



Open-source feature-tracking algorithm for sea ice drift retrieval from Sentinel-1 SAR imagery

Stefan Muckenhuber, Anton Andreevich Korosov, and Stein Sandven

Nansen Environmental and Remote Sensing Center (NERSC), Thormøhlensgate 47, 5006 Bergen, Norway

Correspondence to: Stefan Muckenhuber (stefan.muckenhuber@nersc.no)

Received: 19 November 2015 – Published in The Cryosphere Discuss.: 21 December 2015

Revised: 15 April 2016 – Accepted: 19 April 2016 – Published: 26 April 2016

Abstract. A computationally efficient, open-source feature-tracking algorithm, called ORB, is adopted and tuned for sea ice drift retrieval from Sentinel-1 SAR (Synthetic Aperture Radar) images. The most suitable setting and parameter values have been found using four Sentinel-1 image pairs representative of sea ice conditions between Greenland and Severnaya Zemlya during winter and spring. The performance of the algorithm is compared to two other feature-tracking algorithms, namely SIFT (Scale-Invariant Feature Transform) and SURF (Speeded-Up Robust Features). Having been applied to 43 test image pairs acquired over Fram Strait and the north-east of Greenland, the tuned ORB (Oriented FAST and Rotated BRIEF) algorithm produces the highest number of vectors (177 513, SIFT: 43 260 and SURF: 25 113), while being computationally most efficient (66 s, SIFT: 182 s and SURF: 99 s per image pair using a 2.7 GHz processor with 8 GB memory). For validation purposes, 314 manually drawn vectors have been compared with the closest calculated vectors, and the resulting root mean square error of ice drift is 563 m. All test image pairs show a significantly better performance of the HV (horizontal transmit, vertical receive) channel due to higher informativeness. On average, around four times as many vectors have been found using HV polarization. All software requirements necessary for applying the presented feature-tracking algorithm are open source to ensure a free and easy implementation.

1 Introduction

Sea ice motion is an essential variable to observe from remote sensing data, because it strongly influences the distribution of sea ice on different spatial and temporal scales.

Ice drift causes advection of ice from one region to another and export of ice from the Arctic Ocean to the sub-Arctic seas. Antarctic sea ice is even more mobile and its strong seasonality is linked to the ice transport from high to low latitudes (IPCC, 2013). Furthermore, ice drift generates convergence and divergence zones that cause formation of ridges and leads. However, there is still a lack of extensive sea ice drift data sets with sufficient resolution to estimate convergence and divergence on a spatial scaling of less than 5 km.

The regions of interest are the ice-covered seas between Greenland and Severnaya Zemlya, i.e. the Greenland Sea, Barents Sea, Kara Sea and the adjacent part of the Arctic Ocean. This area is characterized by a strong seasonal cycle of sea ice cover, a large variation of different ice classes (multiyear ice, first-year ice, marginal ice zone etc.) and a wide range of drift speeds (e.g. strong ice drift in Fram Strait).

With systematic acquisition of space-borne Synthetic Aperture Radar (SAR) data over sea ice areas, Kwok et al. (1990) have demonstrated that high-resolution ice drift fields can be derived from SAR data. SAR is an active microwave radar which acquires data independently of solar illumination and weather conditions. Sea ice motion fields of the Arctic Ocean with a grid spacing of 5 km have been produced on a weekly basis between 1997–2012 using Radarsat and ENVISAT (Environmental Satellite) SAR data and the geophysical processor system introduced by Kwok et al. (1990). Thomas et al. (2008a) have used pattern recognition to calculate sea ice drift between successive ERS-1 (European remote-sensing satellite) SAR images with a resolution of 400 m. This work has been continued by Hollands and Dierking (2011) using Advanced SAR (ASAR) data from ENVISAT. Komarov and Barber (2014) used a similar pattern-

matching technique to evaluate ice motion results from dual-polarization Radarsat-2 images.

With the successful launch of Sentinel-1A in April 2014 and the planned launch of Sentinel-1B in early 2016, high-resolution SAR data will be delivered for the first time with open and free access for all users and unprecedented revisit time of less than one day in the Arctic (ESA, 2012). This introduces a new era in SAR Earth observation. Sea ice drift data with medium resolution (10 km) are provided operationally via the Copernicus Marine Environment Monitoring Service (CMEMS, <http://marine.copernicus.eu>), but no sea ice drift algorithm using Sentinel-1 data has been published so far. The objective of this paper is to identify and develop the most efficient open-source algorithm for high-resolution sea ice drift retrieval from Sentinel-1 data.

Our goal is to exploit recent improvements and developments in computer vision by adopting a state-of-the-art feature-tracking algorithm to derive sea ice drift (i.e. vectors of sea ice displacement). Current pattern-matching algorithms constrain the high-resolution vectors with low-resolution estimates for practical reasons. Using feature tracking, drift vectors can be derived independently from the surrounding motion, which leads to better performance e.g. along shear zones. For application on large data sets and for operational use, we considered a computationally efficient algorithm, called ORB (Oriented FAST and Rotated BRIEF) (Rublee et al., 2011), tuned it for sea ice drift retrieval from Sentinel-1 imagery and compared the results with other available feature-tracking algorithms and existing sea ice drift products.

The software requirements necessary for deriving ice drift fields from Sentinel-1 data (Python with OpenCV and the Python toolbox Nansat) are all open source to ensure a free, user friendly and easy implementation.

The paper is organized as follows: Sect. 2 introduces the used Sentinel-1A data product. The ORB algorithm description and the used methods for tuning, comparison and validation are presented in Sect. 3. The recommended parameter set including the tuning, comparison and validation results are provided in Sect. 4. The discussion can be found in Sect. 5.

2 Data

The Sentinel-1 mission, an initiative of the European Union and operated by the European Space Agency (ESA), is composed of a constellation of two identical satellites sharing the same near-polar, sun-synchronous orbit: Sentinel-1A, launched in April 2014, and Sentinel-1B, planned to launch in early 2016. Sentinel-1 carries a single C-band Synthetic Aperture Radar (SAR) instrument measuring radar backscatter at a centre frequency of 5.405 GHz and supporting dual polarization (HH + HV, VV + VH). With both satellites operating, the constellation will have a revisit time of less than one day in the Arctic. Radar data are delivered to Coperni-

cus services within an hour of acquisition with open and free access for all users (ESA, 2012).

The Sentinel-1 product used in this paper is called Extra-wide Swath Mode Ground Range Detected with Medium Resolution. These images cover an area of 400 km × 400 km with a pixel spacing of 40 m × 40 m (resolution: 93 m range × 87 m azimuth; residual planimetric distortions: within 10 m; Schubert et al., 2014) and provide both HH (horizontal transmit, horizontal receive) and HV (horizontal transmit, vertical receive) polarization.

