}}$;
        - calculate the current $L^1_{\text{cen}}[x, y]$ (Eq. 3.9);
        - calculate the current $L^{2U}_{\text{max}}[x, y]$ (Eq. 3.10);
        if $\left(L^1_{cen}[x, y] - L^1_{mid}[x, y]\right) < [0.5, 0.5]$ then
            return $[\Delta x_r, \Delta y_r]$ (Eq. 3.12);
        end if
    end for
    - reduce $|L^1|$ by reducing $L^1_r$ (Eq. 3.6)
end while
Result: Refined sub-pixel image shift estimate $[\Delta x_r, \Delta y_r]$
```

variation, and thus converge faster. If the refinement process keeps diverging, the size of L^1 can keep reducing down to the theoretical minimum of 3 pixels.

Divergence of the iterative refinement process happens rarely and is caused mainly by very low signal to noise rations of the two input images, or by very small image sizes. Furthermore, if the iterative refinement process diverges, it is highly likely that the entire phase correlation landscape contains no single distinct maximum correlation peak, and thus no accurate estimate can be produced from the correlation data by any method.

The iterative sub-pixel centroid refinement process is highly likely to converge to a high precision sub-pixel estimate of the relative shift between the two input images. Intuitively, the process simply moves the L^1 circular sub-region in the direction of higher correlation, allowing it to selectively explore the L^{2U} upsampled correlation landscape around the original $L^{2U}_{\text{max}}[x, y]$ maximum correlation point, and thus converge to a more accurate solution, as opposed to a single centroid calculation with neighborhood centered around the maximum correlation value in the case of non-iterative methods.

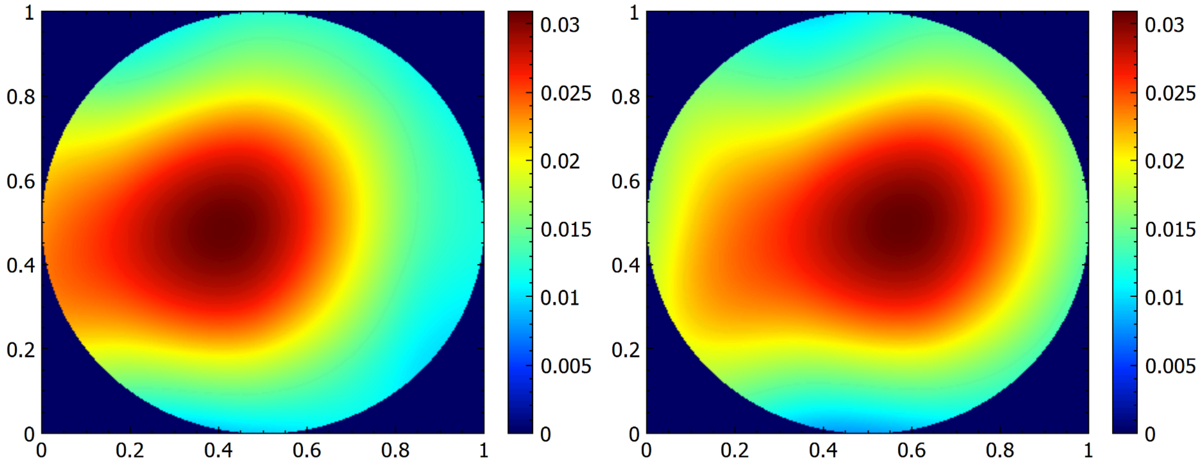
If the final number of iterations, after which the iterative centroid refinement process converged, is denoted by n (satisfying $n \leq N_{\text{max}}$), the final refined sub-pixel image shift estimate $[\Delta x_r, \Delta y_r]$ can be calculated by

$$[\Delta x_r, \Delta y_r] = [\Delta x, \Delta y] + \frac{1}{U}\left(\sum_{i=0}^{n-1}\left(\left\lfloor L^1_{\text{cen}}(i)[x, y] - L^1_{\text{mid}}[x, y]\right\rfloor\right) + L^1_{\text{cen}}(n)[x, y] - L^1_{\text{mid}}[x, y]\right), \quad (3.11)$$

or in other terms, using the $L^3/L^{2U}/L^1$ sub-region hierarchy,

$$[\Delta x_r, \Delta y_r] = L^3_{\text{max}}[x, y] - L^3_{\text{mid}}[x, y] + \frac{1}{U}\left(L^{2U}_{\text{max}}(n)[x, y] - L^{2U}_{\text{mid}}[x, y] + L^1_{\text{cen}}(n)[x, y] - L^1_{\text{mid}}[x, y]\right), \quad (3.12)$$

where the term $L^3_{\text{max}}[x, y] - L^3_{\text{mid}}[x, y]$ corresponds to the raw pixel level shift estimate, the term $L^{2U}_{\text{max}}(n)[x, y] - L^{2U}_{\text{mid}}[x, y]$ corresponds to the integer part of the upsampled sub-pixel



(a) (b)

Figure 3.13: An example of the L^1 sub-region before (a) and after (b) the iterative refinement process described in Algorithm 1.

centroid shift estimate (in the last refinement iteration n) and $L^1_{\text{cen}}(n)[x,y] - L^1_{\text{mid}}[x,y]$ corresponds to the sub-pixel part of the upsampled sub-pixel centroid shift estimate in the last refinement iteration (this is the remaining upsampled sub-pixel centroid shift, which is now smaller than 0.5 pixels in both directions). Furthermore, sub-pixel contributions from the upsampled correlation sub-regions L^{2U} and L^1 need to be divided by the correlation upsampling ratio $U = |L^{2U}|/|L^2|$ to be equal in magnitude to the raw pixel-level shift estimate from L^3, to which they are added.

3.5 Rotation and scale estimation

In the general IPC image alignment procedure, estimates of the rotation and scale parameters φ and s are obtained by exploiting three important Fourier transform properties, namely the Fourier scale theorem (see Eq. 1.5), the fact that translation in the image domain does not affect the frequency domain magnitude spectrum (Fourier shift theorem - Eq. 1.4) and the fact that rotation in the image domain corresponds to equivalent rotation in the frequency domain (Fourier rotation theorem - Eq. 1.6). The task of rotation and scale estimation can thus be transformed into translation estimation between the log-polar transformed magnitude spectra of Fourier transforms of both images.

To significantly improve the robustness of this alignment step, two additional operations are performed. Firstly, a window (see Section 3.1) is applied to both images prior to the Fourier transform magnitude spectrum computation to reduce spectral leakage effects. Secondly, since most of the frequency power of the majority of real-world images is concentrated near the origin (at lowest frequencies), a logarithm function is applied to both magnitude spectra to increase contrast for the subsequent shift estimation.

Once the rotation and scale estimates are obtained, the second input image is rotated and scaled back by the estimated amounts. Afterwards, the sub-pixel shift between the first image and the rotated and scaled second image is calculated. The second image is then shifted back by the computed amount, completing the alignment. Both affine transformations (rotation/scale and translation) can be computed with sub-pixel precision

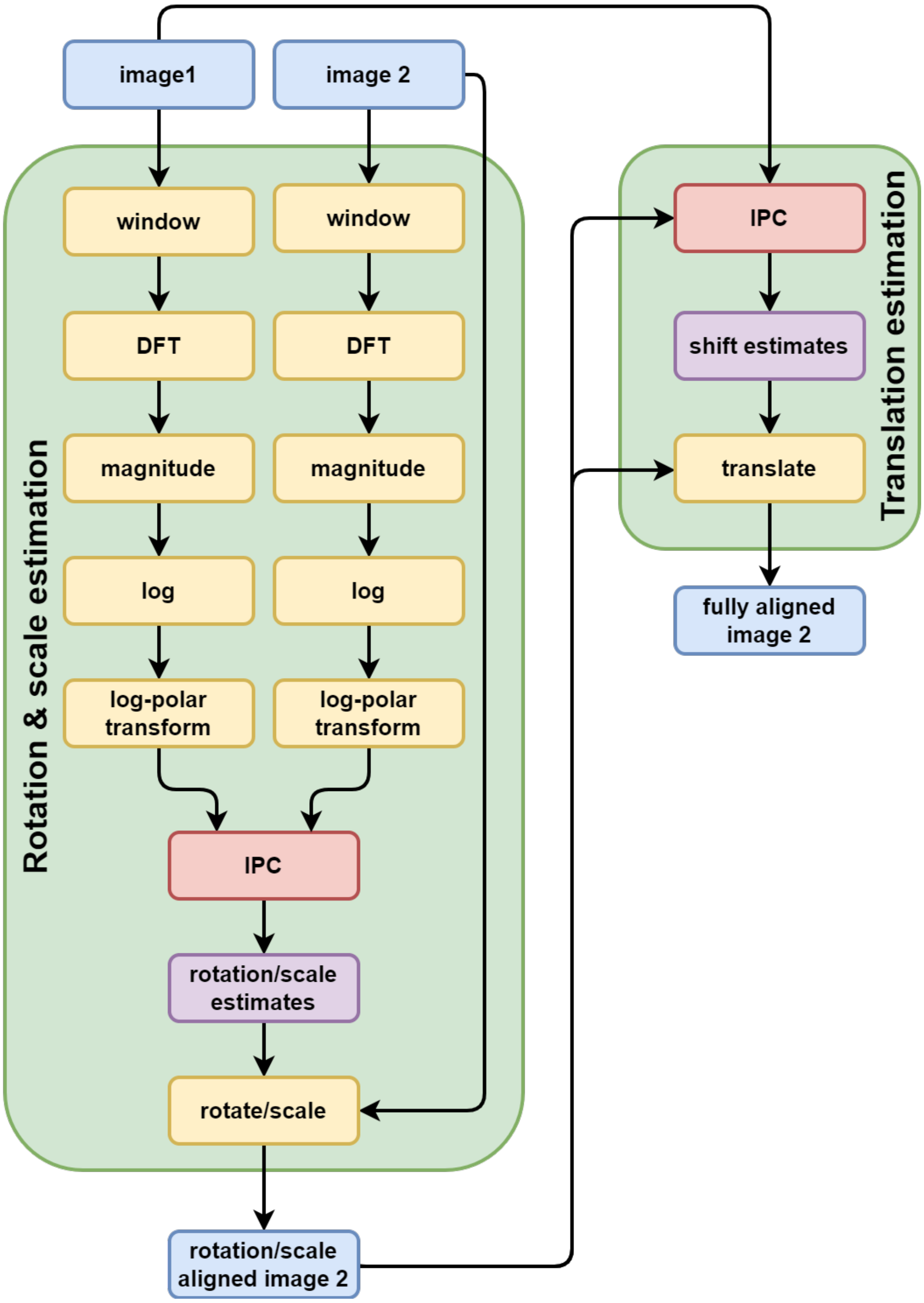
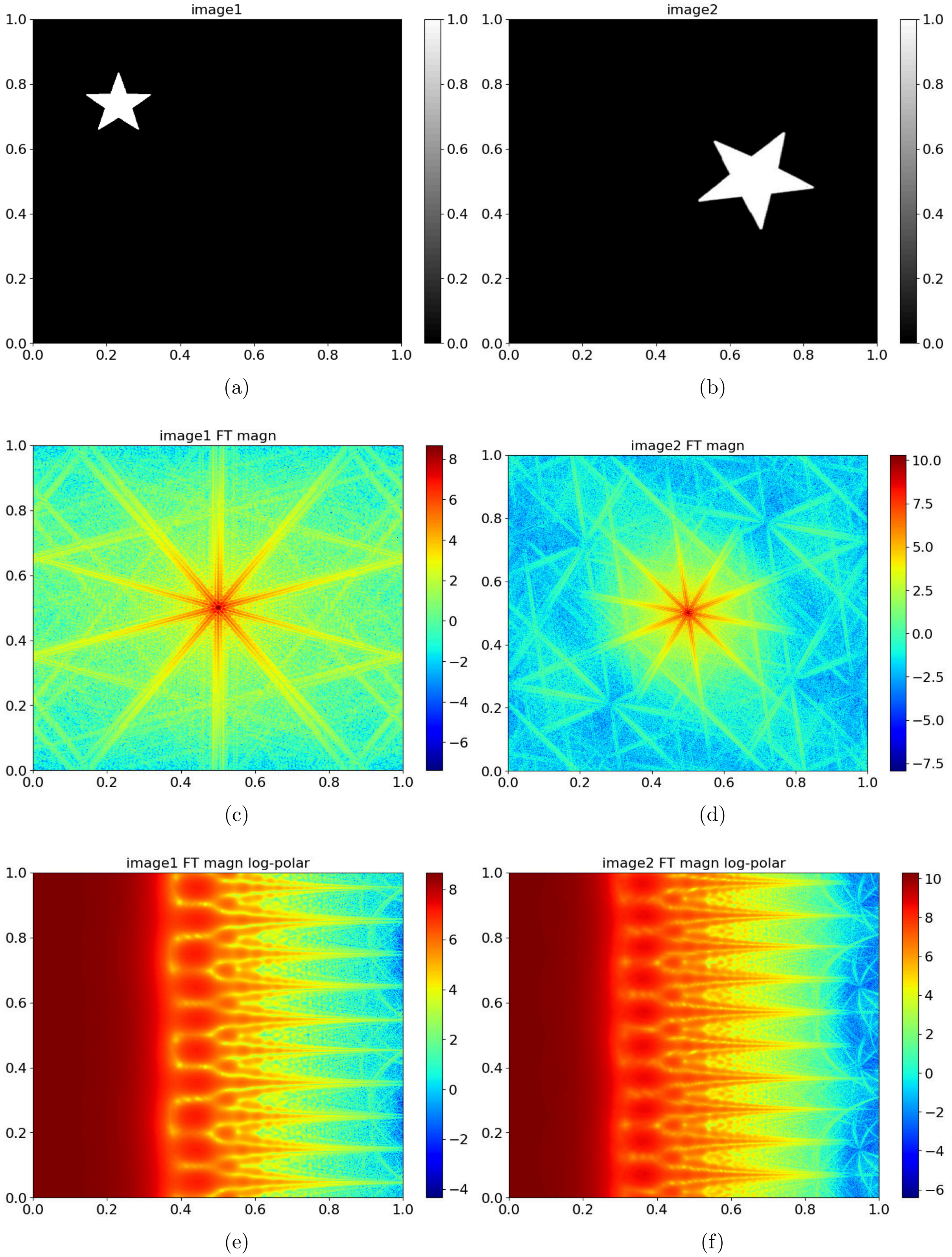


Figure 3.14: A flowchart of the general IPC image alignment process. In this scheme, the second input image is aligned to the first input image. Examples of the frequency log-magnitude and log-polar transform steps can be seen in Fig. 3.15.



(a) (b) (c) (d) (e) (f)

Figure 3.15: An example of the general IPC image alignment method (as shown in Fig. 3.14) on artificially rotated, scaled and shifted images. First input image (a), second input image (rotated, scaled and shifted image 1) (b), log-magnitude frequency spectrum of image 1 (c), log-magnitude frequency spectrum of image 2 (d), log-polar transformed log-magnitude frequency spectrum of image 1 (e), log-polar transformed log-magnitude frequency spectrum of image 2 (f).

via bilinear interpolation. A flowchart describing the general IPC image alignment process can be seen in Fig. 3.14. An example of the log-magnitude frequency spectrum and log-polar transformed log-magnitude frequency spectrum steps of the general IPC image alignment process on artificially rotated, scaled and shifted images can be seen in Fig. 3.15.

3.6 False correlations removal

As discussed previously in Section 2.2, intensity-based image registration algorithms are best suited for registering image pairs with globally uniform motion, rotation and scale. However, for some applications, the input image pairs often contain “almost” uniform motion - meaning that some (usually small) regions in the image are stationary and/or move in a different direction relative to the rest of the image. Standard intensity-based algorithms usually fail to correctly register such image pairs, because these non-uniformly moving regions give rise to “false” peaks in the resulting correlation function. If any one of these false peaks is the most prominent peak (meaning its correlation value is the highest), the image registration process fails. The relative height of each correlation peak depends on the relative area of the corresponding region in the image and on the signal-to-noise ratio (SNR). Regions with good SNR give rise to more prominent peaks when compared to similarly-sized regions with poor SNR.

A typical example of image pairs with “almost” uniform motion are images obtained by defective sensors and/or optics. Such images can contain multiple stable artifacts caused by dust in the optical system or dead/stuck/hot pixels. These artifacts are usually extremely similar (or completely equal in the case of dead/stuck/hot pixels) across many images. This very high similarity (and thus, high SNR) can give rise to prominent correlation peaks located exactly at the origin.

Another good example of an image pair with “almost” uniform motion can be seen in Fig. 3.16. The figure shows two contrast-enhanced solar eclipse images, taken about a minute apart. The black circular region moving towards the bottom left of the image is the Moon, while the rest of the image mainly consists of solar coronal features and the solar wind. Compared to the short time difference between the images, the coronal features are completely static. When registering such image pairs, the correlation peak associated with the movement of the Moon will be completely overshadowed by a very prominent peak at the origin, caused by the static coronal features.

This problem can be overcome by “masking” of the L^3 region origin (see Section 3.3) of the IPC algorithm. The masking step can be done by multiplying L^3 region element-wise by an inverse circular mask, effectively setting all correlations in a given radius around the origin to zero. An example of this L^3 origin masking procedure on the contrast-enhanced solar eclipse images from Fig. 3.16 can be seen in Fig. 3.17 (showing the difference between the resulting L^{2U} regions).

Furthermore, the false correlations removal procedure can be extended to arbitrary correlation regions, not only the origin, as is the case in Fig. 3.17. For example, with input images from Fig. 3.16, a reverse scenario is also possible - a specific application might aim to measure small movements of some coronal structures, disregarding the motion of the Moon. The movement of the Moon can be precisely measured by L^3 region origin masking, and the correlation area around that measured value can then be “masked out” as well, making such measurement possible.

