Ocean Engineering & Oceanography 10

Luc Jaulin
Andrea Caiti
Marc Carreras
Vincent Creuze
Frédéric Plumet
Benoît Zerr
Annick Billon-Coat *Editors*

Marine Robotics and Applications

Springer

Ocean Engineering & Oceanography

Volume 10

More information about this series at http://www.springer.com/series/10524

Luc Jaulin · Andrea Caiti
Marc Carreras · Vincent Creuze
Frédéric Plumet · Benoît Zerr
Annick Billon-Coat
Editors

Marine Robotics and Applications

Springer

Editors
Luc Jaulin
LABSTICC
ENSTA Bretagne
Brest
France

Andrea Caiti
Department of Information Engineering, Centro di Ricerca Enrico Piaggio
University of Pisa
Pisa
Italy

Marc Carreras
Escola Politècnica Superior (UdG), Edifi, Computer Vision and Robotics Institute
University of Girona
Girona
Spain

Vincent Creuze
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM)
Montpellier
France

Frédéric Plumet
ISIR
University Pierre and Marie Curie
Paris
France

Benoît Zerr
Ocean Sensing and Mapping
ENSTA Bretagne
Brest
France

Annick Billon-Coat
Lab-STICC
ENSTA Bretagne
Brest
France

ISSN 2194-6396 ISSN 2194-640X (electronic)
Ocean Engineering & Oceanography
ISBN 978-3-319-70723-5 ISBN 978-3-319-70724-2 (eBook)
https://doi.org/10.1007/978-3-319-70724-2

Library of Congress Control Number: 2017957699

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Every 2 years, the MOQESM conference takes place in Brest (France), in the framework of the SeaTech Week Conference. MOQESM is the meeting point of specialists from two fields, namely coastal hydrography and marine robotics. This is a unique opportunity for industrials and academic scientists to present their respective works to each other. It often leads to very rich scientific exchanges during and after the presentations. Indeed, many hydrographic applications can be performed by AUVs (Autonomous Underwater Vehicles), and, reciprocally, many navigation methods can benefit from the advances in hydrography for localization purpose, target identification, or path planning. The conference is also open to more general marine robotic presentations in fields like design and operation. This book gives to the reader an overview of some of the topics addressed during the MOQESM 16 Conference (October 11–12, 2016). This book is more than the proceedings of the conference, as it includes also original papers that have not been presented during MOQESM, but have been selected by the editors. The topics covered by this book include the following:

1. Acoustics: sonar tracks registration
2. Localization and navigation: simultaneous localization and mapping, range only localization, interval analysis for trajectory estimation, and electric sense for navigation
3. Multi-vehicles methods: securing an area with AUVs and performing adaptive sampling with autonomous sailing boats
4. Design: optimization of propulsion systems, and design and control of an underwater vehicle.

If this is not already the case, we hope that this book will convince you of the interest of bringing coastal hydrography and marine robotics together and we hope that it will make you as enthusiastic as we are to participate to next MOQESM editions.

Montpellier, France
May 2017

Vincent Creuze

Contents

Chapter 1
Fast Fourier-Based Block-Matching Algorithm for Sonar Tracks Registration in a Multiresolution Framework

Florian Nicolas, Andreas Arnold-Bos, Isabelle Quidu and Benoît Zerr

Abstract In the underwater mine warfare context, potential threats are usually detected and classified by means of an Automatic Target Recognition (ATR) chain, especially in case of newly surveyed areas. However, if we can rely on a previously acquired sonar track, it is conceivable to directly compare such a track, said as reference, with a more recent one in order to detect seabed changes such as new objects lying on the seabed. To perform this change detection process, the very first step consists in geometrically aligning the reference and the repeated tracks. In this paper, we detail a block-matching approach using masked Fourier cross-correlation as a similarity metric, to carry out a fast elastic registration in a multi resolution framework. To improve the robustness of the algorithm, the resulting vector field is then filtered thanks to the navigation uncertainty, provided by the INS, along with an Inverse Distance Weighting estimate, to get rid of outliers.