Four image pairs (Table 1) representative of our region of interest have been chosen for parameter tuning. Furthermore, 43 image pairs acquired over Fram Strait and north-east of Greenland (Fig. 8) have been used to test the performance of different feature-tracking algorithms. To ensure an independent evaluation, the 43 test image pairs have not been used for parameter tuning. The two considered sets of image pairs cover both a range of different sea ice conditions (pack ice, fast ice, leads, ridges, marginal ice zone, ice edge etc.) and intervals between the acquisitions. We focused on winter and spring data, since our area of interest experiences the highest sea ice cover during this period.

3 Method

Sentinel-1 data sets were opened and processed with the open-source software Nansat (see Appendix A; Korosov et al., 2015, 2016). Nansat is a scientist-friendly Python toolbox for processing 2-D satellite Earth observation data. It is based on the Geospatial Data Abstraction Library (GDAL) and provides easy access to geospatial data, a simple and generic interface to common operations including reading, geographic transformation and export. Nansat proves to be efficient both for development and testing of scientific algorithms and for fast operational processing. To extend the functionality of GDAL, Nansat reads metadata from XML files accompanying Sentinel-1 data and supplements the GDAL data model with georeference information stored as ground control points (GCPs). Originally GCPs are pairs of latitude/longitude and corresponding pixel/line coordinates. In order to increase the accuracy of the geographic transformation, the projection of GCPs is changed from cylindrical to stereographic, placed at the centre of the scene. The re-projected GCPs are then used by GDAL to calculate geographic coordinates of any pixel in the raster using spline interpolation. Reprojection of GCPs does not require much additional computational effort, but improves the result significantly, particularly at high latitudes.

The normalized radar cross section (σ^0) is calculated from raw Sentinel 1A data using the following equation:

$$\sigma^0 = DN_i^2 / A_i^2, \quad (1)$$

where DN_i is the digital number provided in the source TIFF file, A_i is the value of normalization coefficient from the ac-

Table 1. Sentinel-1 image pairs used for parameter tuning.

Region	First image acquisition time, UTC	Second image acquisition time, UTC	Time gap
Fram Strait	28 Mar 2015 07:44:33	29 Mar 2015 16:34:52	33 h
Svalbard North	22 Apr 2015 06:46:23	23 Apr 2015 13:59:03	31 h
Franz Josef Land	24 Mar 2015 03:21:13	24 Mar 2015 11:30:06	8 h
Kara Sea	22 Apr 2015 11:37:16	24 Apr 2015 11:20:59	48 h

comparing calibration metadata and i is an index of a pixel (Anonymous, 2014). No additional preprocessing of SAR data was performed.

Our algorithm for sea ice drift detection includes three main steps: (a) resampling of raw data to lower resolution, (b) detection and matching of features and (c) comparison/validation.

- To decrease the influence of speckle noise and increase the computational efficiency, the resolution is reduced before applying the ice drift algorithm from 40 to 80 m pixel spacing using simple averaging.
- For detection and tracking of features on large data sets and for operational use, a computationally efficient algorithm, called ORB (Rublee et al., 2011), has been used. In our numerical experiments we tuned the parameters of ORB for optimal SAR sea ice drift application. The most suitable parameter set (including spatial resolution of SAR image, patch size of FAST descriptor, number of pyramid levels, scale factor, etc.) has been evaluated for our area and season of interest.
- The introduced ORB set-up is compared to other available OpenCV feature-tracking algorithms, CMEMS data and manually drawn vectors for performance appraisal and validation.

3.1 ORB algorithm

ORB (Oriented FAST and Rotated BRIEF) is a feature-tracking algorithm introduced by Rublee et al. (2011) as “a computationally efficient replacement to Scale-Invariant Feature Transform (SIFT) that has similar matching performance, is less affected by image noise, and is capable of being used for real-time performance”. ORB builds on the FAST keypoint detector (Rosten and Drummond, 2006) and the binary BRIEF descriptor (Calonder et al., 2010) with many modifications to enhance the performance. It uses FAST to find multiscale keypoints on several pyramid levels and applies a Harris corner measure (Harris and Stephens, 1988) to pick the best keypoints. To achieve rotation invariance, the orientation of the keypoint is calculated by using the intensity-weighted centroid of a circular patch with the located keypoint at the centre. Rublee et al. (2011) states that the ORB descriptor performance is equal to SIFT (Lowe,

2004) and higher than Speeded-Up Robust Features (SURF) (Bay et al., 2006). Like Rublee et al. (2011), we use a brute-force matcher and Hamming distance for feature matching. Unlike SIFT and SURF, ORB is an open-source software and use and distribution are not limited by any licenses.

Before the feature-tracking algorithm can be applied to a satellite image, the SAR backscatter values σ^0 have to be transformed into the intensity i range ($0 \leq i \leq 255$ for $i \in \mathbb{R}$) used in OpenCV. This transformation is done by using Eq. (2) and setting all intensity values below and above the range to 0 and 255.

$$i = 255 \cdot \frac{\sigma^0 - \sigma_{\min}^0}{\sigma_{\max}^0 - \sigma_{\min}^0}, \quad (2)$$

Lower and upper brightness boundaries σ_{\min}^0 and σ_{\max}^0 are user defined and chosen to be constant in order to limit the influence of speckle noise and be independent e.g. of high backscatter values σ^0 over land. Converting the linear backscatter values before the transformation into decibel units has been tested, but decreased the algorithm performance for both channels.

After the transformation into intensity values, keypoints are detected on both SAR scenes using the FAST-9 keypoint detector (Rosten and Drummond, 2006). FAST-9 compares the intensity I_p of a centre pixel to the intensities of pixels on the surrounding circle with a perimeter of 16 pixels (Fig. 1). If there exists a set of nine contiguous pixels in the circle which are all brighter than $I_p + t$, or all darker than $I_p - t$, the centre pixel is recognized as a keypoint. The threshold t is set low enough to get more than the predefined amount N of keypoints.

To detect features of different scales, the keypoint search is performed on several pyramid levels. The number of pyramid levels in combination with the scale factor defines the range and increment of the keypoint detection scaling. A scale factor of 2 means that each next pyramid level has four times fewer pixels, but such a large-scale factor degrades the feature matching score. On the other hand, a small-scale factor close to 1 means to cover a certain scale range needs more pyramid levels and hence, the computational cost increases.