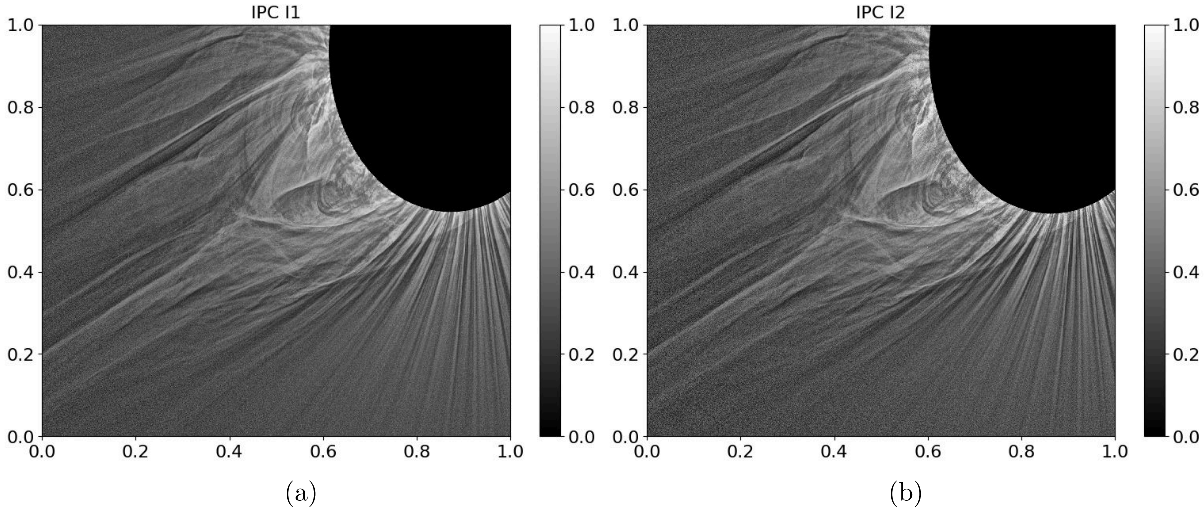


Figure 3.16: Two contrast-enhanced solar eclipse images taken about one minute apart - an example of an image pair with "almost" uniform motion.

3.7 Parameter optimization

The accuracy and robustness of the IPC method is affected by multiple parameters. In some of the following sections, sensible default values of all the parameters are given and explained. These default values have been tested and work well for a wide range of image types and sizes. However, if the registration estimates are to be obtained with absolute maximum sub-pixel accuracy, the IPC parameters shall be adapted (optimized) for a particular image type. Optimal IPC parameters mainly depend on the size of the image, on the image signal-to-noise ratio and on the spatial frequency characteristics of the objects and structures contained within the image.

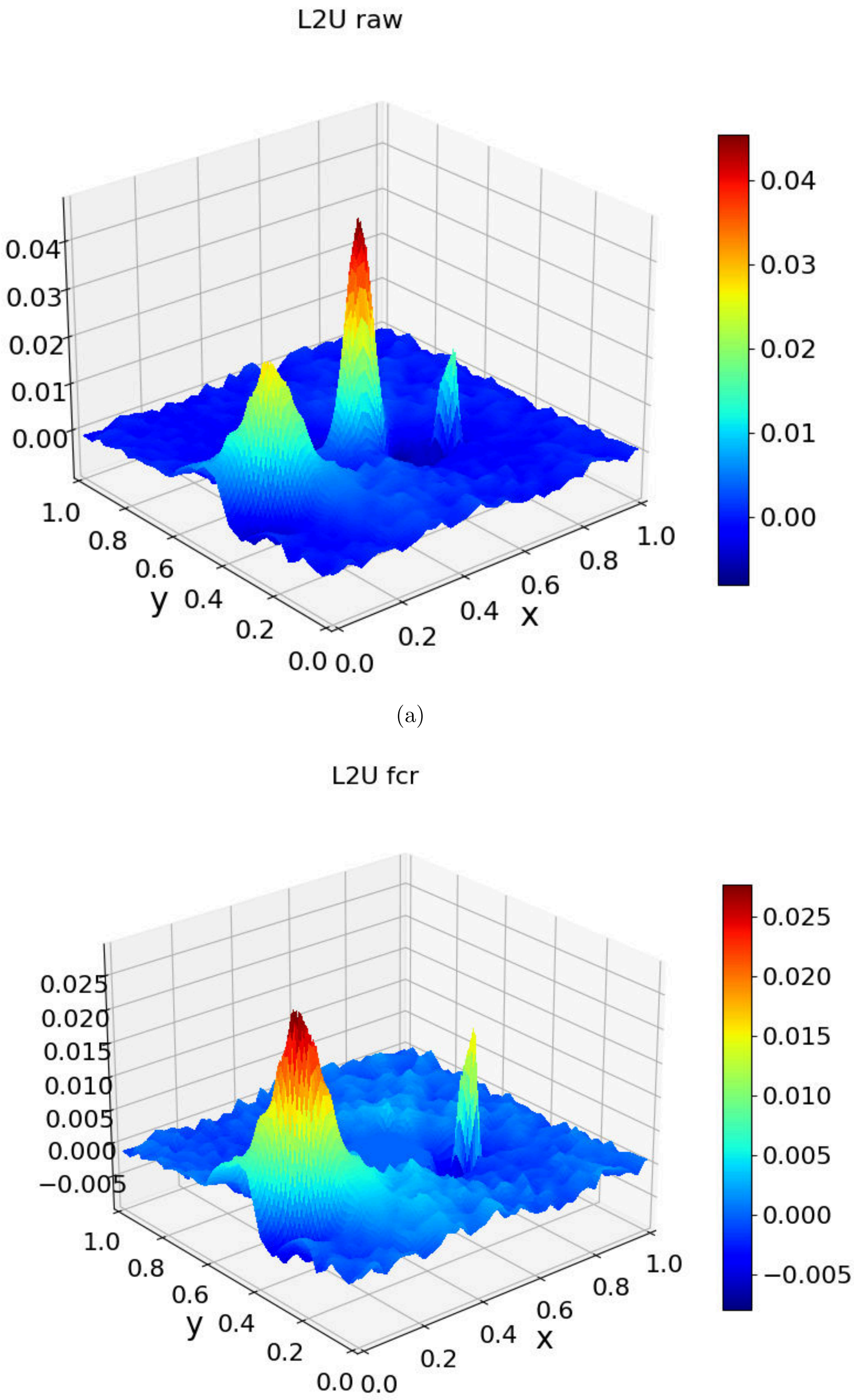
3.7.1 Default parameter setting

Image windowing

As described in Section 3.1, the input images are windowed prior to the discrete Fourier transform computation, to reduce the effects of spectral leakage. In general, most real input images are not periodic in x and y directions, and thus using the rectangular (boxcar) window (which is equivalent to not windowing at all) introduces significant spectral leakage effects in the cross-power spectrum, which in turn degrade the correlation landscape. Therefore, by default, the Hann window (see Blackman and Tukey [1958]) is applied to both input images.

Cross-power spectrum filtering

The cross-power spectrum filtering (described in Section 3.2) is mainly defined by three parameters - the band-pass filter type (rectangular, Gaussian), the amount of low-pass filtering (σ_H) and the amount of high-pass filtering (σ_L). As previously described, the Gaussian filter usually performs better when compared to the rectangular filter, since it does not produce ringing effects in the correlation landscape, and is thus the default. Due to noise, a moderate amount of low-pass filtering is usually beneficial when registering



(a)

(b)

Figure 3.17: IPC L^3 (see Section 3.3) origin masking. Resulting L^{2U} region obtained by registering images from Fig. 3.16 via the IPC sub-pixel image registration method (a), corresponding L^{2U} region with 5px circular origin mask, removing the false correlation peak at the origin (b).

real images. Thus, the default low-pass parameter value is $\sigma_H = 1$. Additionally, in most cases, some level of high-pass filtering is also beneficial, and thus the default high-pass parameter value is $\sigma_L = 0.01$. The cross-power normalization additive factor ξ is set to $\xi = 0$ by default for the sharpest correlation maxima.

Correlation upsampling

The correlation upsampling step (described in Section 3.3) is mainly defined by the size of the upsampled region $|L^{2U}|$ (or equivalently, by the upsampling factor U), and by the interpolation method. In most of the experiments done with various image types, sizes and noise amounts, the bilinear interpolation often gives the best results, and is thus the default interpolation type. The reason for this result is probably the sharpness of the bilinearly-upsampled correlation peak, when compared to the relatively flat correlation peaks obtained by nearest neighbor and bicubic interpolation (see the surface plots in Fig. 3.11). The default size of the L^{2U} sub-region $|L^{2U}| = 357$ is large enough for the possibility of high precision iterative correlation centroid refinement, and small enough for the two-dimensional interpolation to be still rather computationally inexpensive.

Iterative refinement

The optimal size of the L^1 sub-region used in the iterative shift refinement process (see Section 3.3.4 and 3.4) is mainly affected by the signal-to-noise ratio of the input images. Registering images with poor signal-to-noise ratio produces wider and flatter correlation peaks, for which the optimal L^1 region size is usually larger than for the very sharp correlation peaks obtained by registering images with high resolution and quality. The default size of the L^1 sub-region, expressed as the ratio of the L^{2U} sub-region size, is set as $L^1_r = 0.45$ (see Eq. 3.6).

3.7.2 Average sub-pixel registration accuracy function

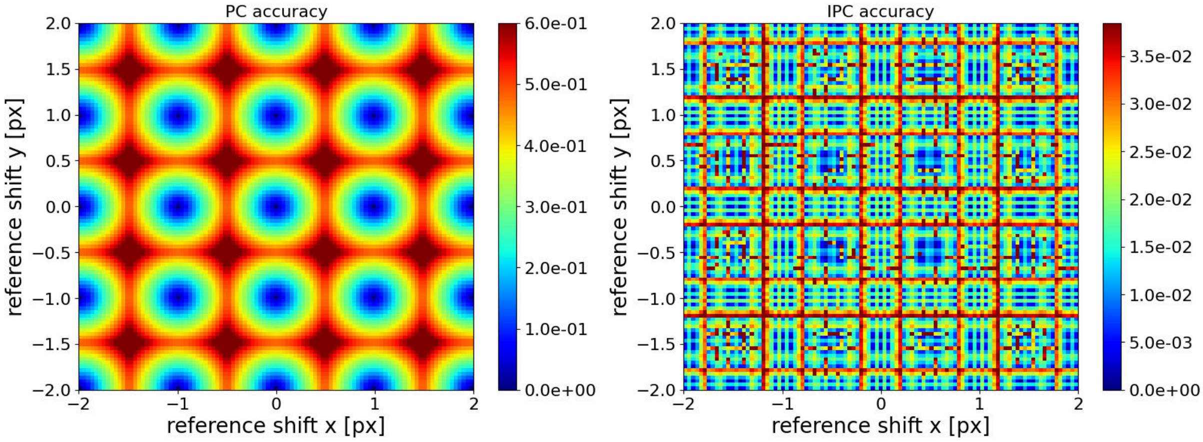
The central part of any optimization problem is the objective function. In this case, there are many unique ways to quantitatively express the performance of an image registration algorithm. The most natural and general measure of sub-pixel image registration performance is the sub-pixel image registration accuracy metric, defined as

$$\varepsilon = \sqrt{(\Delta\hat{x} - \Delta x)^2 + (\Delta\hat{y} - \Delta y)^2}, \tag{3.13}$$

where $[\Delta\hat{x}, \Delta\hat{y}]$ is the refined IPC shift estimate (see Eq. 3.12) and $[\Delta x, \Delta y]$ is the true (reference) underlying sub-pixel shift between the two input images. An image pair with a given pre-determined underlying sub-pixel shift $[\Delta x, \Delta y]$ can be obtained by artificially shifting an image via bilinear interpolation.

Firstly, to evaluate shifts of varying magnitudes and directions (and thus remove measurement bias), the sub-pixel image registration accuracy is evaluated on a linearly spaced two-dimensional grid of artificial shifts $[\Delta x, \Delta y]$, ranging from -2 to 2 pixels in each direction (see Fig. 3.18).

Secondly, mutually independent artificial Gaussian noise is added to each input image, to simulate noise present in real image pairs. For most accurate results, the noise characteristics of the artificial noise (mainly the mean and the standard deviation) should closely match the characteristics of the real noise in the images.



(a)

(b)

Figure 3.18: Examples of sub-pixel image registration accuracy measurement on a linearly spaced two-dimensional grid of 101×101 artificial image shifts ranging from -2 to 2 pixels in each direction. Standard phase correlation accuracy (a), IPC accuracy with default parameters (b).

Finally, to further improve robustness, the sub-pixel image registration accuracy metric can be computed on grids of artificial image pairs from multiple images. The resulting image-average and artificial image shift grid-average sub-pixel registration accuracy can then be defined as

$$\bar{\varepsilon} = \frac{1}{N_I N_x N_y} \sum_{i=1}^{N_i} \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} \sqrt{(\Delta\hat{x}_{ixy} - \Delta x_{ixy})^2 + (\Delta\hat{y}_{ixy} - \Delta y_{ixy})^2}, \tag{3.14}$$

where N_I is the number of evaluated image pairs, N_x is the number of image shift grid points in the x direction and N_y is the number of image shift grid points in the y direction. The average sub-pixel image registration accuracy objective function can be summarized by Algorithm 2.

3.7.3 Domain transforming function

For a given set of evaluated image pairs, the main underlying variables of the average sub-pixel image registration accuracy objective function are the parameters of the IPC algorithm. No special mathematical properties can be observed from the definition of the objective function in Algorithm 2, and thus the objective function is assumed to be non-convex and non-continuous with respect to the IPC parameters.

Most non-convex non-continuous optimizers work with continuous variables, and therefore an additional domain transforming (discretization) step needs to be introduced to the objective function in order to optimize all the logically discrete IPC parameters (window type, band-pass type, correlation interpolation type and $|L^{2U}|$). For all the "type" parameters, the discretization step of the relevant continuously optimized parameter p_i is

Algorithm 2: The average sub-pixel image registration accuracy objective function

```
- initialize $\bar{\varepsilon} = 0$;
for $i = 0$ to $N_I$ do
    - initialize $I_1$ as a copy of the $i$-th input image;
    - add Gaussian noise to $I_1$;
    for $x = 0$ to $N_x$ do
        for $y = 0$ to $N_y$ do
            $\Delta x := 2(2x/(N_x - 1) - 1)$;
            $\Delta y := 2(2y/(N_y - 1) - 1)$;
            - initialize $I_2$ by shifting $i$-th input image by $[\Delta x, \Delta y]$ via bilinear
              interpolation;
            - add Gaussian noise to $I_2$;
            - calculate the estimated IPC shift $[\Delta\hat{x}, \Delta\hat{y}]$ between $I_1$ and $I_2$;
            - add the sub-pixel image registration error (Eq. 3.13) to $\bar{\varepsilon}$;
        end for
    end for
end for
return $\frac{1}{N_I N_x N_y}\bar{\varepsilon}$;
```

defined analogically to the following example (for the correlation interpolation type):

$$\text{correlation interpolation type} = \begin{cases} \text{nearest neighbor} & 0 \leq p_i < 1 \\ \text{bilinear} & 1 \leq p_i < 2 \\ \text{bicubic} & 2 \leq p_i < 3. \end{cases} \tag{3.15}$$

Additionally, all of the continuously optimized parameters p_i are limited to a sensible range of values, and thus, for example, no additional edge cases need to be defined in Eq. 3.15 for $p_i < 0$ or $p_i \geq 3$.

For the integer parameters (like $|L^{2U}|$), the discretization step of the relevant continuously optimized parameter p_i is performed simply as

$$|L^{2U}| = \lfloor p_i \rfloor . \tag{3.16}$$

3.7.4 Training and validation datasets

Similarly to modern machine learning workflows, in order to avoid data overfitting [Hawkins, 2004], a good practise is to split the image datasets used for training (optimization) into training and validation (testing) subsets [Sarle et al., 1996]. The selected optimizer then only uses the information from the training dataset for the optimization process, while the performance on the validation dataset is used as the stopping criterion. The optimization process can be stopped early, when no further improvements in the validation performance are detected - a technique called “early stopping”. For example, in the optimization process shown in Fig. 3.19, the objective function value is still improving after the fifth iteration (generation), while the validation function value is not. In this case, the early stopping method would select the parameters from generation 5 as the optimal parameters.

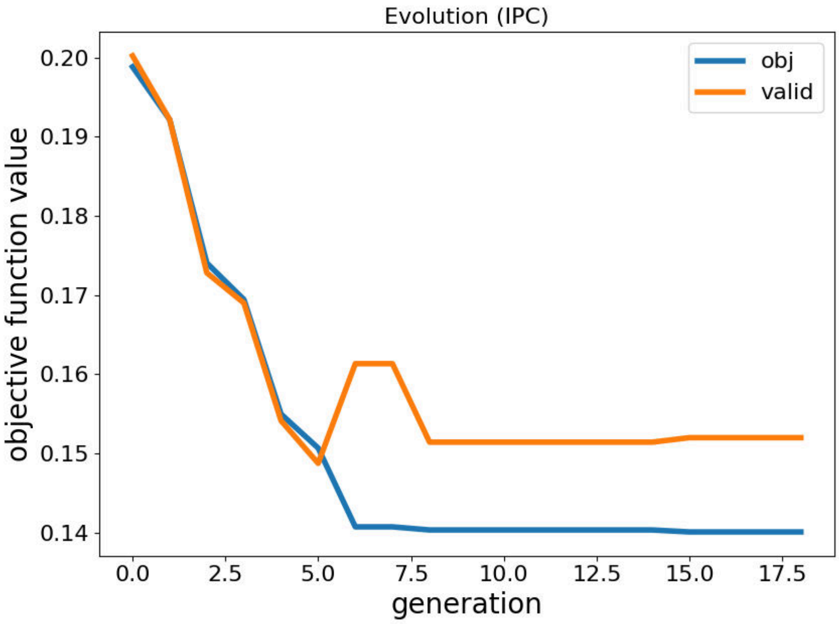


Figure 3.19: Average sub-pixel image registration accuracy improvement during continuous optimization by the differential evolution optimizer on pairs of noisy 64×64 images.

IPC parameters overfitted for a specific training image pair could imply reduced average accuracy for other image pairs, and thus the image pairs used for the IPC optimization process are automatically split into training and validation subsets and the validation sub-pixel image registration accuracy is monitored closely during the optimization process.

3.7.5 Non-convex and non-continuous optimizer

As previously discussed in Section 3.7.3, the average sub-pixel image registration accuracy objective function is generally non-convex and non-continuous. These properties significantly reduce the number of appropriate optimizers. One of the most robust and performant non-convex non-continuous derivative-free state of the art optimizers is the Differential evolution (DE) optimizer [Price, 2013, Das and Suganthan, 2010, Pant et al., 2020]. The only inputs of the DE optimizer are the objective function and the lower and upper bounds of each of the optimized parameters. The DE optimizer then initializes a random "population" of possible solutions within the specified parameter bounds and then produces progressively improved parameter combinations in each iteration (evolution generation), until one of the termination criteria are met. With large enough population size, the DE optimizer is very robust in finding the global objective function optimum.