1.1 Introduction

In the context of sonar imagery, image registration, consisting in geometrically aligning two or more images, is a crucial step as it is often the very first step to further perform several tasks such as navigation correction [1–3], seabed mosaicking

The original version of this chapter was revised: For detailed information please see Erratum. The erratum to this chapter is available at https://doi.org/10.1007/978-3-319-70724-2_10

F. Nicolas (✉) · A. Arnold-Bos
Thales Underwater Systems, Brest, France
e-mail: florian.nicolas@fr.thalesgroup.com; florian.nicolas@ensta-bretagne.org

A. Arnold-Bos
e-mail: andreas.arnold@fr.thalesgroup.com

F. Nicolas · I. Quidu · B. Zerr
Lab-STICC/PRASYS, UMR CNRS 6285, ENSTA Bretagne, Brest, France
e-mail: isabelle.quidu@ensta-bretagne.fr

B. Zerr
e-mail: benoit.zerr@ensta-bretagne.fr

L. Jaulin et al. (eds.), *Marine Robotics and Applications*, Ocean Engineering & Oceanography 10, https://doi.org/10.1007/978-3-319-70724-2_1

[4–6] or also to carry out change detection [7–10]. Such an alignment can therefore be achieved through various methods. We usually split these methods into two groups, the symbolic ones (features-based) [1–5, 7, 11–14] and the iconic ones (intensity-based) [6, 8, 9, 15–20]. While the former try to extract and match features such as objects lying on the seabed, salient points or homogenous areas, the latter directly work with the pixels intensity through various similarity (resp. dissimilarity) measures to maximize (resp.minimize), to find the optimal transformation.

As we are looking for a method which does not rely on any features detection and description step, we focus on intensity-based methods. Indeed, we want such an algorithm to apply whatever the seabed type (presence of man-made objects or not, presence of sand ripples or not, textured seabed). A key point of a block matching algorithm is the choice of the similarity metric. The most famous one is undoubtedly the mutual information [21] which has massively been used by the medical image processing community [22] but also in the underwater field [17, 18]. However, according to Vandrish [3], such a metric can perform poorly compared to other similarity metrics such as the phase correlation. In addition, in [23], we demonstrated that a zero normalized cross-correlation (ZNCC) can be sufficient to perform registration. Moreover, the ZNCC, computed in the Fourier domain, retrieves the optimum through an unique iteration, while other metrics will have to look for the maximum in the entire search space, requiring much more iterations. As our algorithm should be designed to operate in real-time, such an advantage cannot be neglected. Although it could be conceivable to search for such a maximum through an optimization process, we can not insure to converge towards the global optimum. Moreover, the masked version of the Fourier NCC proposed by Padfield [24] is of interest in our case, allowing us to only take into account some parts of sonar tracks during registration.

In Sect. 1.2, we emphasize the key steps of the proposed block-matching algorithm which results on real high-resolution sonar tracks are provided in Sect. 1.3.

1.2 Non-rigid Registration Algorithm

1.2.1 Masked Normalized Cross-Correlation

As it has previously been shown in a rigid context [23], the zero normalized cross-correlation (ZNCC) is suitable to perform sonar tracks registration. The ZNCC between two images f_1 and f_2 can be expressed, in the spatial domain, as:

$$ZNCC(u,v) = \frac{\sum\limits_{(x,y)\in\cap_{u,v}} (f_1(x,y) - \overline{f_1})(f_2(x-u,y-v) - \overline{f_2^{(u,v)}})}{\sqrt{\sum\limits_{(x,y)\in\cap_{u,v}} (f_1(x,y) - \overline{f_1})^2}\sqrt{\sum\limits_{(x,y)\in\cap_{u,v}} (f_2(x-u,y-v) - \overline{f_2^{(u,v)}})^2}} \tag{1.1}$$

where $\cap_{u,v} = \{(x, y) \in f_1 \cap f_2^{(u,v)}\}$ where $f_2^{(u,v)}$ is a shifted version of f_2 with shifts u, v, and $\bar{f}_i$ corresponds to the average of f_i on this same interval.