FAST does not produce a measure of cornerness and Rublee et al. (2011) have found that it has large responses along edges. Harris corner measure (Harris and Stephens, 1988) is used to order the FAST keypoints according to their

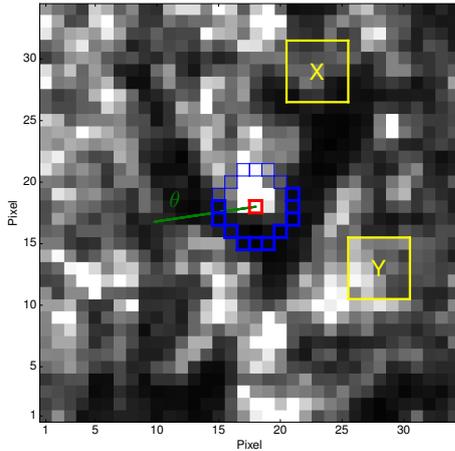


Figure 1. Subset of the first image from Fram Strait pair (Table 1) with centre at 2.31° W, 81.70° N and pixel spacing of 80 m. The centre pixel (red) is recognized as keypoint since ≥ 9 contiguous pixels (bold blue) of the surrounding blue circle have intensity values smaller than the centre minus threshold t . The orientation θ of the keypoint is shown with a green arrow. The displayed area (34×34 pixels) around the keypoint represents the considered patch p used for feature description. The yellow 5×5 pixels sub-windows X and Y are an example for a possible binary test sampling pair with $p(X) < p(Y)$ and hence, $\tau(p; X, Y) = 1$ (Eq. 7).

cornerness and reject less reliable keypoints. Considering a window $w(x, y)$ around the keypoint, the intensity derivatives I_x, I_y in x and y direction can be written in a matrix \mathbf{M} :

$$\mathbf{M} = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}. \quad (3)$$

The eigenvalues λ_1 and λ_2 of \mathbf{M} contain the intensity derivative in the direction of the fastest and slowest change respectively. Based on λ_1 and λ_2 , a score R can be calculated for each keypoint:

$$R = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2, \quad (4)$$

with k being an empirical constant. A high intensity variation in both dimensions returns a high R value. The top N keypoints with the highest R values are used and the rest is rejected.

FAST does not include orientation, but ORB adds a direction to each keypoint using the intensity-weighted centroid from Rosin (1999). The moments m_{pq} of a circular area around the keypoint are used:

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y). \quad (5)$$

The intensity-weighted centroid has its location at the following:

$$C = \left(\frac{m_{01}}{m_{00}}, \frac{m_{10}}{m_{00}} \right). \quad (6)$$

The orientation θ (e.g. green arrow in Fig. 1) represents the direction of the vector connecting the keypoint with the intensity-weighted centroid. The moments m_{pq} are computed with x and y remaining within a circular region of radius r , where r is chosen to be the size of the patch p used for the following feature description Rublee et al. (2011).

After locating and adding orientation to the best N keypoints, a patch p around each keypoint is used for feature description (NB: keypoint refers to 1 pixel, feature refers to description of p). ORB applies a modified version of the binary descriptor BRIEF (Calonder et al., 2010). Rublee et al. (2011) defines a binary test τ for a patch p as follows:

$$\tau(p; X, Y) := \begin{cases} 1 & \text{if } p(X) < p(Y) \\ 0 & \text{if } p(X) \geq p(Y), \end{cases} \quad (7)$$

with $p(X)$ and $p(Y)$ being the intensities at test points X and Y . ORB uses 5×5 sub-windows as test points (e.g. in Fig. 1). Applying n binary tests on a single patch, Rublee et al. (2011) derive a binary feature vector f :

$$f_n(p) := \sum_{1 \leq i \leq n} 2^{i-1} \tau(p; X_i, Y_i). \quad (8)$$

The considered set of n binary tests with test points (X_i, Y_i) can be written in a $2 \times n$ matrix (Rublee et al., 2011):

$$\mathbf{S} = \begin{pmatrix} X_1 & \dots & X_n \\ Y_1 & \dots & Y_n \end{pmatrix}. \quad (9)$$

To be invariant to in-plane rotation, Rublee et al. (2011) steers \mathbf{S} according to the orientation θ using the corresponding rotation matrix \mathbf{R}_θ :

$$\mathbf{S}_\theta = \mathbf{R}_\theta \mathbf{S}. \quad (10)$$

A good set \mathbf{S} of sampling pairs needs to be uncorrelated, so that each pair adds new information to the descriptor and they must have high variance to make features more discriminative. Rublee et al. (2011) applied a greedy search to a large training data set to obtain a set for ORB with $n = 256$ relatively uncorrelated tests with high variance.

After the feature description, OpenCV allows different matching procedures for ORB. Like Rublee et al. (2011), we use brute-force matching and compare each feature of the first image to all features in the second image.

As a comparison measure, we use the Hamming distance, which is equal to the number of positions in which the two considered feature vectors have a different value.

$$\begin{aligned} \mathbf{b}_1 &= 1011101 \\ \mathbf{b}_2 &= 1001001, \end{aligned} \quad (11)$$

For example, comparing the two binary vectors \mathbf{b}_1 and \mathbf{b}_2 returns the Hamming distance $d = 2$, since the third and fifth position have a different value.

Our setting returns the best two matches and applies the ratio test from Lowe (2004) to decide whether the best match

is accepted or rejected. The match is accepted if ratio of the distances $\frac{d_1}{d_2} < \tau$ is below a given threshold. The ratio test eliminates a high number of false matches, while discarding only few correct matches.

3.2 ORB setting and parameter tuning

Achieving the best possible performance of ORB for sea ice drift from Sentinel-1 images requires a good setting and tuning of the parameters shown in Table 2.

It is not recommended to reproject one image onto the projection of the second image before applying the ORB algorithm, since this is computationally very expensive. Instead, geographic coordinates of the matched start and end point shall be calculated independently using the georeference information from GCPs of the first and second image.

Manual interpretation of ice drift results (using the training data from Table 1) reveals that a good compromise between amount of vectors and correct results can be achieved with a Lowe ratio test threshold equal to 0.75. That means that the Hamming distance of the best match has to be less than $0.75 \times$ Hamming distance of the second best match. Tested on the image pairs from Table 1, the ratio test showed a clearly better performance and is computationally less expensive than the alternative cross-check, where features are matched in both directions (first image to second image and vice versa) and rejected if the drift vectors are too different.

Unreasonably high sea ice displacements (e.g. above 40 km for a time difference between two scenes of ~ 30 h) are removed in a post-processing step from the drift field. In addition, displacements below 2.5 km are rejected during the testing to disregard matches over land. This does not influence the number of correct matches, since the sea ice displacement in all considered test images is above 2.5 km.

Based on our observations we assume that the proportion of wrong matches does not increase with increasing total number of matches. Under this assumption the algorithm performance refers to the total number of matches and is used to tune the algorithm parameters in Table 2. ORB is computationally more efficient, enabling testing the parameters over a wide range with high-resolution using both HH and HV polarization.

As a starting point, the tested parameters were set as follows: resize factor = 0.5, patch size = 31, pyramid levels = 8, scale factor = 1.2, HH limits = [0,0.12], HV limits = [0, 0.012] and ratio test = 0.8. As a compromise between performance and computational efficiency, the maximum amount of retained keypoints is set to 100 000. Tested range and parameter meaning are shown in Table 2.

In order to find an optimal value for the tested parameter, it is varied in a reasonable range, the feature-tracking algorithm is applied and the total number of matched vectors is found. Once the most suitable value for a tested parameter is found, it is applied for further testing.