3.8 Efficient and reliable implementation

The efficiency, reliability and safety characteristics of any algorithm implementation are crucial for its applicability in both research and commercial environments. The IPC algorithm is no exception. In most real-world image registration applications, the image registration process implementation needs to be efficient (computing the result in the shortest amount of time possible) and reliable (not producing erroneous results).

3.8.1 Implementation efficiency

Fast Fourier Transform

Since three two-dimensional discrete Fourier transforms need to be calculated during the IPC algorithm (two real-to-complex transforms and one complex-to-real transform), the most straightforward efficiency improvement is to improve the computational efficiency (complexity) of the Fourier transforms. This can be done by using the two-dimensional Fast Fourier Transform (FFT) algorithm [Brigham and Morrow, 1967] instead of the Discrete Fourier Transform algorithm (DFT), which effectively reduces the time complexity from $O(W^2 \cdot H^2)$ to $O(W \cdot H \log(W \cdot H))$, where W and H are the width and height of the two-dimensional input. A frequently used reliable implementation of the FFT algorithm, named FFTW: The Fastest Fourier Transform in the West [Frigo and Johnson, 2005], implemented in the C programming language, can be found at `www.fftw.org`.

GPU utilization

Furthermore, on modern hardware, the computation of the FFT can be quite easily sped up with parallel computation. Most commonly in performance-critical applications, the FFT is computed by utilizing the many cores of modern graphics processing units (GPUs), instead of the central processing units (CPUs). One example of a popular library for computing FFTs on the GPU is the CUDA FFT library (cuFFT), which is available at `developer.nvidia.com/cufft`. The benefit of the cuFFT library is that it has a similar application programming interface (API) to the FFTW library, which makes converting between programs made for computing efficient FFTs on the CPU and GPU easier.

Complex-conjugate-symmetric packing

Since all image pixel values are purely real, the DFT of an image is always complex-conjugate symmetric ($\hat{f}(\xi) = \overline{\hat{f}(-\xi)}$). This means, that all the DFT information of a $W \times H$ image can be stored in just $W \times H/2$ complex values. The benefit of such complex-conjugate-symmetric (CCS) DFT packing is not only in the amount of memory needed to store the DFT, but also in the amount of computation needed to compute the cross-power spectrum. Since the CCS-packed DFT contains only half the amount of DFT values compared to the normal (redundant) image DFT representation, computation of the cross-power spectrum is twice as efficient and takes half the memory.

Reusing window and band-pass matrices

In applications, where the IPC algorithm is run repeatedly with image pairs with the same size (typically some type of optical flow measurements), the window matrix and the band-pass matrix can be computed only once and then reused for all subsequent IPC computations. An elegant way to implement this functionality is via object-oriented programming (OOP).

In the C++ OOP implementation of the IPC algorithm (see Section 3.8.3), firstly, an object of type IPC is created. The creator of this object needs to specify the image size and band-pass parameters (see Section 3.2), for which this IPC object will be used. During the object construction, the window and band-pass matrices are computed for the supplied image size and band-pass parameters. Image registration results can then be obtained by repeatedly calling a computing member method, which reuses the pre-computed window and band-pass parameters with no additional computational cost.

The IPC object can also have methods which change the image size or band-pass parameters dynamically. These modifying methods can be called even after the object was already fully constructed. However, in multi-threaded environments, mutual-exclusion mechanisms need to be added to the modifying methods to prevent data races (simultaneous reading and writing of shared memory). These mutual-exclusion mechanisms effectively make the thread, which tries to call the modifying function (and thus change the window and/or the band-pass matrices) wait until all currently running IPC computations finish, and only then modify the window and band-pass matrices in a thread-safe manner (while all the calls to the compute method wait). The C++ Standard Template Library (C++ STL) has a variety of such mutual-exclusive tools with intuitive API, such as `std::mutex` or `std::recursive_mutex`.

3.8.2 Implementation reliability

Automatic $|L^1|$ reduction

During the iterative refinement process of the IPC algorithm (see Section 3.4), maximum iteration count can be reached in some circumstances. Although this happens very rarely with common images, it means, that the condition of convergence was not met in any of the computed iterations - the sub-pixel centroid coordinates diverged or oscillated. In such cases, the overall accuracy of the algorithm can be improved by reducing the size of the circular upsampled phase correlation peak neighborhood $|L^1|$ (preferably by reducing the L^1 size ratio L^1_r), and running the iterative refinement process again. The likelihood of convergence increases significantly with smaller L^1 sizes. This modification is already included in the iterative refinement process description in Algorithm 1.

There are multiple possible reasons why maximum iteration count can be reached. For example, if the input images contain significant amount of noise (or equivalently, if the input images do not overlap enough), the resulting neighborhood of the strongest phase correlation peak can contain multiple other similarly strong peaks, and thus the sub-pixel centroid can diverge or oscillate between them. Furthermore, reaching the maximum iteration count can be a direct consequence of a too large initial L^1 size.

The automatic L^1 size reduction mechanism solves the divergence/oscillation problem reliably and efficiently, since the computational cost of the iterative refinement process is insignificant when compared to the DFT computation. Returning the computed sub-pixel shift with the largest L^1 size that satisfied the condition of convergence before reaching the maximum iteration count can significantly improve the resulting sub-pixel accuracy compared to the integer pixel result.

Out-of-bounds checking

From the standpoint of algorithm execution safety, the most "dangerous" operation of the IPC algorithm is the extraction of a sub-region from a matrix at a certain point with a certain size. This operation happens in two ways during the IPC process. The first type of sub-region extraction is executed after the raw L^3 phase correlation neighborhood is computed - a square neighborhood of the maximum phase correlation value $L^3_{\max}[x, y]$ is extracted from L^3 with size $|L^2|$ (see Section 3.3). The second type of sub-region extraction is executed in each iteration of the iterative refinement process - a circular sub-region L^1 is extracted from L^{2U} at the current $L^{2U}_{\max}[x, y]$ location with size $|L^1|$ (see Section 3.3).

In cases, where the point of extraction lies sufficiently close to the boundary, its neighborhood with given size might not be entirely within image bounds. Without additional safety precautions, a sub-region extraction operation could in such cases attempt to read data from outside of the allocated memory, which would result in undefined behavior, or instantaneous process termination. The solution of this problem is a quick out-of-bounds check performed before the sub-region extraction. For a $W\times H$ matrix, $S\times S$ square sub-region and $[x, y]$ coordinates of sub-region extraction, the out of bounds check can be implemented simply as the following condition:

$$\left(x - \frac{S}{2} \geq 0\right) \wedge \left(y - \frac{S}{2} \geq 0\right) \wedge \left(x + \frac{S}{2} \leq W\right) \wedge \left(y + \frac{S}{2} \leq H\right). \tag{3.17}$$

If the out-of-bounds condition is not met, the size of the corresponding sub-region ($|L^2|$ or $|L^1|$) is reduced, and the extraction process is attempted again. In the L^2 sub-region case, with each out-of-bounds failure, the sub-region size is reduced by 2 (to maintain odd parity), down to $|L^2| = 3$, after which the algorithm returns pixel precision image shift estimate only. However, it is highly unlikely, that even a L^2 sub-region with $|L^2| = 3$ would not fit entirely in L^3. In the L^1 sub-region case, the sub-region size is reduced by reducing the L^1_r parameter by a fixed amount and then resetting the iterative refinement process.

3.8.3 Available implementation

An efficient and reliable object-oriented implementation of the IPC algorithm, implemented by the author in modern C++ programming language and compatible with the C++ Open Computer Vision Library (OpenCV) [Bradski, 2000], containing all the mentioned efficiency and reliability improvements, can be found at https://github.com/zdenyhraz/shenanigans/blob/master/src/ImageRegistration/IPC.hpp.

3.9 Accuracy measurements and comparison

The performance of various sub-pixel image registration methods can be objectively evaluated by the average sub-pixel image registration accuracy metric (as described in Alg. 2) on a given dataset of shifted image pairs (see an example in Fig. 3.20). The sub-pixel image registration accuracy results for various methods, image sizes and noise levels can be seen in Table 3.1.

Naturally, the accuracy results of all sub-pixel methods show a clear trend of decreasing accuracy with increasing noise levels. The standard deviation of the sub-pixel accuracy also seems to increase with increasing noise levels for all sub-pixel methods. The accuracy results of the IPC method with optimized parameters show significantly improved average sub-pixel accuracy and also improved sub-pixel accuracy standard deviation. This improvement can clearly be seen in Fig. 3.21 (notice the significantly tighter 1-sigma bands for IPC with optimized parameters).

$W \times H$	σ	PC	PCS	IPC	IPCO
32×32	0.00	0.392±0.147	1.344±0.556	0.079±0.033	**0.029±0.019**
32×32	0.02	0.392±0.147	1.583±0.751	0.087±0.045	**0.076±0.038**
32×32	0.05	0.401±0.158	1.734±1.177	0.171±0.089	**0.112±0.058**
64×64	0.00	0.392±0.147	0.253±0.122	0.035±0.017	**0.012±0.006**
64×64	0.02	0.392±0.147	0.321±0.151	0.040±0.020	**0.022±0.011**
64×64	0.05	0.392±0.147	0.370±0.185	0.069±0.034	**0.051±0.026**
128×128	0.00	0.392±0.147	0.239±0.080	0.036±0.017	**0.009±0.005**
128×128	0.02	0.392±0.147	0.270±0.079	0.032±0.016	**0.014±0.007**
128×128	0.05	0.392±0.147	0.244±0.089	0.041±0.020	**0.034±0.017**
256×256	0.00	0.392±0.147	0.173±0.083	0.024±0.013	**0.008±0.005**
256×256	0.02	0.392±0.147	0.170±0.064	0.028±0.014	**0.010±0.005**
256×256	0.05	0.392±0.147	0.169±0.059	0.036±0.017	**0.014±0.007**

Table 3.1: Average sub-pixel image registration accuracy and its standard deviation measured on datasets of image pairs with various sizes (width W and height H) and noise levels (standard deviation σ). Measured image registration methods: standard phase correlation (PC), OpenCV implementation of a 5×5 weighted centroid sub-pixel phase correlation (see https://docs.opencv.org/4.x/d7/df3/group__imgproc__motion.html) (PCS), Iterative Phase Correlation (IPC), Iterative Phase correlation with optimized parameters (IPCO).

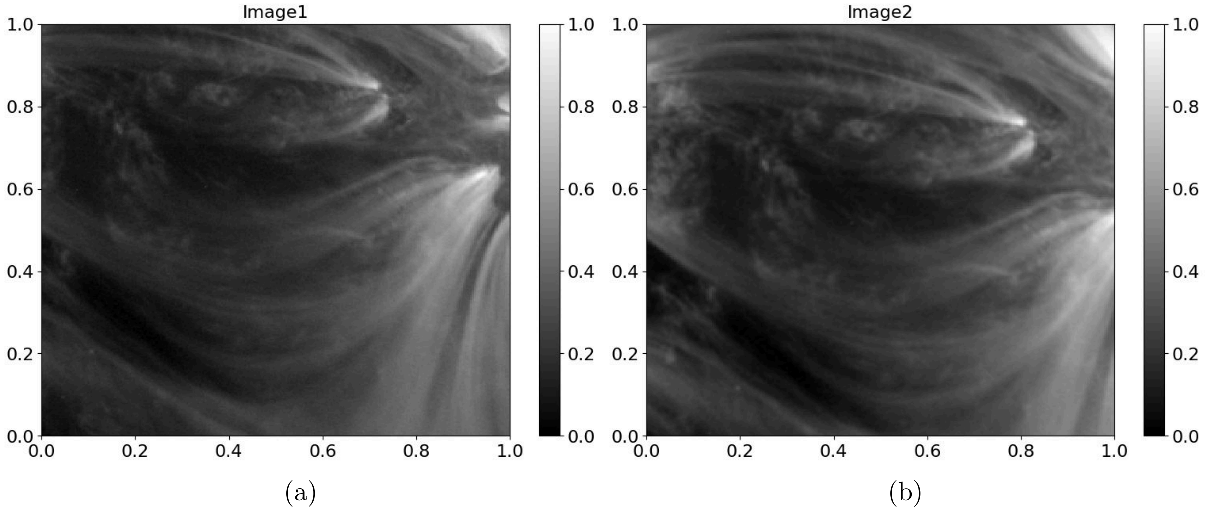


Figure 3.20: An example of a 256×256 shifted image pair used for sub-pixel image registration accuracy measurements in Table 3.1.

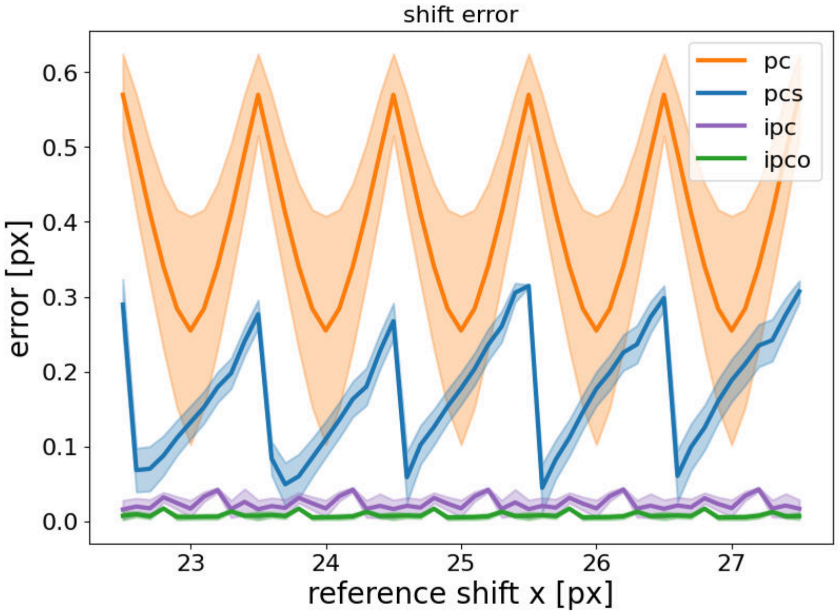


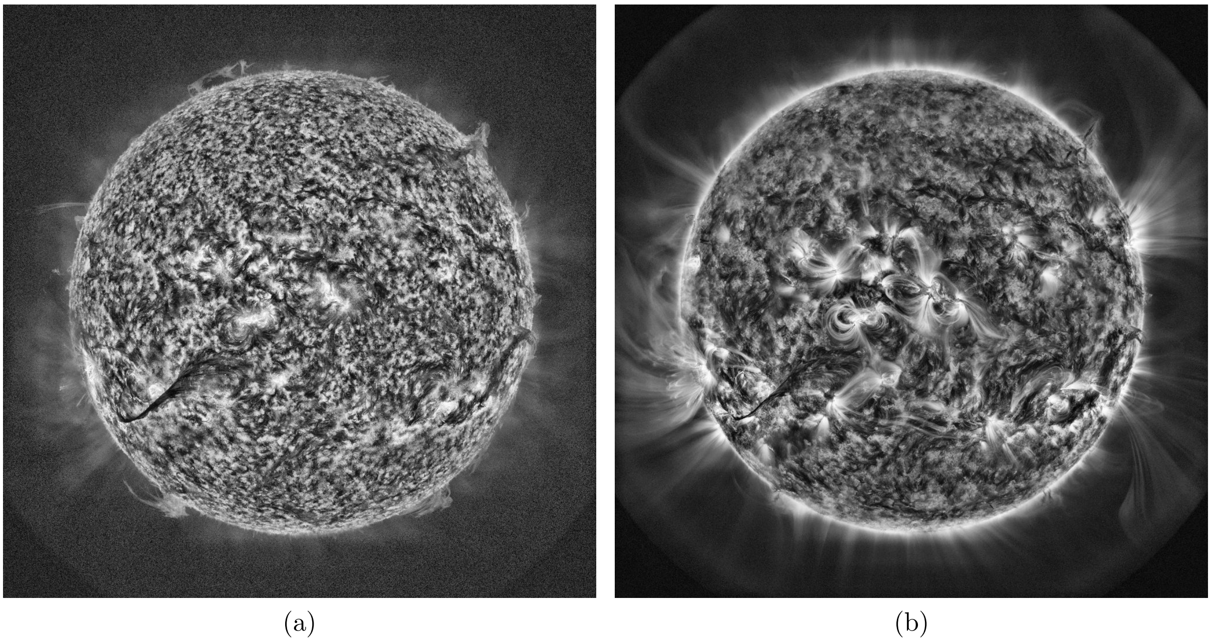
Figure 3.21: An example of sub-pixel image registration accuracy measurements on a dataset of 256×256 image pairs showing y-axis shift average sub-pixel image registration accuracy and its y-axis 1-sigma band for methods from Table 3.1.

Chapter 4

Applications

4.1 Dissimilar image alignment

In many different fields dealing with image data (e.g. astrophotography, microscopy, satellite imaging and more), it is often necessary to compose images taken with different instruments and varying wavelengths. These images can be noticeably misaligned due to differences in instrument positions, varying times of acquisition and/or different instrument optics. Misalignments typically include a combination of image translation, image rotation and scaling. To accurately align two such images, four parameters need to be precisely estimated - translational shifts Δx and Δy in the x and y directions, rotation angle φ and a scale factor s. If these four parameters are computed with good sub-pixel accuracy, the resulting composed image can have distinctly better characteristics than a composed image obtained by standard pixel-level accuracy image registration methods.