In addition, it is possible to insert binary masks (m_i for image f_i) in the ZNCC Fourier formulation (1.2) in order to discard certain regions from both images [24] as they would affect the registration results adversely. This is known as the Masked Normalized Cross-Correlation (MNCC). Thus, in our underwater context, such an ability is welcome for various reasons. Firstly, as we deal with geo-referenced sonar tracks, it allows us to register blocks located near the sonar track borders while only taking into account pixels associated with sonar data. Secondly, as reference and repeated tracks can differ from each other due to differences in their point of view and/or grazing angle, it could sometimes be interesting to remove shadows from the mask to achieve registration. However, we do not need to retrieve a scale factor and a rotation factor. Indeed, we assume that these global parameters can be modeled through our block-matching formulation.

$$
\begin{aligned}
MNCC = & \frac{\mathcal{F}^{-1}(F_2 \odot F_1^*) - \frac{\mathcal{F}^{-1}(F_2 \odot M_1^*) \odot \mathcal{F}^{-1}(M_2 \odot F_1^*)}{\mathcal{F}^{-1}(M_2 \odot M_1^*)}}{\sqrt{\mathcal{F}^{-1}(\mathcal{F}(f_2 \odot f_2) \odot M_1^*) - \frac{(\mathcal{F}^{-1}(F_2 \odot M_1^*))^2}{\mathcal{F}^{-1}(M_2 \odot M_1^*)}} \sqrt{\mathcal{F}^{-1}(M_2 \odot \mathcal{F}(f_1' \odot f_1')) - \frac{(\mathcal{F}^{-1}(M_2 \odot F_1^*))^2}{\mathcal{F}^{-1}(M_2 \odot M_1^*)}}} \\
& m_1' = rot(m_1, 180°) \\
& f_1' = rot(f_1, 180°) \\
& M_2 = \mathcal{F}(m_2) \\
& M_1^* = \mathcal{F}(m_1') \\
& \odot \text{ is the element-wise product operator}
\end{aligned}
\tag{1.2}
$$

1.2.2 Block-Matching Algorithm

As demonstrated in [23], a rigid body transformation can not be chosen to precisely register two sonar tracks. Indeed, sonar data projected in the Earth frame are prone to error due to carrier uncertainty navigation. To overcome such an issue, as detailed in Fig. 1.1, our algorithm first consists in dividing the reference track into several blocks at the first available resolution level by means of a regular grid. The size of such blocks should be small enough to be able to model local deformations but large enough to contain sufficient information about the seabed. Then, a MNCC is computed between such a block from the reference image and another larger one from the repeated track. Figure 1.2 gives two examples of the MNCC between two blocks. Once all blocks have been processed, the yielded vector field must undergo a two-steps filtering as it will be explained in the next Sects. (1.2.3 and 1.2.4). Such a filtered vector field is then used to initialize the registration at the next finer level, allowing to decrease the search space and thus decreasing the computational cost. Finally, the vector field obtained at the finest resolution is used to perform the repeated track warping (Sect. 1.2.6).

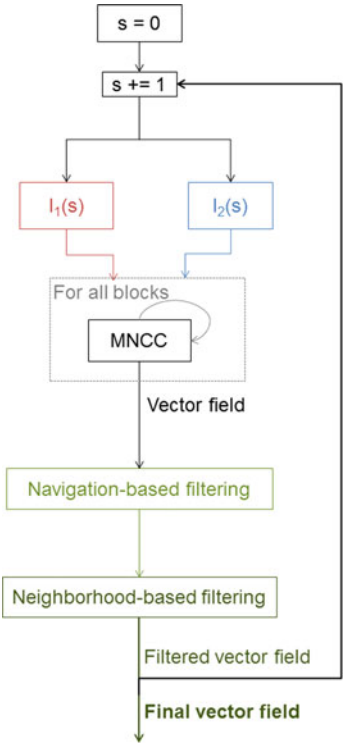

Fig. 1.1 Overview of our multi-resolution block-matching algorithm

1.2.3 Navigation-Based Filtering

As the peak location provided by the MNCC does not necessarily correspond to a physically reliable displacement, we rely on the latitude and longitude uncertainties provided by the Inertial Navigation System (INS), for each transmitted ping, to get rid of such outliers. Such outliers can for example be due to repetitive patterns such as linear trawling net marks or blocks located within homogeneous areas where the correlation peak can be sensitive due to the lack of distinctive features.