3.3 Comparison of ORB to SIFT and SURF

The presented ORB algorithm has been compared to other OpenCV feature-tracking algorithms, namely SIFT (Lowe, 2004) and SURF (Bay et al., 2006), using 43 image pairs acquired over Fram Strait and north-east of Greenland (Fig. 8). SIFT and SURF were used in standard mode and the framework conditions were set to equal for the comparison. Image preprocessing has been carried out as described above, brute-force matching including the Lowe ratio test with threshold 0.75 has been applied for all three algorithms as well as the removal of unreasonably high sea ice displacements in a post-processing step. Since SIFT allows for defining the number of retained keypoints, this parameter has been set to 100 000 as done for ORB. The further tuning of SIFT and SURF is not the aim of this paper, since these two algorithms are not open source and computationally less efficient.

The distribution and reliability of the calculated vector fields have been assessed for each image pair using two parameters on a grid with cell size 1° longitude \times 0.2° latitude: number of derived vectors per grid cell (N) and root mean square distance (D) of all vectors in a grid cell computed as follows:

$$D = \frac{\sqrt{\sum_i (u_i - \tilde{u})^2 + (v_i - \tilde{v})^2}}{N}, \quad (12)$$

where i is the index of a vector inside the grid cell, u_i and v_i are the eastward and northward drift components and \tilde{u} , \tilde{v} the corresponding mean values. To combine the results of several image pairs, the sum of N and the mean of D is considered.

3.4 Validation

The ORB algorithm has been validated against drift data from two independent sources using the image pair Fram Strait (Table 1). First, 350 features were identified by a sea ice expert in both images and manually connected using ArcGIS. Second, sea ice drift vectors were taken from the Copernicus Marine Environment Monitoring Service (CMEMS, <http://marine.copernicus.eu>). The SAR ice drift product of CMEMS is operated by the Technical University of Denmark (DTU) and drift data are provided with a resolution of 10 km using pattern-matching techniques (Pederson et al., 2015, <http://www.seaice.dk/>).

Since the starting locations of ORB, manual and CMEMS vectors do not coincide, the corresponding (ORB) reference vectors were found as nearest neighbours within 5 km radius from the (CMEMS or manual) validation vectors.

Three parameters were considered for the comparison: root mean square error (E), slope (S) and offset (O) of the linear fit between the reference and validation vectors. E was calculated as follows:

Table 2. Recommended set of parameters for retrieval of sea ice drift from Sentinel-1 data using ORB.

Parameter	Meaning	Tested range (increment)	Recommended setting
Amount keypoints	Maximum number of keypoints to retain	–	100 000
Resize factor	Resolution reduction during pre-processing	0.5–1 (0.5)	0.5
Patch size	Size of descriptor patch in pixels	10–60 (1)	34
Pyramid levels	Number of pyramid levels	1–15 (1)	7
Scale factor	Pyramid decimation ratio	1.1–1.4 (0.1)	1.2
$[\sigma_{\min}^0, \sigma_{\max}^0]$ (HH)	Brightness boundaries for HH channel	[0–0.04, 0.01–0.2] (0.01)	[0, 0.08]
$[\sigma_{\min}^0, \sigma_{\max}^0]$ (HV)	Brightness boundaries for HV channel	[0–0.007, 0.001–0.02] (0.001)	[0, 0.013]
Ratio test	Threshold for ratio test	0.5–1 (0.1), 0.7–0.8 (0.01)	0.75

$$E = \sqrt{\frac{\sum_i (u_i - U_i)^2 + (v_i - V_i)^2}{n}}, \quad (13)$$

where i is the index of a vector pair (reference and validation vector) inside the entire sample, u_i and v_i are eastward and northward drift components of the validation vector, U_i and V_i are eastward and northward components of the reference vector and n is the number of vector pairs.

In addition, the CMEMS data have been validated against manual vectors in order to understand the credibility of the reference data.

4 Results

4.1 ORB parameter tuning

Table 2 shows the recommended parameter set for ORB Sentinel-1 sea ice drift application for our region and period of interest. Using these parameters yielded the best compromise between performance and computational efficiency for the four representative image pairs from Table 1.

4.1.1 Patch size

Figure 2 shows that changing the size (length and width) of the considered patch p between 10 and 60 pixels can modify the resulting amount of vectors by an order of magnitude. To resolve drift gradients with high resolution, the patch size shall be as small as possible. Taking this into account and the performance represented by the amount of matches, the most suitable patch size was chosen to be 34 pixels. For our training data set (Table 1), this yields on average around 1 and 4 vectors per 10 km^2 for HH and HV respectively. The four image pairs respond similar to a patch size variation. Franz Josef Land has the highest number of HH matches and the lowest for HV.

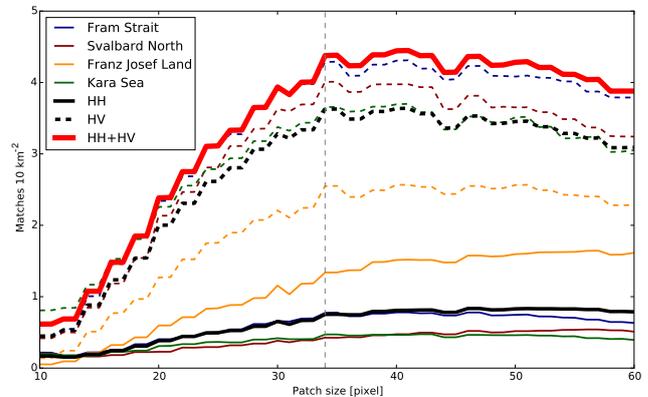


Figure 2. Patch size of descriptor vs. number of matches of the four test image pairs from Table 1. Solid and dashed lines represent results for HH and HV polarization respectively. Mean values of the four image pairs are shown in black and the sum of the mean values in red. Vertical grey line at 34 pixels represents chosen parameter.

4.1.2 Brightness boundaries

The performance of the algorithm (represented by the amount of matches) for different backscatter limits σ_{\max}^0 (Eq. 2) for HH and HV polarization is shown in Fig. 3. Within the chosen backscatter range, the amount of vectors can vary by an order of magnitude. As a compromise between the different results of the four image pairs, we suggest setting the upper brightness boundary σ_{\max}^0 to 0.08 and 0.013 for HH and HV. The chosen lower boundary σ_{\min}^0 is 0 for both HH and HV, because the number of matches decreases for increasing values of σ_{\min}^0 (not shown). Applying this setting on the training data set yields on average around 1 and 4 vectors per 10 km^2 for HH and HV.

4.1.3 Pyramid levels and scale factor

We calculated the number of matches using 1 to 14 pyramid levels and the scale factors 1.1, 1.2, 1.3 and 1.4. As a compromise between performance, i.e. number of matches, and computational efficiency (linked to the number of pyramid

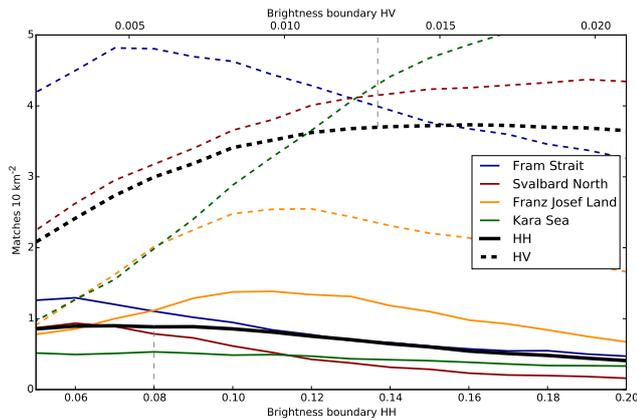


Figure 3. Upper brightness boundary σ_{\max}^0 (Eq. 2) vs. number of matches of the four test image pairs from Table 1. Solid and dashed lines represent results for HH and HV respectively. Black lines are the mean values of the four image pairs. Vertical grey lines at 0.08 (HH) and 0.013 (HV) represent chosen parameters.