(a)

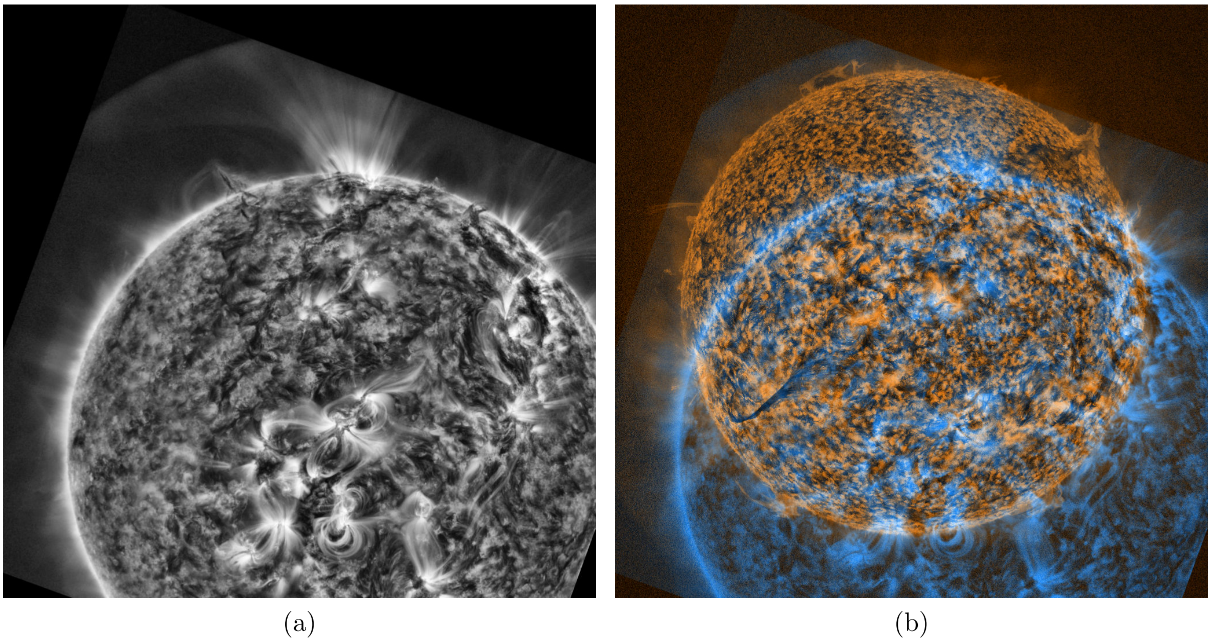
(b)

Figure 4.1: An example of a significantly dissimilar image pair - contrast-enhanced 304Å (a) and 171Å (b) images taken by the Atmospheric Imaging Assembly onboard the Solar Dynamics Observatory.

Both correlation-based and feature-based image registration methods often fail to correctly align images with low levels of similarity. A typical application, where aligning visibly very dissimilar images with high accuracy is required, is the composition of images taken by various instruments of the Atmospheric Imaging Assembly (AIA) onboard the Solar Dynamics Observatory (SDO) [Lemen et al., 2011]. A typical example of a dissimilar image pair obtained by 304Å and 171Å SDO/AIA instruments roughly at the same moment can be seen in Fig. 4.1.

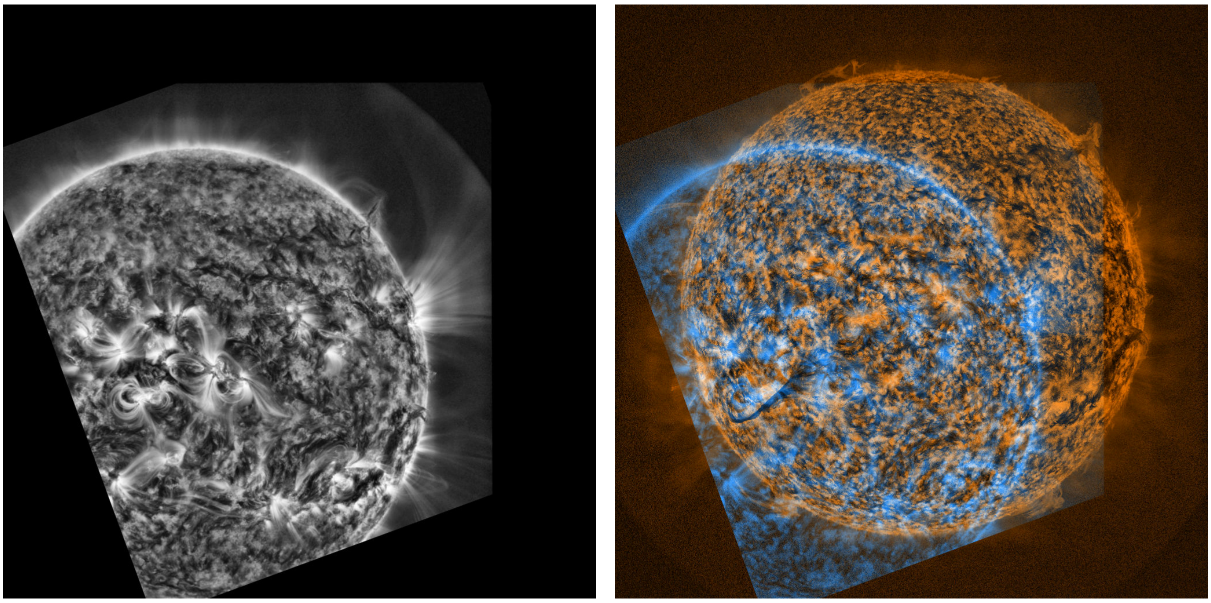
Contrary to standard rotation/scale/translation image alignment methods, the IPC general image alignment method (described in Section 3.5) can adapt to the vastly different characteristics of the input images via parameter optimization (see Section 3.7). This increases the robustness of the alignment and increases the interval of image similarity, in which the method aligns images correctly.

The three main stages of the IPC general image alignment procedure are demonstrated on a pair of dissimilar 304Å and 171Å SDO/AIA images (from Fig. 4.1). To better demonstrate the robustness of the method, the second (171Å) image is significantly artificially rotated, scaled (enlarged) and shifted prior to the alignment process. As described in the IPC general image alignment process flowchart in Fig. 3.14, the second (171Å) image is aligned to the stationary first (304Å) image. Figures 4.2, 4.3 and 4.4 show the gradual alignment process of the second image. Fig. 4.2 shows the second image before the alignment (artificially rotated, scaled and shifted), Fig. 4.3 shows the second image after scale and translation alignment and Fig. 4.4 shows the second image after rotation, scale and translation (complete) alignment.


(a)

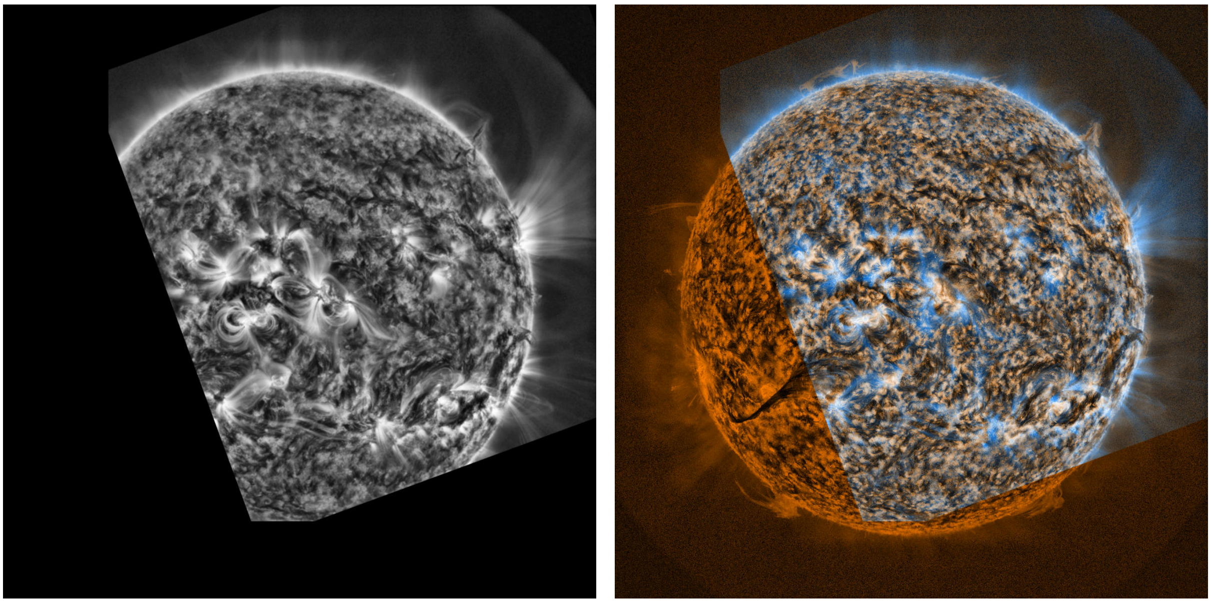
(b)

Figure 4.2: General IPC image alignment of SDO/AIA 304Å and 171Å images - before the alignment procedure. Artificially rotated, scaled and shifted SDO/AIA 171Å (second) image (a), red-blue color composition of original SDO/AIA 304Å image and artificially rotated, scaled and shifted 171Å image (b).



(a) (b)

Figure 4.3: General IPC image alignment of SDO/AIA 304Å and 171Å images - after rotation and scale alignment. Rotation and scale aligned 171Å (second) image (a), red-blue color composition of original SDO/AIA 304Å image and rotation and scale aligned 171Å image (b).



(a) (b)

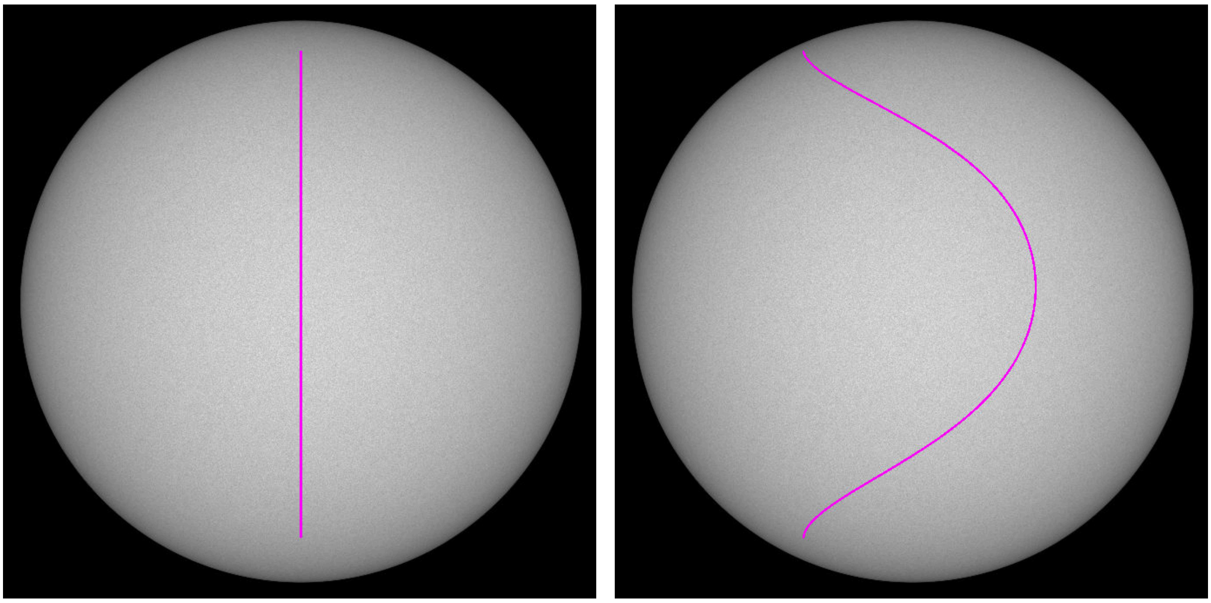
Figure 4.4: General IPC image alignment of SDO/AIA 304Å and 171Å images - after rotation, scale and translation alignment. Rotation, scale and translation aligned 171Å (second) image (a), red-blue color composition of original SDO/AIA 304Å image and rotation, scale and translation aligned 171Å image (b).

Additionally, the alignment figures also contain a “red-blue” color composition of the input images to better visualize the dissimilar alignment. The red channel of the color-composed images contains data from the first (304Å) image, the blue channel contains data from the second (171Å) image and the green channel contains the average of both images. The correctness of the resulting alignment can clearly be identified in the color composition shown in the complete alignment Figure 4.4 (b).

4.2 Solar differential rotation speed measurement

4.2.1 Introduction

The identification of sunspots with features on the solar surface led to the discovery of solar rotation in the 17th century, and the variation of the rotation speed with latitude, or the differential rotation rate (see Fig. 4.5), by Scheiner [1630]. Advances in observational tools over the centuries led to the discovery of the oscillation modes of the Sun [Leighton et al., 1962], which established the field of helioseismology [see review by Leibacher et al., 1985]. Helioseismology further revealed that the differential rotation rate changes with depth in the convection zone [Thompson et al., 1996]. Given that the differential rotation rate is now established as a necessary mechanism for generating solar magnetic fields, i.e. the solar dynamo [Elsasser, 1950, Babcock and Babcock, 1955, Parker, 1955], the accurate determination of the differential rotation rate, starting from the solar surface, becomes even more compelling.


(a) $t = 0$

(b) $t = 27$ days

Figure 4.5: Demonstration of the solar differential rotation effect on SDO/HMI images during a 27 day period ($\sim$ 1 solar revolution - rough average between the rotation period at the equator and the rotation period near the poles). Hypothetical line formed by regions aligned on the central meridian (a), hypothetical curve formed by the same regions after a 27 day period (b).

Recently Beck [2000] compared the differential rotation rates derived from the most widely used techniques, namely the Doppler shifts of photospheric spectral lines, the Doppler feature tracking, the magnetic feature tracking and the p-mode splittings. He found a large disparity in the values inferred from these different techniques. The difficulties hampering the accuracy of the measurement are directly linked to the complexity of observed features on the solar surface itself. Additional limitations due to foreshortening effects close to the poles limit the viability of these techniques to low latitudes. Sunspot tracking adds to the limitation, since they are rarely observed above 30° latitude. Furthermore, since all tracked features, such as sunspots, supergranules and granules, for example, are subject to intrinsic temporal variability, they will inevitably bias any measurement of rotation rate.

4.2.2 Measurement technique

In this work, the inference of the solar rotational speed at different latitudes is achieved by measuring the local west-east and north-south image shifts Δx, Δy between pairs of *Solar Dynamics Observatory* (SDO) *Helioseismic and Magnetic Imager* (HMI) continuum full disk 4096x4096 images with time step Δt by means of image registration (see Fig. 4.6). Measured image shifts are then projected onto the solar surface, allowing the corresponding angular shifts $\Delta\varphi_x$, $\Delta\varphi_y$ to be found. The average angular velocities ω_x and ω_y are then calculated based on the time difference Δt between the two images following the relations:

$$\omega_x = \frac{\Delta\varphi_x}{\Delta t}, \tag{4.1}$$

$$\omega_y = \frac{\Delta\varphi_y}{\Delta t}, \tag{4.2}$$

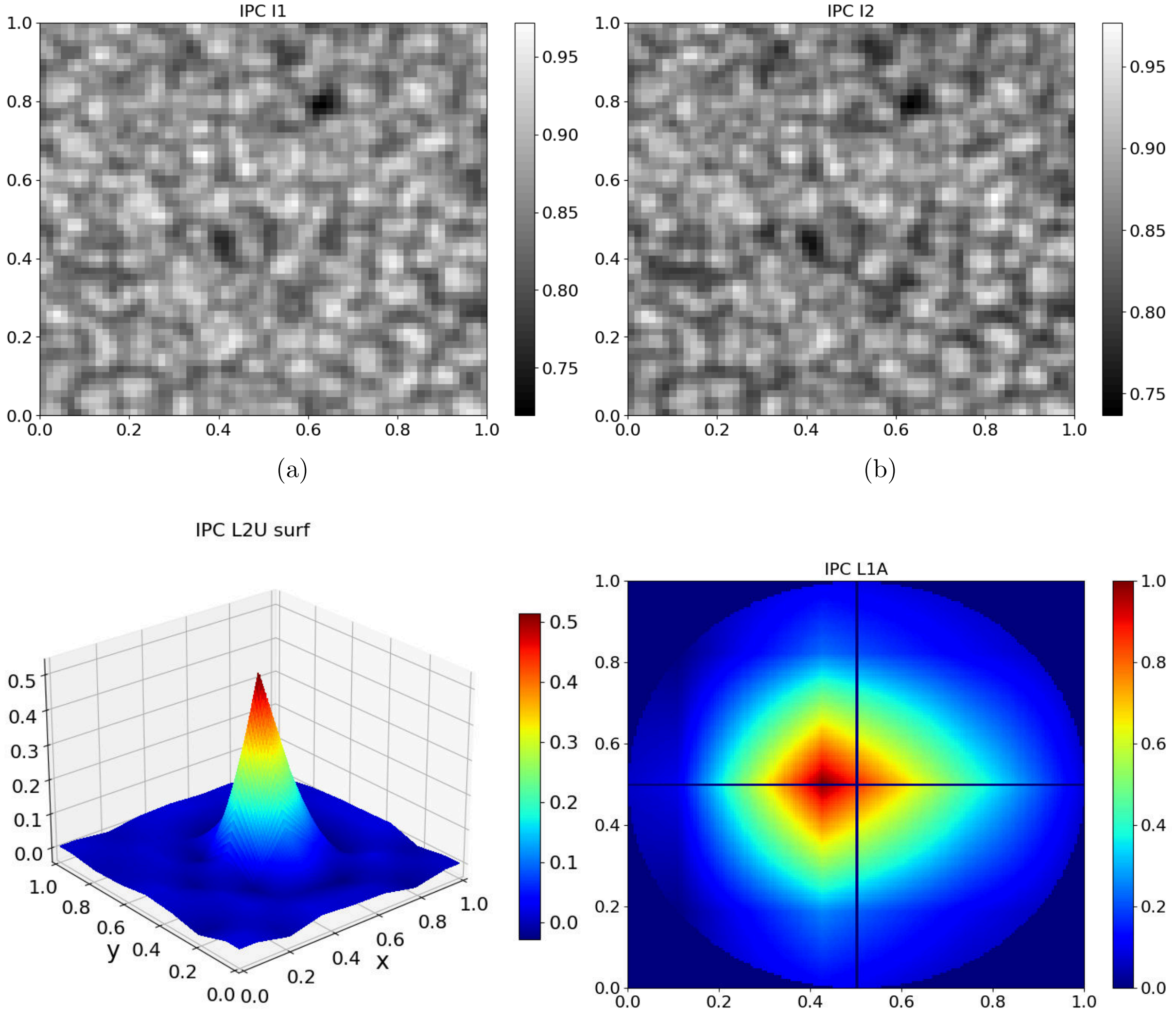
where the time difference Δt is determined by the SDO HMI continuum instrument temporal cadence Δ_t and the difference between the image indices Δ_p, namely

$$\Delta t = \Delta_p \Delta_t. \tag{4.3}$$

The SDO/HMI temporal cadence is constant ($\Delta_t = 45\text{s}$). Pairs of consecutive images ($\Delta_p = 1$) are used to ensure the best signal-to-noise rato for the image registration and thus to achieve a high precision measurement of the image shifts Δx and Δy.