Thus, let p_i^{ref} (resp. p_i^{rep}) be the observation of the ith true contact p_i (unobserved) acquired during the reference (resp. repeated) pass:

$$\begin{aligned} p_i^{ref} &\sim \mathcal{N}(p_i, \Sigma_i^{ref}) \\ p_i^{rep} &\sim \mathcal{N}(p_i, \Sigma_i^{rep}) \end{aligned} \tag{1.3}$$

Σ_i^{ref} (resp. Σ_i^{rep}) is the reference (resp. repeated) diagonal covariance matrice.

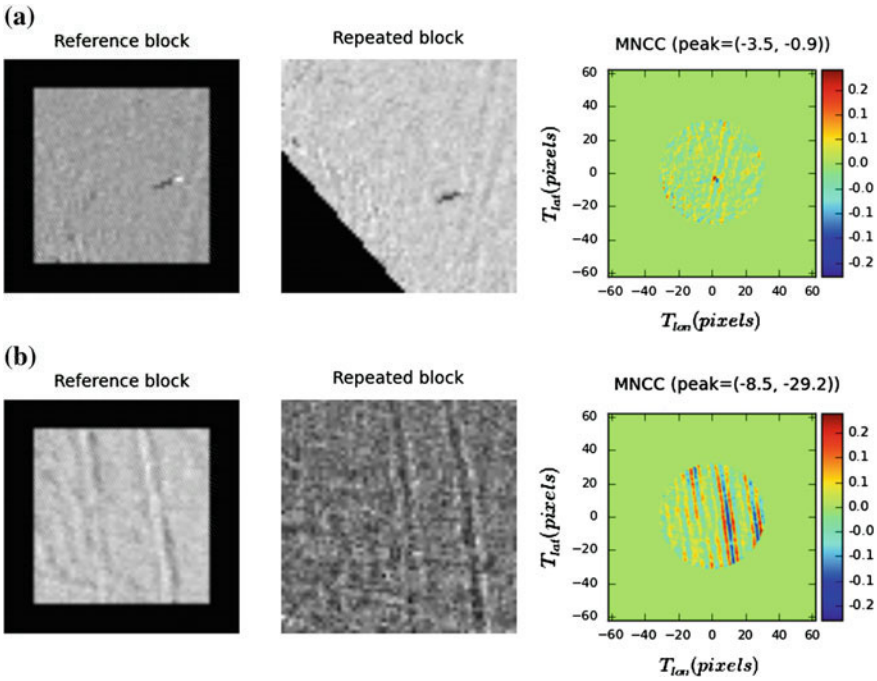

Fig. 1.2 Examples of masked normalized cross-correlation computed at the coarsest resolution level. **a** MNCC yielding the expected displacement between both blocks owing to the mine-like contact. **b** MNCC providing an erroneous displacement due to the linear trawling net mark raising an ambiguity

As p_i^{ref} and p_i^{rep} are considered independent, the distribution of $d_i = p_i^{ref} - p_i^{rep}$ can be written as:

$$\begin{aligned} d_i &\sim \mathcal{N}(0, \Sigma_i^{ref} + \Sigma_i^{rep}) \\ &\sim \mathcal{N}(0, \Sigma_i) \end{aligned} \tag{1.4}$$

with:

$$\begin{aligned} \Sigma_i &= \begin{pmatrix} \left(\sigma_{i,lat}^{ref}\right)^2 + \left(\sigma_{i,lat}^{rep}\right)^2 & 0 \\ 0 & \left(\sigma_{i,lon}^{ref}\right)^2 + \left(\sigma_{i,lon}^{rep}\right)^2 \end{pmatrix} \\ &= \begin{pmatrix} \sigma_{i,lat}^2 & 0 \\ 0 & \sigma_{i,lon}^2 \end{pmatrix} \end{aligned} \tag{1.5}$$