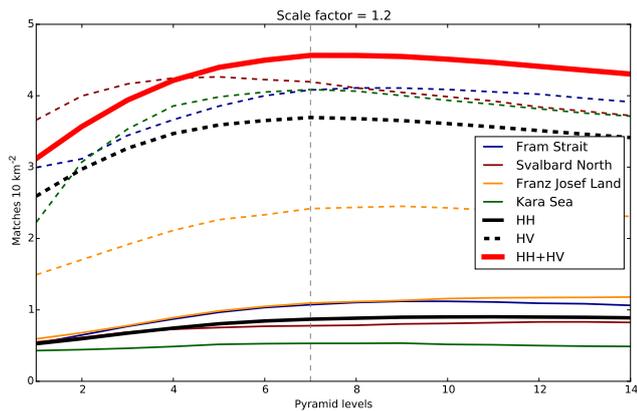


Figure 4. Number of pyramid levels vs. number of matches of the four test image pairs from Table 1 for a scale factor of 1.2. Solid and dashed lines represent results for HH and HV polarization. Mean values are shown in black and the sum of the mean values in red. Vertical grey line at 7 represents chosen number of pyramid levels.

levels), a scale factor of 1.2 with seven pyramid levels was chosen. As shown in Fig. 4, the number of matches does not increase significantly when using more than seven pyramid levels and even decreases towards 14 pyramid levels.

4.2 HH and HV comparison

Figures 2, 3 and 4 display the HH and HV results with solid and dashed lines. All image pairs show significantly better performance of the HV channel. On average, around four times as many vectors have been found using HV. Even the image pair Franz Josef Land (Table 1), which has the best HH and the worst HV performance, shows more than two times as many vectors using HV channel. However, due to the different appearance of sea ice in the HH and HV im-

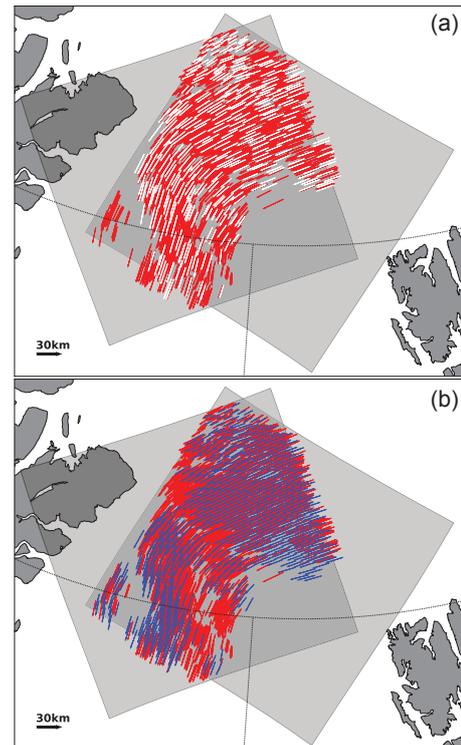


Figure 5. Sea ice drift of the Sentinel-1 image pair Fram Strait (Table 1). (a) Manually drawn vectors are shown in white and the computed ORB vectors in red. (b) shows ORB vectors in comparison to the drift vectors from the CMEMS/DTU data (blue).

age, the spatial distribution of the resulting drift vectors is also slightly different.

Figure 6 shows the spatial distribution of identified keypoints and matched features in a 200×200 pixels sub-image from image pair Fram Strait (Table 1). The results for HH and HV are displayed in two separate panels. The density of identified keypoints in HH (11 keypoints per 10×10 pixels window) is in the same order of magnitude as in HV (15 keypoints per 10×10 pixels window). This is expected, since the number of retained keypoints for both channels is set to 100 000 for the entire scene. However, the number of matched features in HH is significantly lower (0.15 features per 10×10 pixels window) than in HV (1.6 features per 10×10 pixels window). The observed difference in matching success can be explained by looking at the frequency distribution of the radar backscatter standard deviation in a sliding window with same size as used for feature description (34×34 pixels). The comparison in Fig. 7 shows that HH provides a few windows with very high variability, i.e. high standard deviation, but the majority have very low backscatter variability (sharp peak with mode 20). On the HV image, however, most of the windows have a medium to high backscatter variability (wide peak with mode 25) which is more favourable for keypoint detection.

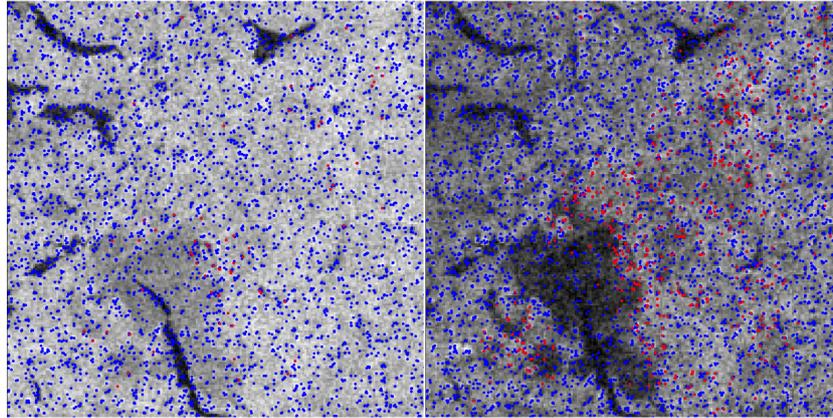


Figure 6. Identified keypoints (blue) and matched features (red) on a 200×200 pixels sub-image from the pack ice area in image pair Fram Strait (Table 1). Results of HH are shown in the left panel and HV in the right panel.

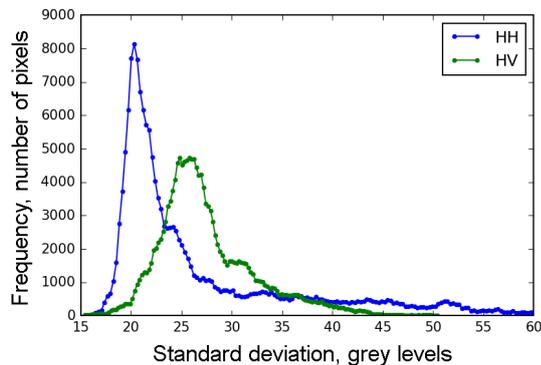


Figure 7. Frequency distribution of radar backscatter standard deviation using a 34×34 pixels sliding window (step = 1 pixel) on a 1000×1000 pixels sub-image from image pair Fram Strait (Table 1). The radar backscatter is scaled to range 0–255 using Eq. (2). The considered sub-image covers pack ice, marginal ice zone and small parts of open water. Results for HH are shown in blue and HV in green.