4.2.3 Time step choice

The accuracy and reliability of the image registration technique depends heavily on the chosen time step Δt. As mentioned in the previous section, a time step of 45s is used for all the measurements throughout the manuscript. The reasoning behind this choice can be seen visually in Figures 4.8 and 4.7. Since the SDO/HMI images mostly contain information about the granulation on the solar surface (see Fig 4.6 (a) and (b)), the time difference between the two pictures is limited by the lifetime of these granules. A typical granule has a lifetime of around 10-15 minutes, which limits the time step to around 4 minutes, to still keep good overlap and correlation between the images. The lowest possible time step of 45s is thus chosen to maximize the signal-to-noise ratio. Figure 4.8 shows how the image shift measurement becomes consistently highly unreliable with time steps larger than 4 minutes, while Figure 4.7 visualizes the phase correlation maximum



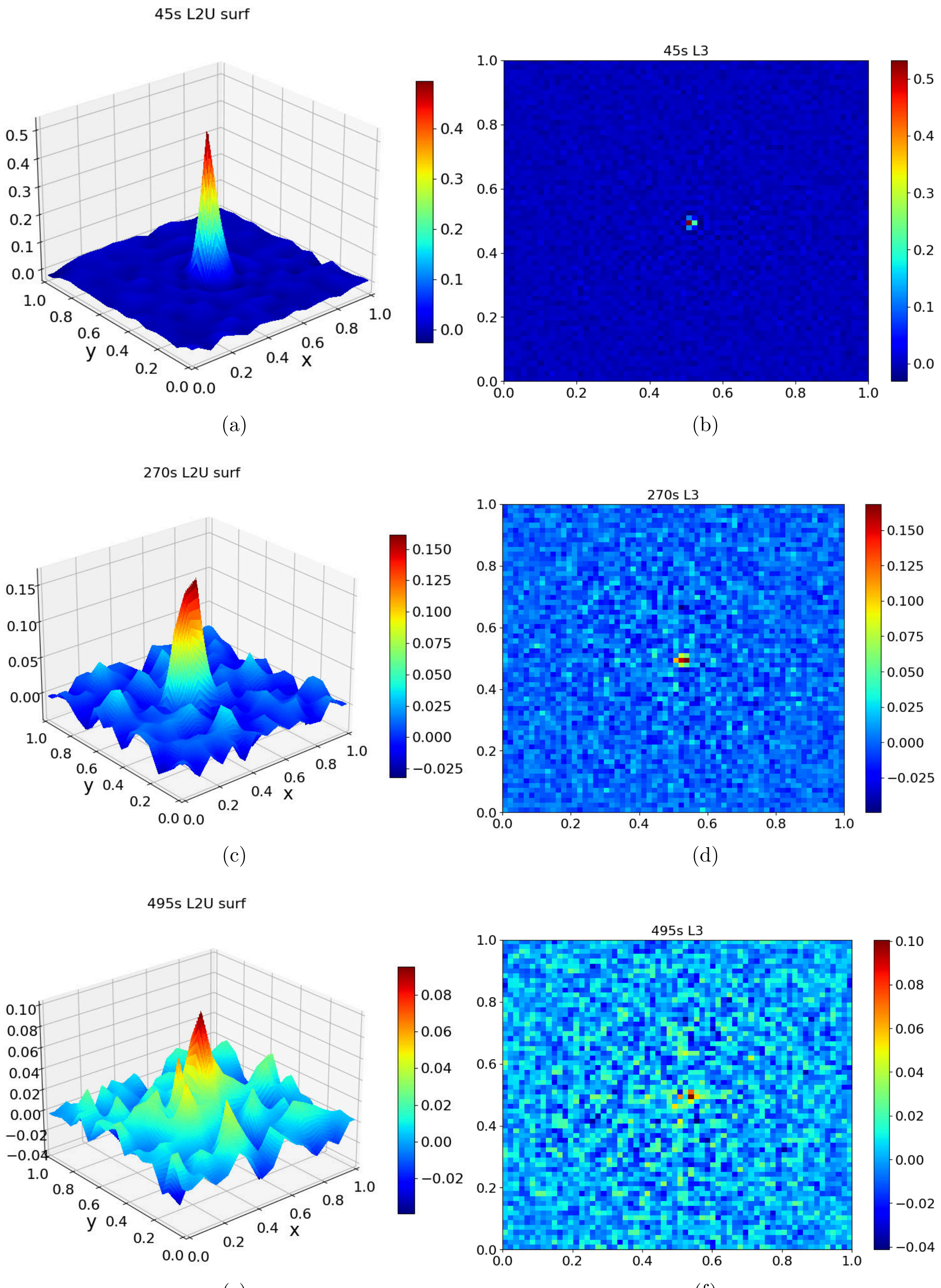
(a)

(b)

(c)

(d)

Figure 4.6: High precision sub-pixel image shift estimation via iterative phase correlation on a pair of 64×64 SDO/HMI image sub-regions with time step $\Delta t = 45$s - first input image (a), second input image (b), surface-heatmap plot of the IPC L^{2U} sub-region computed from the two input images (c), heatmap of the corresponding IPC L^{1} sub-region with marked centroid location (d).



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.7: Iterative phase correlation sub-regions obtained from pairs of 64×64 SDO/HMI images with 45s, 270s and 495s time steps. Corresponding surface-heatmap plots of L^{2U} IPC sub-regions (a), (c), (e), heatmap plots of L^3 IPC sub-regions (b), (d), (f). Note the significant decrease of maximum correlation for increasing time steps (the color ranges were min/max normalized for each landscape).

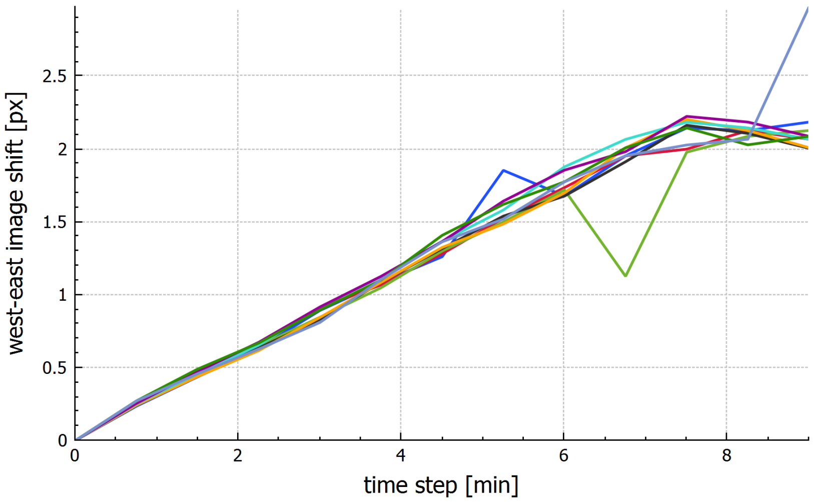


Figure 4.8: West-east image shifts measured by the IPC algorithm for multiple pairs of SDO/HMI images with different time steps Δt. Each curve represents different random first image.

peak quality degradation with larger time steps - note the sharp decline of the maximum phase correlation values from (a) to (c).

4.2.4 The image registration algorithm

The local image shifts were calculated for a chosen set of pixel coordinates (and their neighborhoods) by the iterative phase correlation (IPC) sub-pixel image registration algorithm. As the vast majority of west-east shifts between pairs of consecutive ($\Delta t = 45s$) images lie in the interval $[0.1, 0.2]$ px, the high sub-pixel precision of the IPC algorithm justifies the choice of the method. The main relevant advantage of the IPC algorithm is the potential for very high sub-pixel spatial accuracy with correct parameters, due to its iterative nature. These parameters can be optimized beforehand for a given type of image (different sizes, different amounts of noise, etc.). The main limitation of the technique is the amount of noise present in the images and the magnitude of the shift between them. Naturally, with higher noise levels, the method starts to lose its sub-pixel accuracy. Similarly, with larger image shifts, the amount of overlap between the two images starts to diminish, which also hinders the quality of the correlation landscape and thus lowers the accuracy of the resulting measured image shift. Both of these limiting effects are minimized by choosing a very short time step.

For each of the registered pixels, two equal-sized square neighborhoods (sub-regions) are selected from each of the two SDO/HMI images at equal locations relative to the image reference pixel $[x_{mid}, y_{mid}]$, defined in the header of each SDO/HMI *Flexible Image Transport System - FITS* file. This way, changes in the image coordinates of the solar center caused by movements of the SDO are taken into account and do not degrade the results. Since the time interval between the correlated images is only 45 seconds, all such changes are negligible in comparison with the measured image shift. Median absolute shift of the solar center was measured at approximately 0.005px in both directions (both according to the FITS file header and according to measurements by the IPC method). The corresponding image sub-region pairs are then iteratively phase correlated, yielding the pixel shift estimates Δx and Δy for each of the selected locations.

4.2.5 Locations of the image shift measurements

To reduce the effects of distortions arising due to the perspective projection of the 3D solar surface onto the 2D image, only pixels lying on the central meridian (and their small neigborhoods) are considered. This also allows for the construction of a central meridian flow speed map, with solar latitude θ on the vertical axis and time t (assuming the image pairs are ordered chronologically) on the horizontal axis. For each of the considered picture pairs, the central sub-region pixels are distributed evenly across a portion of the central meridian, symmetrically around the central pixel $[x_{mid}, y_{mid}]$. The entire meridian (from $\theta = -90$ to $\theta = 90$) is not considered, given the noticeable distortions caused by the perspective projection of the solar surface near the geographic poles. The vertical pixel size of the limited meridional "field of view" is labeled as Θ_y, the number of individual registered sub-regions on the meridian of each picture pair is labeled as M and index each of the sub-regions from north to south with the ascending index is labeled as μ (starting from zero). It is further assumed that the standard image coordinate system has the x coordinate (image column) increasing from left to right, and the y coordinate (image row) increasing from top to bottom. To only measure the shifts on the central meridian, the central horizontal pixel coordinates of each sub-region $x(\mu)$ need to be fixed for all images at

$$x(\mu) = x_{mid}, \tag{4.4}$$

while the central vertical pixel coordinates of each sub-region $y(\mu)$ (satisfying the equidistant and symmetry conditions mentioned above) can be set as

$$y(\mu) = y_{mid} + \Delta_y \left(\mu - \left\lfloor \frac{M}{2} \right\rfloor \right), \tag{4.5}$$

where Δ_y is the vertical spacing between the equidistant sub-region centers and can be determined by

$$\Delta_y = \frac{\Theta_y}{M-1}. \tag{4.6}$$

The solar latitude $\theta(\mu)$ can be trigonometrically derived for every meridional sub-region index μ as

$$\theta(\mu) = \arcsin \left(\frac{\Delta_y \left(\lfloor \frac{M}{2} \rfloor - \mu \right)}{R} \right) + \theta_0, \tag{4.7}$$

where the two newly introduced variables R and θ_0 denote the size of the solar radius in pixels and the latitude of the observer (instrument), and are both described in the SDO/HMI FITS file header. For the differential rotation measurement, the vertical field of view is chosen as $\Theta_y = 3400$ px and the number of sub-regions as $M = 851$ (an odd number of sub-regions, assuring the existence of a central element). According to equation 4.6, this results in a $\Delta_y = 4$ px vertical spacing between the equidistant meridional central sub-region pixels.

4.2.6 Orthographic projection of the image shift

Since the SDO/HMI image is essentially a 2D perspective projection of the 3D solar surface, the measured meridional image shifts $\Delta x(\mu)$ and $\Delta y(\mu)$ (calculated by iteratively phase correlating pairs of image sub-regions with centers at pixel coordinates $[x(\mu), y(\mu)]$ given by equations 4.4 and 4.5) need to be projected back onto the solar surface and transformed into the angular shifts $\Delta\varphi_x(\mu)$ and $\Delta\varphi_y(\mu)$. Since the distance between the Solar Dynamics Observatory and the Sun is $\sim 200R_s$, the image projection was assumed orthographic (parallel projection lines orthogonal to the projection plane - point of perspective at infinite distance). As only the central meridian sub-region shifts are calculated for every image pair, the following image shift backprojection equations depend only on the latitude information $\theta(\mu)$ (given in equation 4.7) and can also be derived trigonometrically. These orthographic backprojections transform the image pixel shifts $\Delta x(\mu)$, $\Delta y(\mu)$ into the angular shifts $\Delta\varphi_x(\theta(\mu))$ and $\Delta\varphi_y(\theta(\mu))$ given by

$$\Delta\varphi_x(\theta(\mu)) = \arcsin\left(\frac{\Delta x(\mu)}{R\cos\left(\theta(\mu)\right)}\right), \tag{4.8}$$

$$\Delta\varphi_y(\theta(\mu)) = \theta(\mu) - \arcsin\left(\sin\left(\theta(\mu)\right) - \frac{\Delta y(\mu)}{R}\right). \tag{4.9}$$

4.2.7 Equations for the image shift-based solar angular velocity

The six previously described equations (the general equations 4.1 and 4.2 for the angular velocities $\omega_x(\theta(\mu))$ and $\omega_y(\theta(\mu))$, equation 4.3 for the temporal difference between the two images Δt, equation 4.7 for the solar latitude $\theta(\mu)$ and the orthographic projection equations 4.8 and 4.9 for the angular shifts $\Delta\varphi_x(\theta(\mu))$ and $\Delta\varphi_y(\theta(\mu))$) are combined into the final equations for the estimated angular velocities $\omega_x(\theta(\mu))$ and $\omega_y(\theta(\mu))$ of a meridional pixel with index μ (with pixel indexing as described in section 4.2.5) in the form of

$$\begin{aligned}\omega_x(\mu) = \frac{1}{\Delta_p\Delta_t}\operatorname{arccsc}\Biggl(&\frac{R}{\Delta x(\mu)}\cos\Biggl(\arcsin\Biggl(\\ &\frac{\Theta_y}{M-1}\frac{1}{R}\left(\left\lfloor\frac{M}{2}\right\rfloor - \mu\right)\Biggr) + \theta_0\Biggr)\Biggr)\end{aligned} \tag{4.10}$$

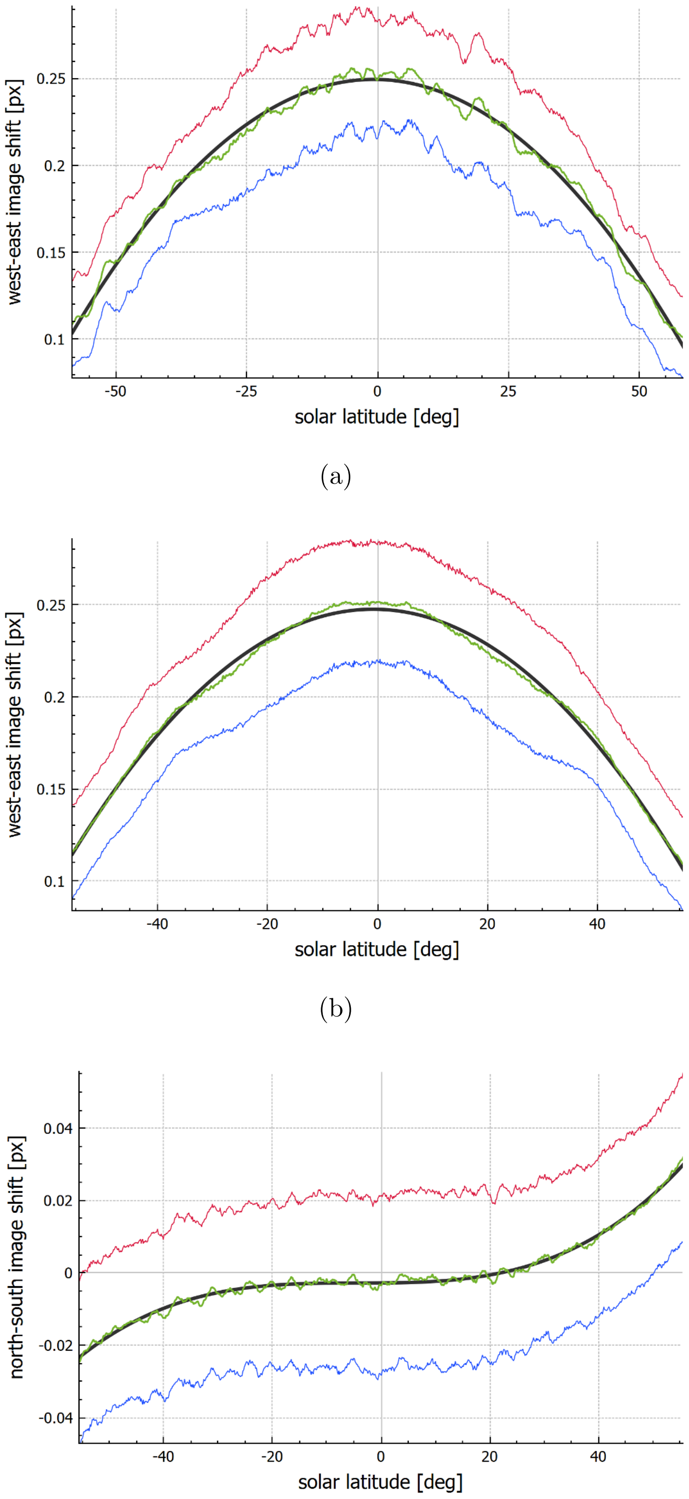
for the west-east solar angular velocity, and

$$\begin{aligned}\omega_y(\mu) = \frac{1}{\Delta_p\Delta_t}\Biggl(&\arcsin\left(\frac{\Delta_y\left(\left\lfloor\frac{M}{2}\right\rfloor - \mu\right)}{R}\right) + \theta_0 - \\ &\arcsin\left(\sin\left(\arcsin\left(\frac{\Delta_y\left(\left\lfloor\frac{M}{2}\right\rfloor - \mu\right)}{R}\right) + \theta_0\right) - \frac{\Delta y(\mu)}{R}\right)\Biggr)\end{aligned} \tag{4.11}$$

for the north-south solar angular velocity.