A registration vector $v_i \begin{pmatrix} \Delta_{i,lat} \\ \Delta_{i,lon} \end{pmatrix}$ will thus be kept if and only if, it is contained in the following uncertainty ellipsis:

$$\left(\frac{\Delta_{i,lat}}{\sigma_{i,lat}}\right)^2 + \left(\frac{\Delta_{i,lon}}{\sigma_{i,lon}}\right)^2 \leq K \tag{1.6}$$

K can be chosen according to the χ^2 distribution (with 2 degrees of freedom), to set the probability to keep such a vector.

1.2.4 Neighborhood-Based Filtering

In the previous subsection, vectors have separately been filtered in examining their latitude and longitude components. At the output of such a norm-based filtering, it can still remain outliers. Indeed, all vectors contained in the ellipse formed by axes $(\sigma_{lat}, \sigma_{lon})$ are remaining while they do not have a physical meaning. Therefore, it is now necessary to perform another filtering rather based on their neighborhood. To do so, our approach is based on the computation, for a given vector X (origin x, orientation $\theta(x)$), of its inverse distance weighting estimate $\widehat{\theta}(x)$ (1.7) thanks to its N neighboring vectors X_i (origin x_i, orientation $\theta(X_i)$) (1.7). In addition to the distance, the previously computed cross-correlation peak value $\gamma(X_i)$ is introduced in the weights computation. The comparison of $\theta(X)$ with $\widehat{\theta}(X)$ in one hand, and $\widehat{\|X\|}$ with $\|X\|$ in the other hand, thus allows to decide wether the current vector should be considered as an outlier (1.8). T_θ and $T_{\|.\|}$ are user-defined thresholds.

$$\omega(X, X_i) = \frac{\gamma(X_i)}{d(x, x_i)^p}$$

$$\widehat{\theta}(X) = \frac{\sum_{i=1}^{N} \theta(X_i)\omega(X, X_i)}{\sum_{i=1}^{N} \omega(X, X_i)} \tag{1.7}$$

$$\widehat{\|X\|} = \frac{\sum_{i=1}^{N} \|X_i\|\omega(X, X_i)}{\sum_{i=1}^{N} \omega(X, X_i)}$$

$$\|\widehat{\theta}(X) - \theta(X)\| \leq T_\theta$$

$$\frac{1}{T_{\|.\|}} \leq \frac{\widehat{\|X\|}}{\|X\|} \leq T_{\|.\|} \tag{1.8}$$

The vector field variation should obviously be considered as slow for our approach to be valid.

1.2.5 Multi-resolution Framework

The multi-resolution framework consists in initializing the next level thanks to the vector field computed at the previous one (Fig. 1.3). Thus, this allows to reduce the search space along the resolution levels (denoted by s) i.e. constraining the vector field while speeding up the algorithm.

As some vectors will be dismissed once filtering steps are done, it is possible to generate new vectors at the next scale by duplicating inlying vectors in their neighborhood in order not to run out of vectors.

Regarding the blocks size, a compromise has to be found. Indeed, let S_{ref} (resp. S_{rep}) be the size of a block from the reference track (resp. repeated track). S_{ref} should be large enough to contain enough information to locally describe the seabed but narrow enough to be able to model local deformations. However, S_{rep}, which is relatively large at the very first scale, decreases along the resolution levels as the correlation peak is more and more accurately localized.

1.2.6 Warping Functions

Once the final vector field has been obtained, the next step consists in computing a transformation for every pixel belonging to the repeated track. Several transformation functions have been considered such as the radial basis functions whose general form is given in (1.9). The radial basis function $R(.)$ only depends on the distance between the ith data point and the point x ($x \in \mathbb{R}^2$ in the case of 2D image registration) to interpolate.