4.3 Comparison with SIFT and SURF

A total of 177 513, 43 260 and 25 113 vectors are found for the 43 test image pairs (Fig. 8) using ORB, SIFT and SURF respectively (Fig. 9a). Comparing the vector fields using the sum of N and the mean of D , as described in Sect. 3, shows that ORB covers the largest area with close to 1000 vectors per grid cell and lower root mean square distance values.

Comparing the distributions of N (Q-Q plot in Fig. 10, left panel), shows that ORB derives in all cases around five times as many vectors than SIFT and SURF. The Q-Q plot in the right panel of Fig. 10 considers the distributions of D . For $D < 500$ m, the vectors derived by ORB exhibit a higher variability within one grid cell (slightly higher D), probably due to a larger number of vectors N . For the higher root



Figure 8. Overlapping area of 43 Sentinel-1 image pairs used to compare ORB, SIFT and SURF. The image pairs have been acquired between 2 January and 21 March 2015 with time gaps varying between 7 and 48 h.

mean square values ($D > 500$ m), SIFT and SURF vectors are much less consistent than ORB vectors (higher D).

4.4 Computational efficiency

The OpenCV feature-tracking algorithms ORB, SIFT and SURF in combination with the Python toolbox, Nansat, are computationally efficient (total processing time on regular MacBook Pro: 2–4 min) and allow high-resolution sea ice drift retrieval from data sets with large temporal and spatial extent. The processing times shown in Table 3 are based on testing the algorithms on a MacBook Pro from early 2013 with a 2.7 GHz Intel Core i7 processor and 8 GB 1600 MHz DDR3 memory. Applying the introduced ORB algorithm

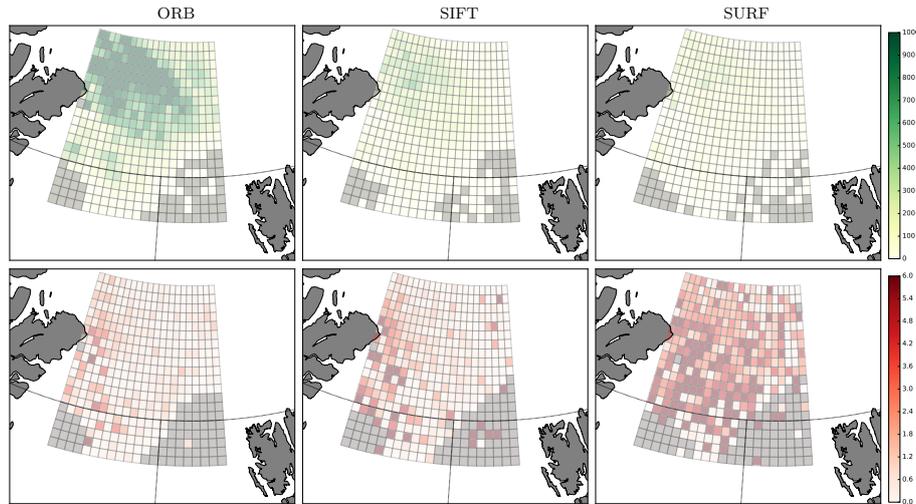


Figure 9. Sea ice drift derived from 43 Sentinel-1 image pairs (Fig. 8) using both HH and HV channel and ORB (first column, 177 513 vectors), SIFT (second column, 43 260 vectors) and SURF (third column, 25 113 vectors) algorithm. The panels show the sum of the number of vectors per grid cell N (green, first row) and mean root mean square distance D in km (red, second row).

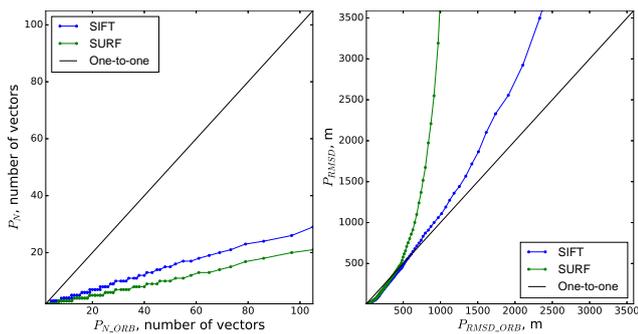


Figure 10. Q-Q plot of number of vectors N (left panel) and root mean square distance D (right panel) from results shown in Fig. 9. Tuned ORB algorithm (x axis) compared to SIFT (y axis, blue dots) and SURF (y axis, green dots).

needs 36 and 67 % of the processing time to compute drift fields with SIFT and SURF.

4.5 Validation

Since reference vectors were searched only within a given radius of the validation vectors, the number of matches decreased for the ORB vs. manual comparison from 350 possible matches to 314, for ORB vs. CMEMS from 560 to 436 and for CMEMS vs. manual from 350 to 201 (Table 4). The average distances between compared vectors were 1702, 2261 and 3440 m for ORB vs. manual, ORB vs. CMEMS and CMEMS vs. manual respectively.

The validation of ORB vectors with manually derived vectors (Fig. 5a, Table 4) reveals a high accuracy of our tuned ORB algorithm with root mean square error $E = 563$ m, slope $S = 1.02$ and offset $O = -372$ m. Given the displace-

Table 3. Processing times for sea ice drift computation from one channel.

Process	Time [s]
Create two Nansat objects from Sentinel-1 image pair	21.1
Read matrixes from two Nansat objects	48.8
Apply feature-tracking algorithm – ORB	65.8
Apply feature-tracking algorithm – SIFT	181.8
Apply feature-tracking algorithm – SURF	98.5

ment range for the used image pair of 10–35 km, the relative error of the algorithm (ratio of E to mean drift) is 2.5 %.

The vector distributions of ORB and CMEMS (Fig. 5b) are similar. ORB covers a larger area in total, but in a few regions only CMEMS provides drift information. The ORB vs. CMEMS comparison gives an error $E = 1641$ m, slope $S = 1.03$ and offset $O = 265$ m (Table 4).

Validating CMEMS using manual data results in the highest root mean square error $E = 1690$ m with slope $S = 0.98$ and offset $O = -415$ m (Table 4).

Decreasing the threshold radius between reference and validation vectors does not influence the error E significantly but reduces the number of found matching vectors, especially when comparing CMEMS and manual vectors.