4.2.8 Data sets used for the measurement

The west-east and north-south solar angular velocities $\omega_x(\theta)$ and $\omega_y(\theta)$ were measured for two sets of $P = 1999$ pairs of consecutive images. The first data set uniformly spans the time interval from 1.1.2020 to 2.2.2020. This period approximately equals to one full solar revolution at the equator (where the flow is faster than average) and 80% of a revolution near the poles (where the flow is slower than average).

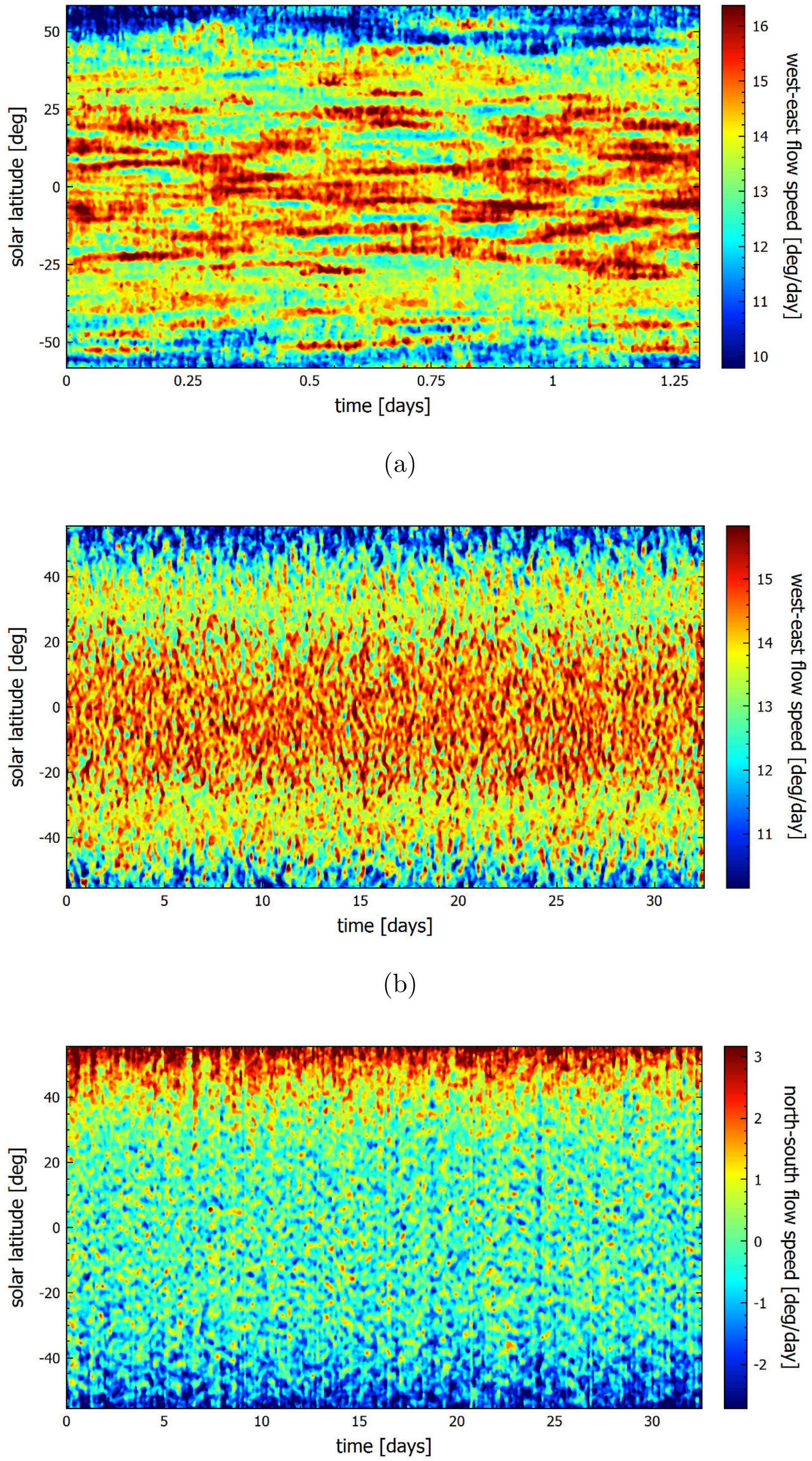


(a)

(b)

(c)

Figure 4.9: Average west-east and north-south image shift profiles measured on 2000 SDO/HMI continuum full-disk image pairs. First row: average west-east image shift profile $\overline{\Delta x(\theta)}$ (b) (green) obtained from images spanning approximately a one day period, together with its best least squares second degree polynomial fit curve (black) and one standard deviation band (blue, red). Second row: same as the first row for a one month period. Third row: same as the first row for north-south flow with a third degree polynomial fit curve.



(a)

(b)

(c)

Figure 4.10: Solar west-east and north-south flow speed maps obtained from orthographically backprojecting image shifts measured on 2000 SDO/HMI continuum full-disk image pairs back onto the solar surface. First row: west-east flow speed map (a) obtained from images spanning approximately a one day period. Second row: same as the first row for a one month period. Third row: same as the first row for north-south flow.

The second data set uniformly spans the time interval from 1.1.2020 to 2.1.2020. This period approximately equals to 4% of a solar revolution at the equator and 3% of a revolution near the poles.

The duration of both data set intervals (~ 1 day and ~ 1 month) were chosen deliberately to measure both the potentially interesting short-lived phenomena and also the overall rotational speed profile. The specific initial date of both data sets (1.1.2020) was chosen arbitrarily and is not in any way special - both image data sets show a very "calm" behaviour during this period, with almost no clearly discernible large scale features, such as sunspots, present.

The size of the central meridional sub-regions, from which the image shifts were calculated, was set to 256x256. This size proved to be large enough for good image registration quality, and at the same time small enough to provide good spatial flow speed resolution.

4.2.9 Flow speed maps

Since M sub-regions were equidistantly distributed among different latitudes on the central meridian of every picture and measurements for P picture pairs were made, very detailed 2D west-east and north-south central meridian flow speed maps with $P \times M$ resolution could be constructed for each data set. These maps can be seen in Fig. 4.10.

Although the measured angular velocity clearly varies with time in a given latitude band, the temporally coherent angular velocity values in the flow speed maps demonstrate the good image registration quality. Measured from horizontally (temporally) adjacent image shifts, although being computed from entirely different pairs of images, they are very similar at each latitude, which is demonstrated by the existence of the clearly distinguishable coherent faster than average and slower than average flows in Fig. 4.10 (a).

The west-east and north-south image shift $\Delta x(\theta)$, $\Delta y(\theta)$ and angular velocity $\omega_x(\theta)$, $\omega_y(\theta)$ values from all the P image pairs were also averaged at each latitude, resulting in picture-pair averages $\overline{\Delta x(\theta)}$, $\overline{\Delta y(\theta)}$ (see Sections 4.2.10 and 4.2.11) and $\overline{\omega_x(\theta)}$, $\overline{\omega_y(\theta)}$ (see Section 4.2.12). The rather smooth characteristics of the average west-east and north-south image shift profiles $\overline{\Delta x(\theta)}$ and $\overline{\Delta y(\theta)}$ (see Fig. 4.9) can serve as good indicators of the entire rotational speed measurement quality.

4.2.10 West-east image shift profile

Upon closer inspection of the average west-east image shift profiles obtained from both data sets, very coherent curves with very little noise can be seen. The profile obtained from the one day data set (Fig. 4.9 (a)) is somewhat more chaotic than that of the one month data set (Fig. 4.9 (b)), which is expected, as the profile captures local flows and events happening on time scale comparable to the entire data set time span. These events can be seen very clearly as reddish and blueish streaks in the flow speed maps of both data sets, showing regions of faster than average or slower than average flow, lasting approximately 8 hours. The shorter time span data set flow speed map shows individual flows in more detail, while the longer time span flow speed map demonstrates the practically constant duration and presence of these flows. The profile obtained from the longer time span data set is almost perfectly smooth, displaying the overall global average west-east rotational speed profile.

4.2.11 North-south image shift profile

Although being roughly one order of magnitude less significant than its west-east counterpart (~ 0.01px compared to ~ 0.2px), the measurement result of the average north-south image shift profile $\overline{\Delta y(\theta)}$ is also quite interesting. The north-south profile clearly shows more noisy behaviour, which can be caused by being near the accuracy limit of the image registration precision. In the north-south part of the registration, the iterative phase correlation method is estimating north-south shifts around 0.01 pixels between two 256×256 images, and such shifts are very hard to estimate with higher precision. However, a very clear overall trend can still be seen in the average profile (see Fig. 4.9 (c)).

In the northern solar hemisphere, the flow seems to be southwards, in addition to the natural (∼20× faster) eastwards rotational flow. This southwards part of the flow gets more significant closer to the north pole, roughly following a cubic (or arcsine) trend. The north-south image shift profile appears to be very close to being central-symmetric, meaning that the flow appears to be southwards in the northern hemisphere and northwards in the southern hemisphere, resulting in an overall flow towards the equator in both hemispheres, which speeds up significantly near the poles.

4.2.12 The final flow speed profiles

According to multiple studies [Zhang et al., 2015, Lamb, 2017], the southern solar hemisphere seems to be rotating (slightly, but noticeably) faster, than the northern hemisphere. The usual west-east flow trigonometric fit (denoted $\omega^{\dagger}(\theta)$) of the form

$$\omega^{\dagger}(\theta) = A + B\sin^2(\theta) + C\sin^4(\theta) \tag{4.12}$$

is always zero-symmetric, since even powers of the function $\sin(\theta)$ are always even, and thus cannot capture such north-south asymmetric behaviour. This results in unnecessarily large curve fitting errors if the flow speed profile is not separated into northern and southern parts. On the contrary, polynomial fit of a second degree (denoted $\omega^*_x(\theta)$) in the simple form of

$$\omega^*_x(\theta) = a + b\theta + c\theta^2 \tag{4.13}$$

can capture any northern-southern hemisphere asymmetries with much less error and with the same amount of fitting parameters. This fit is thus included in all the image shift and flow speed profile figures, where the flow speed is not separated into northern and southern parts.

As the average north-south image shift profile clearly trends towards a cubic, it is only natural to fit the resulting north-south flow speeds with a polynomial of a third degree in the simple form of

$$\omega^*_y(\theta) = a + b\theta + c\theta^2 + d\theta^3. \tag{4.14}$$

Flow speed curve type	**A**	**B**	**C**
W-E (B) (1 m.)	14.21	-1.61	-2.89
W-E (B) (1 d.)	14.27	-1.46	-2.66
W-E (N) (1 m.)	14.20	-1.93	-2.75
W-E (N) (1 d.)	14.26	-1.24	-3.47
W-E (S) (1 m.)	14.23	-1.28	-3.02
W-E (S) (1 d.)	14.28	-1.68	-1.84
N-S (N) (1 m.)	0.01	-0.82	8.18
N-S (N) (1 d.)	-0.01	-0.62	7.40
N-S (S) (1 m.)	-0.14	0.61	-5.84
N-S (S) (1 d.)	-0.27	1.29	-7.57

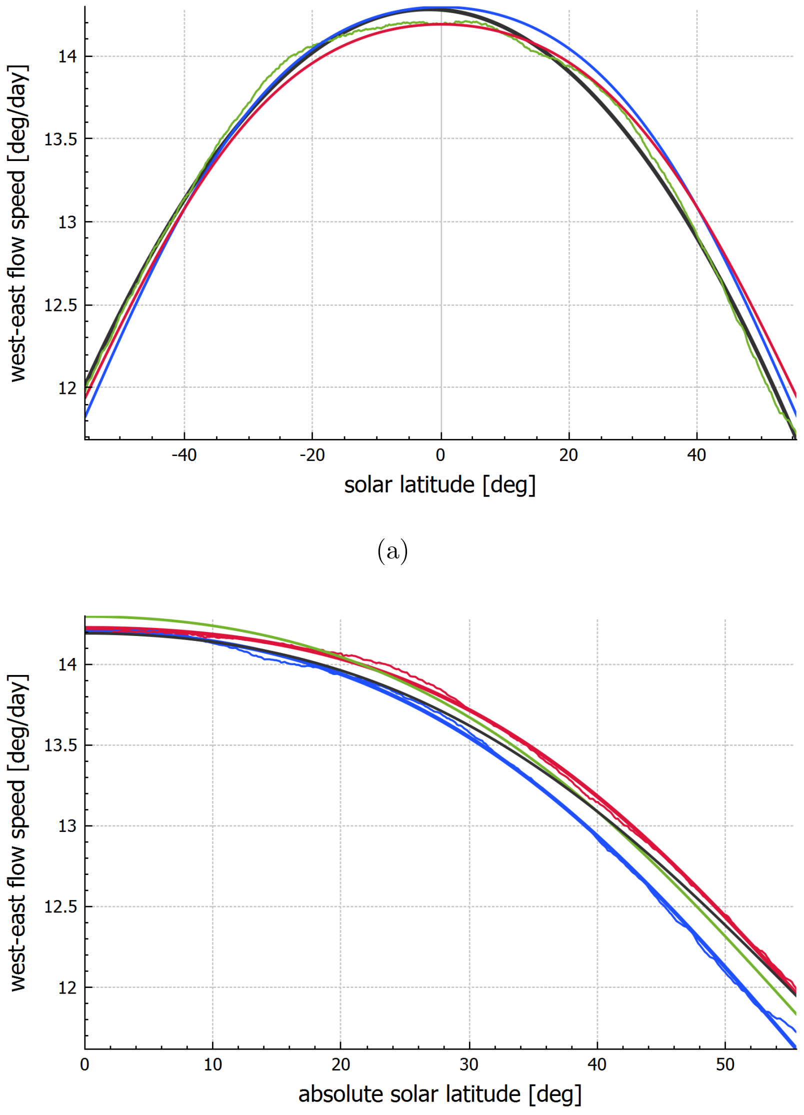
Table 4.1: Best (least squares) fitting coefficients A, B, C for different types of west-east (W-E) and north-south (N-S) trigonometric flow speed profile fits for northern (N), southern (S) or both (B) hemispheres, obtained from both one month (1 m.) and one day (1 d.) data sets in degrees per day according to Equation 4.12.

Flow speed curve type	**a**	**b**	**c**	**d**
W-E (1 m.)	14.28	-0.17	-2.58	-
W-E (1 d.)	14.34	-0.14	-2.38	-
N-S (1 m.)	-0.11	-0.18	0.47	3.31
N-S (1 d.)	-0.17	-0.23	0.20	3.42

Table 4.2: Best (least squares) fitting coefficients a, b, c, d for different types of west-east (W-E) and north-south (N-S) polynomial flow speed profile fits (both hemispheres together), obtained from both one month (1 m.) and one day (1 d.) data sets in degrees per day according to Equations 4.13 and 4.14.

The values of the fitting parameters A, B, C and a, b, c, d for all mentioned flow profiles in degrees per day for northern, southern and both hemispheres together are provided in Tables 4.1 and 4.2. The final average west-east and north-south flow speed profiles, together with their corresponding polynomial fits (and together with profiles inferred by different techniques, for comparison, in the west-east case) can be seen in Fig. 4.11 (a) and 4.12 (a).

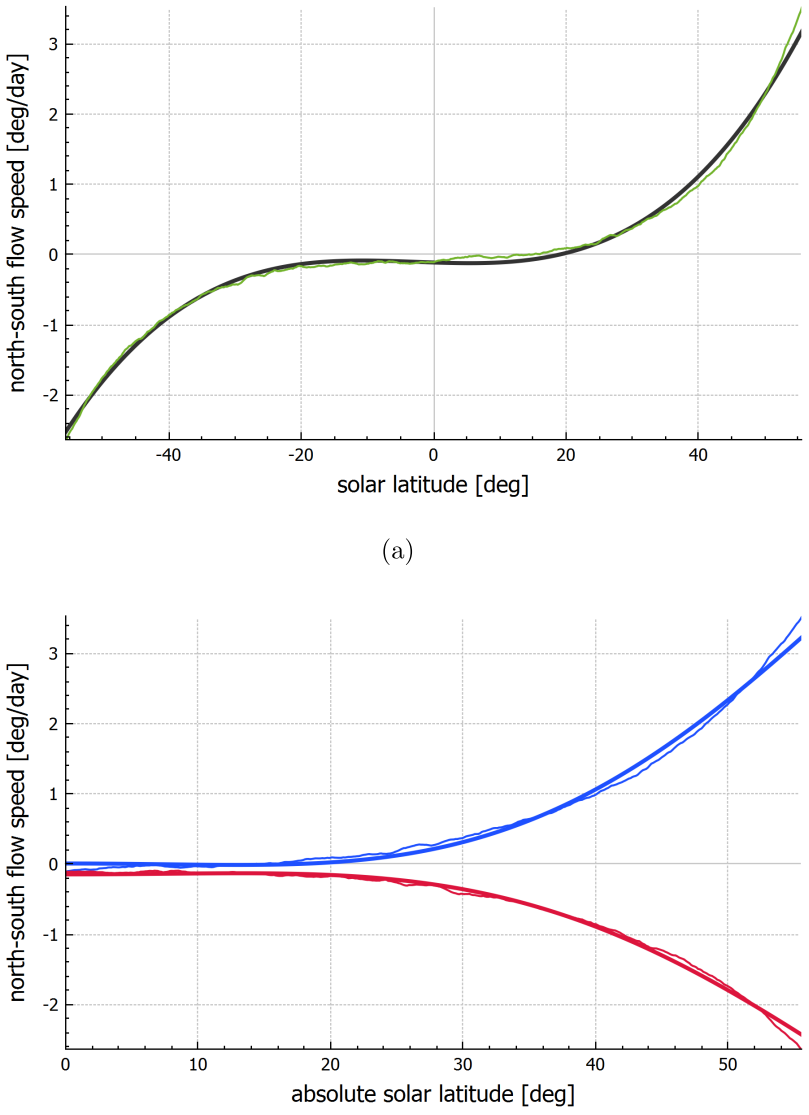
Furthermore, for a better visualization of the North-South asymmetry of the resulting west-east and north-south flows, Fig. 4.11 (b) and 4.12 (b) show the average flow profiles and their corresponding best least squares trigonometric fits (according to Equation 4.12) separately for each hemisphere, with absolute latitude on the horizontal axis.