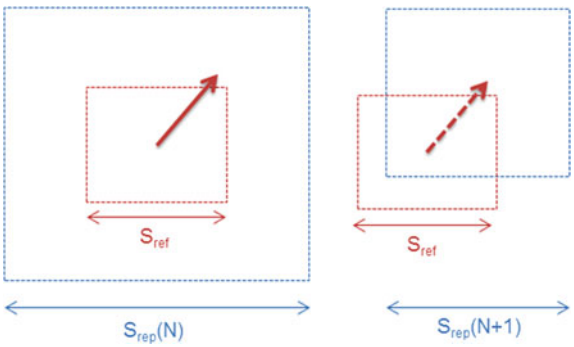

Fig. 1.3 (*from left to right*). At the coarsest level, for both blocks from the reference (red dotted square) and repeated (blue dotted square) tracks, the MNCC provides a displacement vector (red arrow). Thus, at the next level, the location of the block from the repeated track is initialized by means of the previously computed vector. This allows to decrease the search space along the resolution levels to speed up the registration process

$$f(x) = \sum_{i=1}^{n} \alpha_i \underbrace{R(d_i(x))}_{\text{Radial basis function}} + \underbrace{p_m(\boldsymbol{x})}_{\text{polynomial of degree m}} \tag{1.9}$$

The most famous RBF function is probably the Thin Plate Spline (TPS) (1.10) which has extensively been used in image registration.

$$R(d_i) = d_i^2 log(d_i) \tag{1.10}$$

However, in our case, although the center of blocks are arranged according to a regular grid, our algorithm may finally yield a sparse vector field whose density significantly varies in the space (vectors can thus all be located in a very small area of the sonar track). Thus, we often deal with extrapolation rather than interpolation, and, as the TPS have no local support property, it often fails to provide a reliable field.

Moreover, as we target real time processing, the interpolation complexity is a key factor. Indeed, while the TPS complexity is $O(nN)$ [25], n and N respectively being the number of data points and the number of points we want to interpolate, we prefer using faster methods such as triangulation-based ones or B-splines. As their name suggests, triangulation-based methods consists in triangulating the data points through Delaunay triangulation and then perform piecewise interpolation in each triangle by means of linear functions or cubic ones. A con of triangulation-based methods comes from the fact that they are only able to perform the interpolation inside the convex hull. To overcome this, we use the Shepard's method to estimate the vector field on sonar track borders and thus extend the convex hull.

Regarding B-splines methods, they have broadly been used in non-rigid and optimization-based medical image registration [26, 27], as they are locally supported (i.e. it only relies on some neighbors to perform the interpolation). Given a 2D control lattice ϕ, they can be expressed as

$$f(x, y) = \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(s) B_l(t) \phi_{i+k, j+l}, \tag{1.11}$$

where $i = \lfloor x \rfloor - 1$, $j = \lfloor y \rfloor - 1$, $s = \lfloor x \rfloor - x$, $t = \lfloor y \rfloor - y$. B_k and B_l are B-spline basis functions defined as

$$\begin{aligned} B_0(t) &= \frac{(1 - t^3)}{6}, \\ B_1(t) &= \frac{(3t^3 - 6t^2 + 4)}{6}, \\ B_2(t) &= \frac{(3t^3 - 6t^2 + 4)}{6}, \\ B_3(t) &= \frac{t^3}{6}. \end{aligned} \tag{1.12}$$

In Sect. 1.3.2, multilevel B-splines [28] and a triangulation-based approach [29] are compared.