5 Discussion and outlook

The open-source feature-tracking algorithm ORB (Oriented FAST and Rotated BRIEF) has been tuned for sea ice drift retrieval from Sentinel-1 SAR imagery and used for processing winter and spring data in the ice-covered oceans between Greenland and Severnaya Zemlya. Validating cal-

Table 4. Comparison of ORB, CMEMS and manually derived sea ice drift data from image pair Fram Strait (Table 1). The total numbers of derived vectors are 6920 (ORB), 560 (CMEMS/DTU) and 350 (manual). The # vector pairs is the number of used vector pairs for comparison, i.e. vector pairs with maximum 5 km distance. The average distance refers to the starting locations of the used reference and validation vectors. E is the root mean square error, S and O are slope and offset of the linear fit.

Algorithm	E [m]	S	O [m]	# vector pairs	Average distance [m]
ORB vs. manual	563	1.02	-372	314	1702 ± 1325
ORB vs. CMEMS	1641	1.03	265	436	2261 ± 1247
CMEMS vs. manual	1690	0.98	-415	201	3440 ± 1105

culated drift results against manually derived vectors, we found that our algorithm ($E_{\text{ORB}} = 563$ m) had a distinctly higher accuracy than the drift data set provided by CMEMS ($E_{\text{CMEMS}} = 1690$ m). The given root mean square errors E represent a combination of three error sources:

- error of manual ice drift identification introduced by the sea ice expert
- difference between derived and reference vector due to different geographical location of the starting point (maximum 5 km)
- actual error of the algorithm.

Hence, the actual error of the tuned ORB algorithm is expected to be even lower than 563 m.

As expected, the application of the tuned ORB algorithm is much more efficient than manual ice drift assessment, e.g. 6920 vectors have been calculated within 3 min, whereas identifying 350 sea ice drift vectors manually takes several hours. The number of calculated vectors can be increased by returning a higher number of keypoints (e.g. 1 000 000). However, the processing time increases proportional to the square of the considered keypoints and the algorithm performance becomes suboptimal at some point.

The presented ORB algorithm also outperforms other available feature-tracking algorithms, such as SIFT and SURF not only in processing time, but also in quantity and quality of drift vectors, measured by the two introduced indexes N and D . This proves that ORB is the best option for feature-tracking of sea ice on Sentinel-1 SAR imagery.

The algorithm tuning has been performed using winter and spring data, since our area of interest experiences the highest sea ice cover during this period. During summer and autumn, most considered areas have very little or no ice cover (e.g. Barents Sea and Kara Sea), making ice drift calculation during this period less meaningful. Nevertheless, some areas, like the western Fram Strait, experience sea ice cover during the entire year. Dependence of the algorithm performance on the season needs to be evaluated in future work. Computing

sea ice drift from summer and autumn data is expected to be more demanding, since features might be destroyed by melting.

Comparing the four considered image pairs, Franz Josef Land yields the highest number of HH matches, accompanied by the lowest number from HV channel. A distinctly shorter time difference between the acquisitions (8 h for Franz Josef Land compared to more than 30 for the other image pairs) might be one reason for an improved HH performance. That would conclude that HH features are less preserved over time and increasing the repeat frequency of the satellite (as planned with Sentinel-1B) will improve the algorithm performance, in particular for the HH channel. The sea ice conditions are another important factor when comparing the algorithm performance for different scenes. The image pair Fram Strait includes the marginal ice zone in the eastern part and multiyear ice in the north-west. Not many matches are expected in the marginal ice zone, but the multiyear ice includes more stable deformation pattern, like ridges, that lead to a good feature-tracking performance. The image pair Svalbard North includes a very small part of the marginal ice zone and the major part is comparable homogeneous pack ice with long cracks along a prevailing direction. Franz Josef Land and Kara Sea are clearly less homogeneous and show a mixture of ice floes with different scales and newly formed young ice. This paper has focused on finding the most suitable algorithm for a range of ice conditions found in the considered area and we can give an idea how ice conditions and acquisition time might affect the ORB feature-tracking performance. Further investigations need to be carried out in order to evaluate the algorithm performance for different ice conditions and other areas like the Beaufort Sea or Antarctica.

Komarov and Barber (2014) have evaluated sea ice drift results from dual-polarization Radarsat-2 imagery using a combination of phase and cross-correlation. Comparing the polarization channels, HH is more sensitive to small-scale roughness, whereas the HV channel provides more stable, large-scale features linked to ice topography. Komarov and Barber (2014) concludes that the combination of HH and HV is beneficial, since more reliable vectors are provided and the vector distributions complement each other. They also found that noise floor stripes in the HV images do not affect the motion tracking from pattern matching. We can extend this discussion for feature based algorithms. Using noise removal for HV and angular correction for HH has been tested, but did not improve the feature-tracking results, i.e. a lower number of vectors has been found. Like Komarov and Barber (2014), we recommend the usage of both channels since the vector distributions are complementary. However, using feature tracking, HV provides about four times as many vectors than HH, making HV the more informative channel. The different performance can be explained by a higher variability of the HV backscatter intensity, considering a window with the same size as used for feature description (34×34 pix).

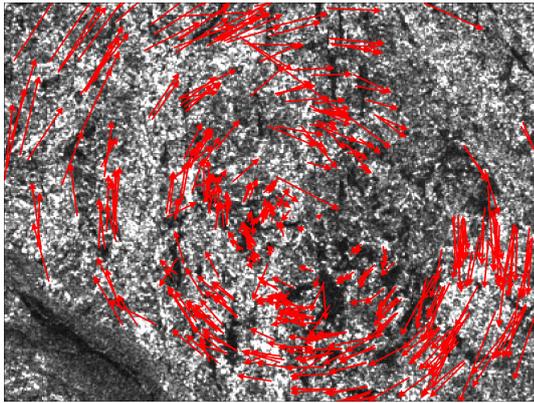


Figure 11. Sea ice drift anomaly (compared to mean drift of the scene) detected in a 300×400 pix (24×32 km) sub-image from Fram Strait (Table 1) close to the marginal ice zone.

Contemporary algorithms for calculating sea ice drift vectors from consecutive image pairs are based either on feature tracking or pattern matching. The feature-tracking approach detects keypoints on two images based solely on the backscatter distribution of the images without taking other keypoints into account. Hence, ORB identifies the keypoints independently. Based on the keypoint locations, the binary feature vectors are calculated. During the second step, all features in the first images are compared to all features in the second image without taking drift information from surrounding vectors into account, i.e. the matching of features from one image to the other is also done independently. Although very close keypoints may share some pixels during the feature description process (i.e. overlap of the considered patches around the keypoints), the detection of keypoints and matching of features are done independently. Eventually, feature-tracking vectors are independent of each other in terms of position, lengths and direction, allowing very close drift vectors to point into different directions.

Figure 11 illustrates 430 drift vector anomalies detected in a 300×400 pix (24×32 km) sub-image from Fram Strait (Table 1) close to the marginal ice zone. The anomalies are calculated as the difference to the mean drift of the entire scene. This example shows that very small-scale dynamic processes, such as the observed rotation, can be detected and quantified with the feature-tracking approach.