(a)

(b)

Figure 4.11: Average measured west-east flow speed profile $\overline{\omega_x(\theta)}$ at different latitudes in degrees per day. Measured from a one month period from 1.1.2020 to 2.2.2020. The flow speed profile was obtained by temporally averaging values at each latitude in the corresponding flow speed map (Fig. 4.10 (b)). Figure (a) contains the entire average west-east profile (green) with its corresponding best least squares second degree polynomial fit $\omega_x^*(\theta)$ (black). Furthermore, two sample west-east trigonometric flow speed profiles inferred by different techniques [Lamb, 2017, Howard et al., 1983] are also included (blue, red) in (a). Figure (b) contains the average measured west-east flow speed profile with its corresponding best least squares trigonometric fit curves $\omega_x^\dagger(\theta)$ for northern (blue) and southern (red) hemisphere separately at different absolute latitudes, together with the two sample west-east trigonometric flow speed profiles inferred by different techniques (green, black) in (b).



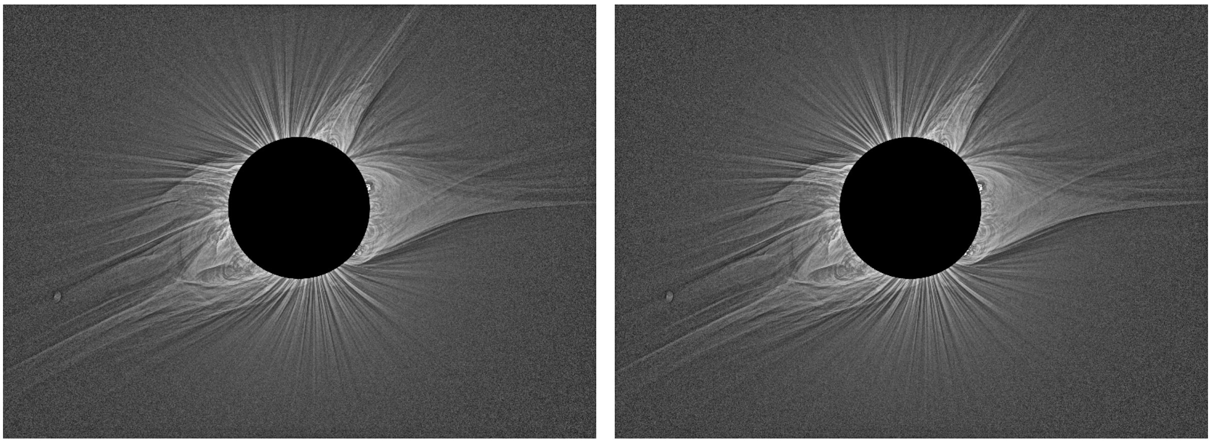
(a)

(b)

Figure 4.12: Average measured north-south flow speed profile $\overline{\omega_y(\theta)}$ at different latitudes in degrees per day. Measured from a one month period from 1.1.2020 to 2.2.2020. The flow speed profile was obtained by temporally averaging values at each latitude in the corresponding flow speed map (Fig. 4.10 (c)). Figure (a) contains the entire average north-south profile (green) with its corresponding best least squares third degree polynomial fit $\omega_y^*(\theta)$ (black). Figure (b) contains the average measured north-south flow speed profile with its corresponding best least squares trigonometric fit curves $\omega_y^\dagger(\theta)$ for northern (blue) and southern (red) hemisphere separately at different absolute latitudes.

4.3 Solar wind speed measurement

The measurement of solar wind speed from processed contrast-enhanced composite solar eclipse images [Druckmüller et al., 2006, Druckmüller, 2009] (see Fig. 4.13) is a good example of an image registration task with non-uniform motion. The composite solar eclipse images contain both relatively stationary regions (e.g. coronal loops) and regions moving in different directions with different speeds (e.g. solar wind, solar flares, the Moon). This means, that (global) intensity-based (correlation) methods are not well suited for this task, and (local) feature-based methods need to be used instead. Furthermore, due to the contrast-enhancing process, the images often contain regions with very low average signal-to-noise ratio. The feature detector/descriptor pair thus needs to be very robust and resilient to noise.



(a) (b)

Figure 4.13: A pair of processed composite solar eclipse images with $\Delta t = 11.8$s captured during the 2017 total solar eclipse near Mitchell, Oregon.

Arrow index	**Solar wind speed [km/s]**
1	576
2	617
3	628
4	669
5	715
6	715
7	730
8	742
9	800

(a)

Arrow index	**Solar wind speed [km/s]**
1	496
2	545
3	589
4	618
5	671
6	709
7	749
8	835

(b)

Table 4.3: Measured solar wind speeds corresponding to the color-coded arrows depicted in Fig. 4.14 - Mitchell, Oregon observing site (a) and Fig. 4.15 - Alliance, Nebraska observing site (b).

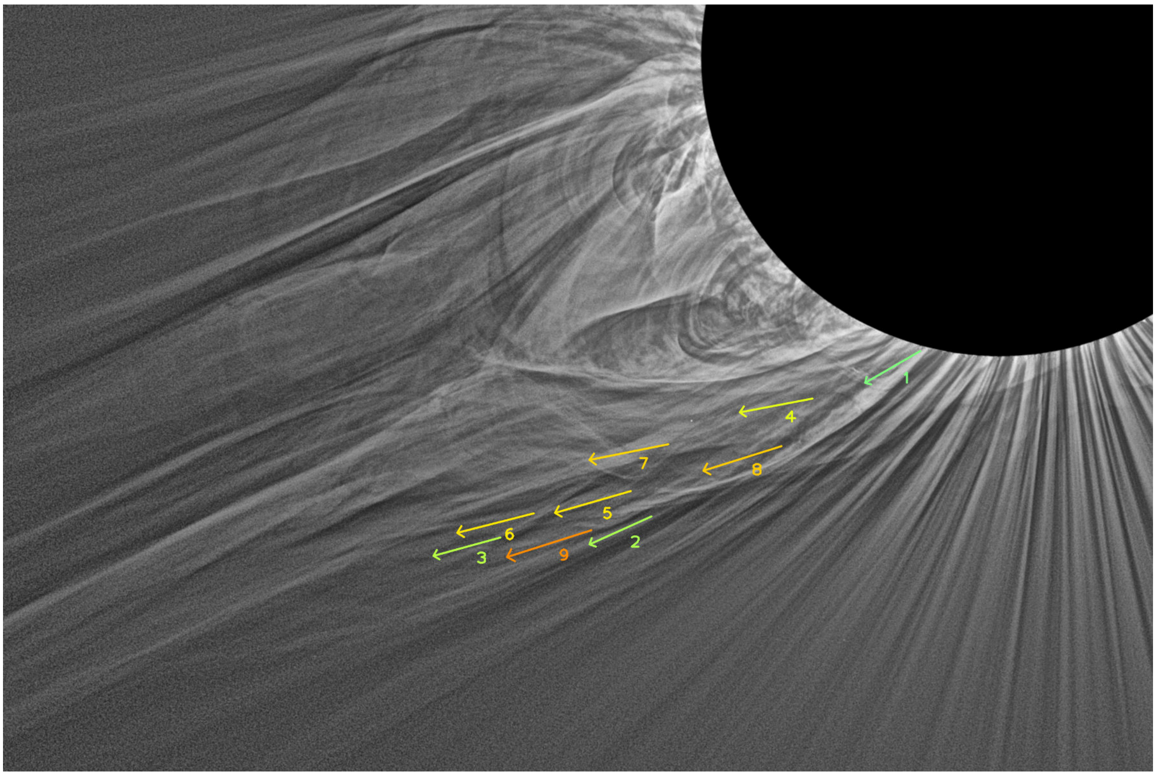


Figure 4.14: Solar wind speeds measured by the U-SURF method on sets of 10 processed composite solar eclipse images. The depicted arrows originate from the corresponding automatically detected U-SURF keypoints and are scaled and color-coded according to the relative measured solar wind speed - from blue (relatively slow) to red (relatively fast). The actual corresponding measured solar wind speeds can be found in Table 4.3 (a). Captured during the 2017 total solar eclipse at Mitchell, Oregon observing site.

The Speeded up robust features (SURF) detector/descriptor (described in Section 2.2) is a very good fit for this difficult task. Furthermore, a special version of the SURF algorithm, called "upright" SURF (or U-SURF), which is not invariant to image rotation, can be used in this case, since the contrast-enhanced composite solar eclipse images do not contain any significantly scaled or rotated regions. This improves the algorithm computational performance, but more importantly in this case, also further increases the overall robustness of the results.

To detect the SURF points of interest (keypoints) in all of the contrast-enhanced solar eclipse images, a threshold Hessian determinant value of 100 was chosen. This value proved to be a good compromise between detecting many low quality features and detecting too few features. To describe the neighborhood of the detected keypoints, the U-SURF (not rotationally invariant) version of the SURF descriptor was used. Detected keypoints were matched by a brute-force L^2 similarity metric comparison of the U-SURF descriptors. The resulting matches were then filtered by Lowe's ratio test (as described in Section 2.2.4) with a strict threshold ratio of 0.4. Overlapping matches (matches, whose initial keypoints were less than 40 pixels apart) were filtered out to improve overall clarity, always in favor of the match corresponding to a higher solar wind speed.

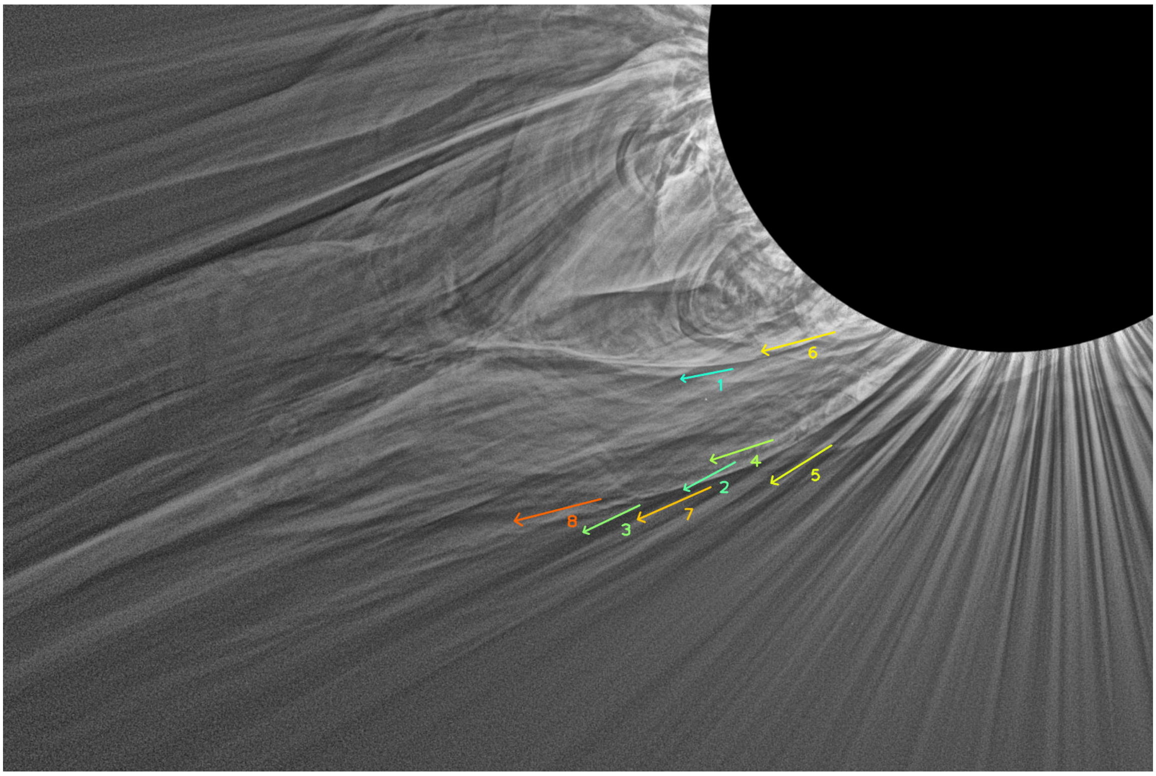


Figure 4.15: Solar wind speeds measured by the U-SURF method on sets of 10 processed composite solar eclipse images. The depicted arrows originate from the corresponding automatically detected U-SURF keypoints and are scaled and color-coded according to the relative measured solar wind speed - from blue (relatively slow) to red (relatively fast). The actual corresponding measured solar wind speeds can be found in Table 4.3 (b). Captured during the 2017 total solar eclipse at Alliance, Nebraska observing site.

Chapter 5

Summary and conclusions

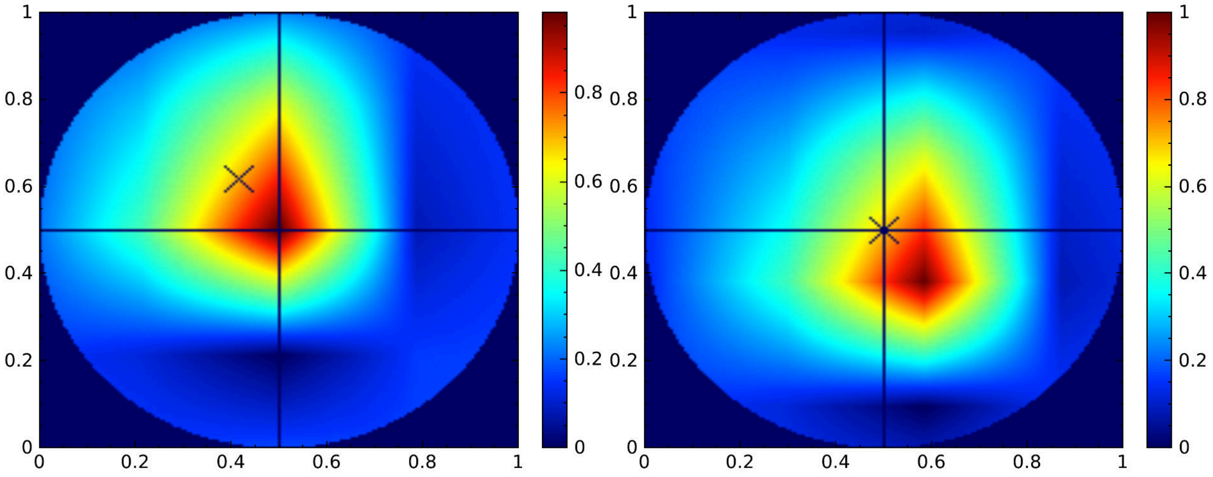
The first main goal of this thesis was to develop a novel state of the art high precision image registration method focusing on sub-pixel registration accuracy. The new Iterative Phase Correlation (IPC) method is based on a reliable pixel-level accuracy phase correlation method. The IPC method is a major non-trivial extension of the standard phase correlation method, including procedures like image windowing, adjustable cross-power spectrum filtering, correlation upsampling, weighted correlation centroid computation and subsequent iterative centroid refinement.

Many other sub-pixel cross-correlation and phase correlation extensions were previously developed, usually exploiting a particular idea to achieve a sub-pixel accuracy on the order of 0.1px. These ideas include image upsampling [Debella-Gilo and Kääb, 2011], fitting the correlation peak with an analytical function [Heid and Kääb, 2012, Abdou, 1998, Foroosh et al., 2002], calculating the centroid of the correlation peak [Michel and Rignot, 1999], counting the fractional number of cross-power spectrum phase cycles [Balci and Foroosh, 2006] and cross-power spectrum upsampling [Alba et al., 2015, Young and Driggers, 2006, Zhang et al., 2011]. The IPC method combines multiple of these ideas among with iterative accuracy refinement into a single reliable and efficient method regularly achieving sub-pixel accuracy on the order of 0.01px.

The iterative centroid refinement step, which is the main source of high sub-pixel precision of the IPC method, is a very intuitive process, since it iteratively moves in the direction of higher average correlation. The great potential of the iterative centroid refinement process is neatly demonstrated on an artificially created image pair in Fig. 5.1, where the upsampled correlation position corresponding to the true underlying sub-pixel image shift is marked. Although not having any prior information about the underlying sub-pixel shift or its corresponding upsampled correlation location, the iterative centroid refinement process converges to a point almost perfectly spot-on the ideal location, reaching accuracy on the order of 0.001px.

Furthermore, all the parameters of the IPC method can be easily optimized to maximize sub-pixel accuracy for a given image dataset. The main IPC parameters affecting image windowing, cross-power spectrum filtering, correlation upsampling and iterative centroid refinement gradually adapt to a specific kind of image during a differential evolution optimization process, optimizing a sophisticated non-convex non-continuous average sub-pixel accuracy metric. The IPC optimization process does not only improve overall average sub-pixel error (registration accuracy), but also overall sub-pixel error standard deviation (registration robustness).

Additionally, since the IPC method is an intensity-based method, it is computationally



(a) L^1 before refinement

(b) L^1 after refinement

Figure 5.1: Demonstration of the very high potential sub-pixel accuracy of the iterative phase correlation method obtained during the iterative centroid refinement process on a pair of artificially shifted images. Upsampled correlation location corresponding to the underlying "true" sub-pixel shift is marked with a black "**x**". Before the iterative refinement process (a), the image shift estimate is $[-4, 5]$ px, which amounts to an error of $[0.2909, 0.5872]$ px. After the iterative refinement process (b), the sub-pixel image shift estimate is $[-4.2928, 4.5799]$ px, which amounts to an error of $[-0.001947, -0.007315]$ px.

very efficient, mainly due to the use of the Fast Fourier Transform algorithm to compute the phase correlation. During each iteration in the iterative refinement process, only the centroid of a small correlation neighborhood needs to be recomputed, and thus the iterative character of the method does not significantly affect the overall computational efficiency when compared to the standard non-iterative cross-correlation and phase correlation methods. The IPC method can thus also serve as a good replacement of standard correlation methods in resource-constrained applications.

The second main goal of this work was to find novel applications of state of the art high precision image registration methods (including the newly developed IPC method), mainly focused on astrophysical measurements and observations.

The first application of high precision image registration methods researched and described in this work is the alignment of images with low levels of similarity. A good example of such image pairs are the SDO/AIA 94Å, 131Å, 171Å, 211Å, 304Å and 335Å images. Even though these images contain information about vastly different phenomena (e.g. flaring/active regions, chromosphere, corona), the IPC general image alignment method is able to align them reliably. This includes the correct estimation and correction of a combination of image scale, image rotation and image translation.