1.3 Experimental Results

1.3.1 Dataset

To evaluate our registration algorithm, we have two high-resolution synthetic aperture sonar tracks acquired offshore Brittany coasts, near Brest, on September 2015. Such data have been acquired with the SAMDIS sonar sensor manufactured by Thales. While this sensor is able to perform multi-aspect acquisition, we only rely on the broadside aspect to perform our registration. The carrier is also equipped with an iXBlue PHINS inertial navigation system whose data are fused with a Teledyne RDI Workhorse 300 data velocity log (DVL). First and foremost, we project both sonar tracks in the Earth frame before storing them at different resolutions through a hierarchical data format.

We have chosen three different resolution levels, ranging from 0.5 m to 0.12 m, to benefit from the multi-resolution framework. A given resolution level of a track is obtained by successively locally averaging its previous scale. We get rid of coarsest levels and finest ones, as they respectively do not represent interesting features and are polluted by speckle noise (Fig. 1.4).

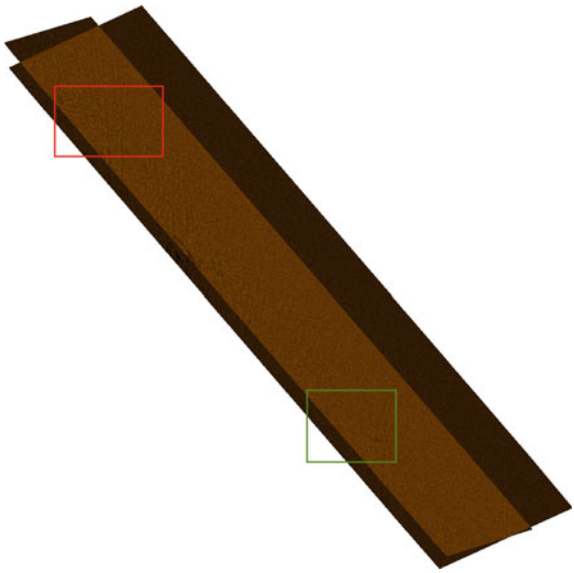

Fig. 1.4 Geo-projected reference and repeated sonar tracks before registration with patch A (red) and B (green)

1.3.2 Results

Intermediate results of the registration procedure, using the parameters detailed in Table 1.1, are shown in Fig. 1.5 to illustrate the role of each step. Indeed, while the very first vector field computed at the coarsest scale contains several outliers (a), the navigation-based filtering is able to get rid of them as they do not match the navigation uncertainties provided by the INS. The remaining outliers (the two pink

Table 1.1 Block-matching semi-operational parameters used in our experiment

Parameters used in our experiment							
Resolution level	S_{ref} [m]	S_{rep} [m]	d	N	K	T_θ	$T_{\|.\|}$
Coarsest	30	40	2	all vectors	5.991	45°	1.4
Intermediate	20	22					
Finest	15	16					

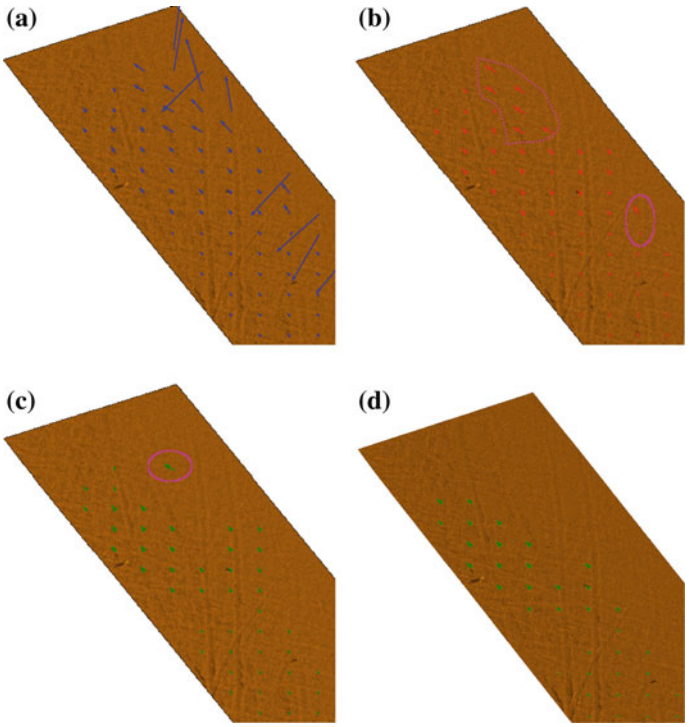

Fig. 1.5 Patch extracted from the repeated track. **a** Vector field after block-matching at the coarsest resolution level. **b** Vector field at the coarsest resolution after navigation-based filtering. **c** Vector field at the coarsest resolution after neighborhood-based filtering. **d** Filtered vector field at the finest resolution level