Another major astrophysical application of the IPC algorithm is the novel iterative phase correlation technique of solar differential rotation estimation. This technique offers a new method for the empirical determination of the differential rotation rate of the solar photosphere. It is based on the measurement of locally varying image shifts between consecutive (Δt = 45s) solar images. The technique was applied to SDO/HMI data and roughly covered one Carrington rotation of continuous observations from 1.1.2020 to 2.2.2020. The technique has a number of exceptional properties which make it superior to other correlation-based techniques, mainly in the domain of high spatial and temporal resolution.

The local shift measurement between two images is not limited to contrasty features like sunspots, as it can be applied even to very low contrast structures, such as granulation and/or faculae. Hence, the differential rotation rate thus determined is independent of any a priori selection of solar features, and can be successfully applied to any location in the photosphere. Unlike magnetic feature tracking methods, the IPC differential rotation measurement method does not introduce systematic measurement biases arising from the fact that the rotational velocities of various types of magnetic features do not necessarily correspond to the underlying rotation rate of the photosphere and further depend significantly on feature size, morphology and age [Ward, 1966, Gilman and Howard, 1984, Zappala and Zuccarello, 1991].

The unprecedented $\Delta t = 45$s time step resulted in the most precise and reliable sub-pixel image shift measurements, as larger time steps correspond to significant non-uniform granulation changes, for which a sub-pixel image shift can not be measured reliably. The extreme temporal resolution of the IPC method of differential rotation measurement enables the study of short-period photospheric phenomena and the comparison of measurements on different days separated by an event that may have a global impact on the solar magnetic field, such as a solar flare or a coronal mass ejection.

Furthermore, given that the technique does not rely on visible features, it can be used throughout a full solar cycle independently of the presence of specific features in the photosphere. While methods based on solar markers yield a limited number of data points, the iterative phase correlation technique is applicable to every pixel close to central meridian within a given latitude band. This yields a unique opportunity to study the rotation rate of the Sun locally.

The method is also able to estimate the locally varying north-south shifts between consecutive solar images. Although being very small and hard to measure, the north-south image shifts obtained by the novel image registration method show coherent results, displaying a consistent global trend. The reliability of the north-south image shift measurement allows further studies of the general (not only rotational) photospheric movements with great detail. The north-south flow speed results show an overall flow towards the solar equator, with speeds becoming more significant closer to the geographic poles. Both the west-east and north-south flow speed profiles were fitted with polynomial and trigonometric curves, whose best (least squares) parameters were calculated and compared with other methods.

The reliability of the approach is underscored by the fact that the average sidereal differential rotation curve obtained is in the middle of other published results. At the same time it also clearly shows the now well established N-S rotational speed asymmetry. Furthermore, the quality of the results is further supported by the smoothness of the average west-east image shift profile obtained from a single Carrington rotation of SDO/HMI image data and also by the coherence of the measured flow speed values inferred from temporally adjacent images. As an added bonus, the novel technique described in this work allows the study of short-lived and previously undetected phenomena associated with the solar magnetic field.

Intensity-based image registration methods are not suited for all astrophysical applications. In some measurements, temporally adjacent images contain multiple regions moving in significantly different directions and speeds. For such applications, feature-based image registration methods are more appropriate. One such astrophysical application of feature-based image registration methods described in this work is the measurement of the solar wind speed from composite solar eclipse images. In this approach, the solar wind speed is estimated from regional shifts obtained by the SURF feature detector and descriptor on multiple pairs of processed contrast-enhanced composite solar eclipse images. The region locations correspond to the automatically detected feature keypoints. To improve the robustness of this technique, only features with very high Hessian threshold are considered. Furthermore, feature pairs not passing Lowe's ratio test with a strict L^2 similarity ratio of 0.4 are discarded. The technique of feature-based image registration of pairs of contrast-enhanced composite solar eclipse images offers a novel way of solar wind speed measurement at various distances from the solar surface.

In conclusion, the high sub-pixel accuracy of the newly developed IPC image registration method is not only able to significantly improve the accuracy of some existing astrophysical measurements based on image registration techniques, but also allow novel measurements and observations in domains where image registration (or any other) techniques were previously not applicable.

Published articles

The following articles relating to this dissertation have been published:

- [Hrazdíra et al., 2020a] Iterative phase correlation algorithm for high-precision subpixel image registration. *The Astrophysical Journal Supplement Series*, 247(1):8, 2020a.

  Article citations (excluding self-citations), as of February 6, 2023:

  - [Croft, 2021] Direct geometrical measurement of the Hubble constant from galaxy parallax: predictions for the Vera C. Rubin Observatory and Nancy Grace Roman Space Telescope. *Monthly Notices of the Royal Astronomical Society*, 501(2), pp.2688-2703.
  - [Wu et al., 2021] Details-preserving multi-exposure image fusion based on dual-pyramid using improved exposure evaluation. *Results in Optics*, 2, p.100046.
  - [Rasmy et al., 2021] Automatic sub-pixel co-registration of remote sensing images using phase correlation and Harris detector. *Remote Sensing*, 13(12), p.2314.
  - [Balluff et al., 2021] An overview of image registration for aligning mass spectrometry imaging with clinically relevant imaging modalities. *Journal of Mass Spectrometry and Advances in the Clinical lab.*
  - [Liang et al., 2022] High-precision Multichannel Solar Image Registration Using Image Intensity. *The Astrophysical Journal Supplement Series*, 261(2), p.10.
  - [Kratz et al., 2022] ISM-assisted tomographic STED microscopy. *Optics Express*, 30(2), pp.939-956.

- [Hrazdíra et al., 2020b] Measuring solar differential rotation with an iterative phase correlation method. *The Astrophysical Journal Supplement Series*, 252(1):6, 2020b.

  Article citations (excluding self-citations), as of February 6, 2023:

  - [Liang et al., 2022] High-precision Multichannel Solar Image Registration Using Image Intensity. *The Astrophysical Journal Supplement Series*, 261(2), p.10.
  - [Edwards et al., 2022] A Solar-cycle Study of Coronal Rotation: Large Variations, Rapid Changes, and Implications for Solar-wind Models. *The Astrophysical Journal*, 928(1), p.42.
  - [Silva, 2021] Investigating Magnetic Energy Injection in Solar Active Regions. University of Northumbria at Newcastle (United Kingdom).

Bibliography

I. E. Abdou. Practical approach to the registration of multiple frames of video images. In *Visual Communications and Image Processing'99*, volume 3653, pages 371–382. SPIE, 1998.

E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. Pyramid methods in image processing. *RCA engineer*, 29(6):33–41, 1984.

A. Alahi, R. Ortiz, and P. Vandergheynst. Freak: Fast retina keypoint. In *2012 IEEE conference on computer vision and pattern recognition*, pages 510–517. Ieee, 2012.

A. Alba, J. F. Vigueras-Gomez, E. R. Arce-Santana, and R. M. Aguilar-Ponce. Phase correlation with sub-pixel accuracy: A comparative study in 1d and 2d. *Computer Vision and Image Understanding*, 137:76–87, 2015.

P. F. Alcantarilla, J. Nuevo, and A. Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell*, 34(7):1281–1298, 2011.

P. F. Alcantarilla, A. Bartoli, and A. J. Davison. Kaze features. In *European conference on computer vision*, pages 214–227. Springer, 2012.

V. Argyriou and T. Vlachos. A study of sub-pixel motion estimation using phase correlation. In *BMVC*, pages 387–396. Citeseer, 2006.

H. W. Babcock and H. D. Babcock. The sun's magnetic field, 1952-1954. *The Astrophysical Journal*, 121:349, 1955.

M. Balci and H. Foroosh. Subpixel registration directly from the phase difference. *EURASIP Journal on Advances in Signal Processing*, 2006:1–11, 2006.

B. Balluff, R. M. Heeren, and A. M. Race. An overview of image registration for aligning mass spectrometry imaging with clinically relevant imaging modalities. *Journal of Mass Spectrometry and Advances in the Clinical lab*, 2021.

H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

J. G. Beck. A comparison of differential rotation measurements–(invited review). *Solar physics*, 191(1):47–70, 2000.

P. Bing, X. Hui-Min, X. Bo-Qin, and D. Fu-Long. Performance of sub-pixel registration algorithms in digital image correlation. *Measurement Science and Technology*, 17(6): 1615, 2006.

R. Blackman and J. Tukey. The measurement of power spectra dover publications. *Inc, New York*, 1958.

R. N. Bracewell and R. N. Bracewell. *The Fourier transform and its applications*, volume 31999. McGraw-Hill New York, 1986.

G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

E. O. Brigham and R. E. Morrow. The fast fourier transform. *IEEE Spectrum*, 4(12): 63–70, 1967. doi: 10.1109/MSPEC.1967.5217220.

M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

A. Casella, I. De Falco, A. Della Cioppa, U. Scafuri, and E. Tarantino. Exploiting multicore and gpu hardware to speed up the registration of range images by means of differential evolution. *Journal of Parallel and Distributed Computing*, 133:307–318, 2019.

J. Chen, K. Ji, H. Deng, and S. Feng. Analysis of correlation registration algorithms in the observation of solar magnetic field. In *Unifying Electrical Engineering and Electronics Engineering*, pages 1241–1249. Springer, 2014.

R. A. Croft. Direct geometrical measurement of the hubble constant from galaxy parallax: predictions for the vera c. rubin observatory and nancy grace roman space telescope. *Monthly Notices of the Royal Astronomical Society*, 501(2):2688–2703, 2021.

S. Das and P. N. Suganthan. Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, 15(1):4–31, 2010.

M. Debella-Gilo and A. Kääb. Sub-pixel precision image matching for measuring surface displacements on mass movements using normalized cross-correlation. *Remote Sensing of Environment*, 115(1):130–142, 2011.

H. N. Do, T.-J. Chin, N. Moretti, M. K. Jah, and M. Tetlow. Robust foreground segmentation and image registration for optical detection of geo objects. *Advances in Space Research*, 2019.

M. Druckmüller. Phase correlation method for the alignment of total solar eclipse images. *The Astrophysical Journal*, 706(2):1605, 2009.

M. Druckmüller, V. Rušin, and M. Minarovjech. A new numerical method of total solar eclipse photography processing. *Contrib. Astron. Obs. Skalnaté Pleso*, 36:131–148, 2006.

L. Edwards, D. Kuridze, T. Williams, and H. Morgan. A solar-cycle study of coronal rotation: Large variations, rapid changes, and implications for solar-wind models. *The Astrophysical Journal*, 928(1):42, 2022.

W. M. Elsasser. The earth's interior and geomagnetism. *Reviews of Modern Physics*, 22 (1):1, 1950.

H. Foroosh, J. B. Zerubia, and M. Berthod. Extension of phase correlation to subpixel registration. *IEEE transactions on image processing*, 11(3):188–200, 2002.

M. Frigo and S. G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on "Program Generation, Optimization, and Platform Adaptation".

P. Gilman and R. Howard. Variations in solar rotation with the sunspot cycle. *The Astrophysical Journal*, 283:385–391, 1984.

P. Gravel, J. Verhaeghe, and A. J. Reader. 3d pet image reconstruction including both motion correction and registration directly into an mr or stereotaxic spatial atlas. *Physics in Medicine & Biology*, 58(1):105, 2012.

D. M. Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

T. Heid and A. Kääb. Evaluation of existing image matching methods for deriving glacier surface displacements globally from optical satellite imagery. *Remote Sensing of Environment*, 118:339–355, 2012.

R. Howard, J. Adkins, J. Boyden, T. Cragg, T. Gregory, B. LaBonte, S. Padilla, and L. Webster. Solar rotation results at mount wilson. *Solar Physics*, 83(2):321–338, 1983.

Z. Hrazdíra, M. Druckmüller, and S. Habbal. Iterative phase correlation algorithm for high-precision subpixel image registration. *The Astrophysical Journal Supplement Series*, 247(1):8, 2020a.

Z. Hrazdíra, M. Druckmüller, and S. Habbal. Measuring solar differential rotation with an iterative phase correlation method. *The Astrophysical Journal Supplement Series*, 252(1):6, 2020b.

M. Jenkinson and S. Smith. A global optimisation method for robust affine registration of brain images. *Medical image analysis*, 5(2):143–156, 2001.

J. Kratz, C. Geisler, and A. Egner. Ism-assisted tomographic sted microscopy. *Optics Express*, 30(2):939–956, 2022.

D. A. Lamb. Measurements of solar differential rotation and meridional circulation from tracking of photospheric magnetic features. *The Astrophysical Journal*, 836(1):10, 2017.

M. Lefébure and L. D. Cohen. Image registration, optical flow and local rigidity. *Journal of Mathematical Imaging and Vision*, 14(2):131–147, 2001.

J. W. Leibacher, R. W. Noyes, J. Toomre, and R. K. Ulrich. Helioseismology. *Scientific American*, 253(3):48–57, 1985.

R. B. Leighton, R. W. Noyes, and G. W. Simon. Velocity fields in the solar atmosphere. i. preliminary report. *The Astrophysical Journal*, 135:474, 1962.

J. R. Lemen, D. J. Akin, P. F. Boerner, C. Chou, J. F. Drake, D. W. Duncan, C. G. Edwards, F. M. Friedlaender, G. F. Heyman, N. E. Hurlburt, et al. The atmospheric imaging assembly (aia) on the solar dynamics observatory (sdo). In *The solar dynamics observatory*, pages 17–40. Springer, 2011.

C. Leng, H. Zhang, B. Li, G. Cai, Z. Pei, and L. He. Local feature descriptor for image matching: A survey. *IEEE Access*, 7:6424–6434, 2019.

S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *2011 International conference on computer vision*, pages 2548–2555. Ieee, 2011.

G. Levi and T. Hassner. Latch: learned arrangements of three patch codes. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016.

C. Li, B. Lu, Y. Zhang, H. Liu, and Y. Qu. 3d reconstruction of indoor scenes via image registration. *Neural Processing Letters*, 48(3):1281–1304, 2018.

S. Li, M. Peng, B. Zhang, X. Feng, and Y. Wu. Auto-registration of medium and high spatial resolution satellite images by integrating improved sift and spatial consistency constraints. *International Journal of Remote Sensing*, 40(14):5635–5650, 2019.

B. Liang, X. Chen, L. Yu, S. Feng, Y. Guo, W. Cao, W. Dai, Y. Yang, and D. Yuan. High-precision multichannel solar image registration using image intensity. *The Astrophysical Journal Supplement Series*, 261(2):10, 2022.

D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

M. T. Mahmood and I. H. Lee. Well-distributed feature extraction for image registration using histogram matching. *Applied Sciences*, 9(17):3487, 2019.

R. Michel and E. Rignot. Flow of glaciar moreno, argentina, from repeat-pass shuttle imaging radar images: comparison of the phase correlation method with radar interferometry. *Journal of Glaciology*, 45(149):93–100, 1999.

M. Pant, H. Zaheer, L. Garcia-Hernandez, A. Abraham, et al. Differential evolution: A review of more than two decades of research. *Engineering Applications of Artificial Intelligence*, 90:103479, 2020.

E. N. Parker. Hydromagnetic dynamo models. *The Astrophysical Journal*, 122:293, 1955.

K. V. Price. Differential evolution. In *Handbook of optimization*, pages 187–214. Springer, 2013.

L. Rasmy, I. Sebari, and M. Ettarid. Automatic sub-pixel co-registration of remote sensing images using phase correlation and harris detector. *Remote Sensing*, 13(12):2314, 2021.

E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

W. S. Sarle et al. Stopped training and other remedies for overfitting. *Computing science and statistics*, pages 352–360, 1996.

C. Scheiner. *Rosa Ursina Sive Sol*. 1630. book 4, part 2.

C. Shapiro, B. Rowe, T. Goodsall, C. Hirata, J. Fucik, J. Rhodes, S. Seshadri, and R. Smith. Weak gravitational lensing systematics from image combination. *Publications of the Astronomical Society of the Pacific*, 125(934):1496, 2013.

A. C. A. Silva. *Investigating Magnetic Energy Injection in Solar Active Regions*. University of Northumbria at Newcastle (United Kingdom), 2021.

A. J. Smith. Cache memories. *ACM Comput. Surv.*, 14(3):473–530, sep 1982. ISSN 0360-0300. doi: 10.1145/356887.356892. URL https://doi.org/10.1145/356887.356892.

K. Takita, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi. High-accuracy subpixel image registration based on phase-only correlation. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 86(8):1925–1934, 2003.

M. Thompson, J. Toomre, E. Anderson, H. Antia, G. Berthomieu, D. Burtonclay, S. Chitre, J. Christensen-Dalsgaard, T. Corbard, M. DeRosa, et al. Differential rotation and dynamics of the solar interior. *Science*, 272(5266):1300–1305, 1996.

E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE transactions on pattern analysis and machine intelligence*, 32(5): 815–830, 2009.

F. Ward. *Determination of the solar-rotation rate from the motion of identifiable features*. Number 226. Air Force Cambridge Research Laboratories, Office of Aerospace Research . . . , 1966.

L. Wu, J. Hu, C. Yuan, and Z. Shao. Details-preserving multi-exposure image fusion based on dual-pyramid using improved exposure evaluation. *Results in Optics*, 2:100046, 2021.

S. S. Young and R. G. Driggers. Superresolution image reconstruction from a sequence of aliased imagery. *Applied Optics*, 45(21):5073–5085, 2006.

R. Zappala and F. Zuccarello. Angular velocities of sunspot-groups and solar photospheric rotation. *Astronomy and Astrophysics*, 242:480–487, 1991.

L. Zhang, K. Mursula, and I. Usoskin. Solar surface rotation: Ns asymmetry and recent speed-up. *Astronomy & Astrophysics*, 575:L2, 2015.

X. Zhang, M. Abe, and M. Kawamata. Reduction of computational cost of poc-based methods for displacement estimation in old film sequences. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 94(7):1497–1504, 2011.

H. Zhou and Y. Yu. Applying rotation-invariant star descriptor to deep-sky image registration. *Frontiers of Computer Science*, 12(5):1013–1025, 2018.