contours in (b)) are then filtered using the neighborhood-based approach described in Sect. 1.2.4. However, as it can be seen from (c), an outlier remains after all these filtering steps, but, as the correlation is probably not stable along the resolution levels, it disappears from the final vector field (d), and thus allows to obtain a more reliable warping of the repeated track.

We here provide both interpolated fields (X and Y axes) computed by means of Delaunay triangulation with cubic piecewise function and multilevel B-splines (Fig. 1.6) for the intermediary resolution level consisting in an image of 3072 $\times$ 3072 pixels. While the B-splines method provides a smoother field than the triangulation-based one (due to extrapolation outside the initial convex hull), our current implementation takes 28s to be computed whereas the triangulation-based one only needs 1.6s.

Nonetheless, as our sonar tracks are practically rendered using OpenGL, instead of computing the displacement for each pixel of the repeated track, we only compute such a displacement for the OpenGL vertices. Indeed, thus, when the vertices are shifted, the sonar track, which is stored as a texture in OpenGL, will be automatically warped through GPU interpolation, thus speeding-up the warping process.

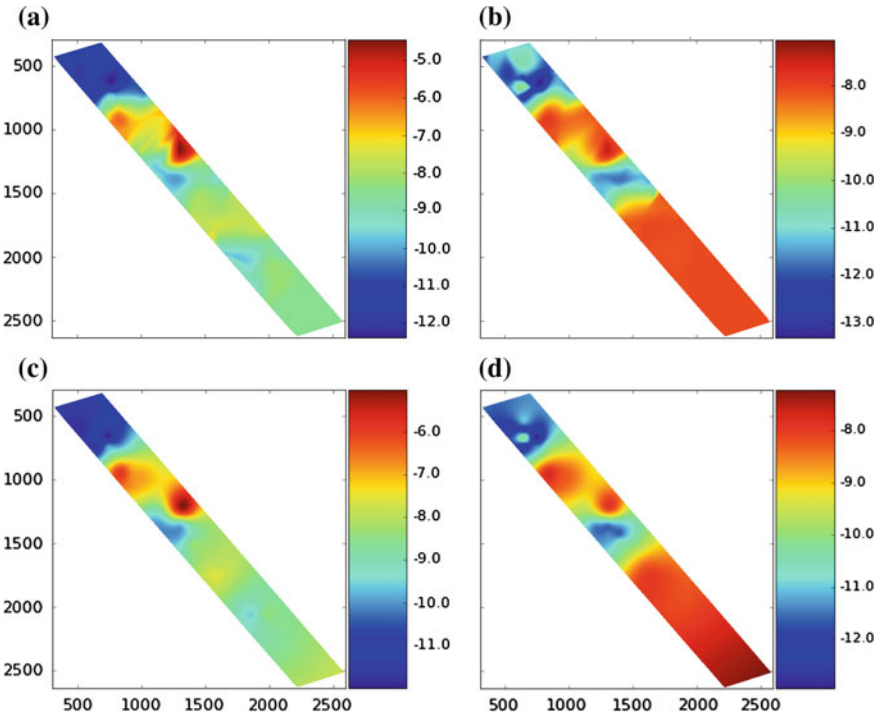

Fig. 1.6 Interpolated vector fields (in pixels). Interpolation through Delaunay triangulation with cubic piecewise function for X(**a**) and Y(**b**) axes. Interpolation by means of multilevel B-splines for X(**c**) and Y(**d**) axes