Common pattern-matching techniques limit the independence of neighbour vectors for practical reasons. First, pattern matching is usually performed on a regular grid, determining the position and distance between vectors. Second, pattern matching often follows a pyramid approach in order to speed up processing (Thomas et al., 2008a): low-resolution drift is initially estimated using large subwindows and large steps. This first guess constrains the following pattern matching to a finer scale. Repeating this procedure increases the resolution of the end product, but length and

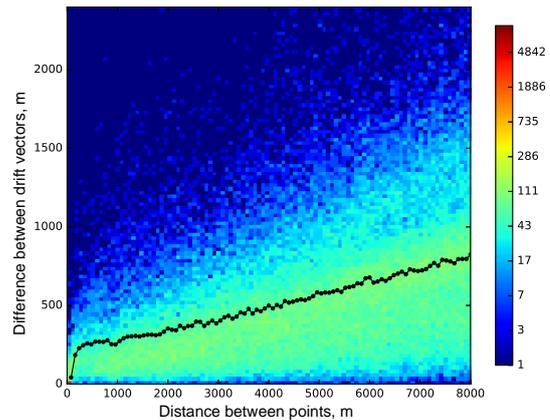


Figure 12. Variogram of drift vectors (black line) on top of 2-D histogram of distance between vectors and difference between vectors estimated from vectors identified on the Sentinel-1 image pair Fram Strait (Table 1). Colour of the 2-D histogram indicates the number of vectors.

direction of the high-resolution vectors depend on the low-resolution estimates, i.e. neighbour vectors depend on each other. Although pattern matching can be designed to retrieve independent vectors by varying the extent of the correlation area and the spacing between vectors, for practical reasons the overlap between the correlation areas is usually half the size of the area (Thomas et al., 2008b).

The independence of feature-tracking vectors has positive and negative implications. On one hand, very close vectors that are independent in length and direction allow identification of ice deformation at very high resolution. The variogram (Fig. 12), which shows how vector differences depend on the distance between them (Cressie, 1993), indicates that very close vectors may differ significantly, although the difference is generally linearly proportional to the distance. On the other hand, feature-tracking vectors are not evenly distributed in space, and large gaps may occur between clouds of densely located vectors. Spatial irregularity is not optimal for systematic detection of divergence and shear zones and calculation of deformation.

Therefore, computationally efficient feature tracking should be complemented by systematic pattern matching to deliver evenly distributed, high-resolution vector fields. Combining the two different drift calculation approaches and making use of the respective advantages is planned as the next step of our research.

Appendix A: Open-source distribution

The presented work is entirely based on open-source software (Python, OpenCV and Nansat) and satellite images with open and free access for all users. Sentinel-1 SAR data can be downloaded at no cost, in near real time under <https://scihub.esa.int/dhus/>. The used programming language is Python, a free and open-source software available under <https://www.python.org>. The OpenCV (open-source Computer Vision) programming library includes the ORB algorithm, and a Python-compatible version can be downloaded under <http://opencv.org>. To handle and read the satellite data, Nansat is used, which is a scientist-friendly Python toolbox for processing 2-D satellite Earth observation data (source code incl. installation description can be found under <https://github.com/nanscenter/nansat>). The presented sea ice drift algorithm including an application example can be downloaded from https://github.com/nanscenter/sea_ice_drift.

Acknowledgements. This research was supported by the Norwegian Research Council project IceMotion (high-resolution sea-ice motion from synthetic aperture radar using pattern tracking and Doppler shift, project number 239998/F50) and by the European-Russian Centre for cooperation in the Arctic and Sub-Arctic environmental and climate research (EuRuCAS) under the European Commission FP7 programme (INCO-LAB), GA#295068. We thank both developer groups of the ORB feature-tracking algorithm and the Earth observation toolbox Nansat. The used satellite data was provided by the European Space Agency. A special thanks to the three anonymous reviewers, Tim Dunker and Wolfgang Dierking (AWI) for their comments and valuable input that improved the paper.

Edited by: D. Notz

References

- Anonymous: <https://earth.esa.int/web/sentinel/technical-guides/sentinel-1-sar> (last access: 25 April 2016), 2014.
- Bay, H., Tuytelaars, T., and Van Gool, L.: Surf: Speeded Up Robust Features, in: *Computer Vision – ECCV 2006*, 9th European Conference on Computer Vision, Proceedings, Part I, 7–13 May 2006, Graz, Austria, 404–417, doi:10.1007/11744023_32, 2006.
- Calonder, M., Lepetit, V., Strecha, C., and Fua, P.: BRIEF: Binary Robust Independent Elementary Features, CVLab, EPFL, Lausanne, Switzerland, 1281–1298, 2010.
- Cressie, N.: *Statistics for spatial data: Wiley series in probability and statistics*, Wiley-Interscience, New York, 15, 105–209, 1993.
- ESA: Sentinel-1 ESA's Radar Observatory Mission for GMES Operational Services, ESA Communications, SP-1322/1, ESA, the Netherlands, 15–21, 2012.
- Harris C. and Stephens M.: A Combined Corner and Edge Detector, in: *Proceedings of the Alvey Vision Conference*, edited by: Taylor, C. J., Alvey Vision Club, 23.1–23.6, doi:10.5244/C.2.23, 1988.
- Hollands, T. and Dierking, W.: Performance of a multiscale correlation algorithm for the estimation of sea-ice drift from SAR images: initial results, *Ann. Glaciol.*, 52, 311–317, 2011.
- IPCC: *Climate Change 2013: The Physical Science Basis*, in: *Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*, edited by: Stocker, T. F., Qin, D., Plattner, G.-K., Tignor, M., Allen, S. K., Boschung, J., Nauels, A., Xia, Y., Bex, V., and Midgley, P. M., Cambridge University Press, Cambridge, UK and New York, NY, USA, 323–335, doi:10.1017/CBO9781107415324, 2013.
- Komarov, A. S. and Barber, D. G.: Sea Ice Motion Tracking From Sequential Dual-Polarization RADARSAT-2 Images, *IEEE T. Geosci. Remote*, 52, 121–136, doi:10.1109/TGRS.2012.2236845, 2014.
- Korosov, A. A., Hansen, M. W., and Yamakava, A.: Nansat – scientist friendly toolbox for processing satellite data, *World Ocean Science Congress*, Korosov, A. A., Hansen, M. W., and Yamakava, A., 2–8 February, Cochin, India, 2015.
- Korosov, A. A., Hansen, M. W., Yamakawa, A., Dagestad, K. F., Vines, A., Riechert, M., Myasoedov, A., Morozov, E. A., and Zakhvatkina, N.: Nansat-0.6.8, Zenodo, doi:10.5281/zenodo.45703, 2016.
- Kwok, R., Curlander, J. C., McConnell, R., and Pang, S.: An Ice Motion Tracking System at the Alaska SAR Facility, *IEEE J. Ocean. Eng.*, 15, 44–54, 1990.
- Lowe, D. G.: Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision*, 60, 91–110, 2004.
- Pedersen, L. T., Saldo, R., and Fenger-Nielsen, R.: Sentinel-1 results: Sea ice operational monitoring, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 26–31 July 2015, Milan, Italy, 2828–2831, doi:10.1109/IGARSS.2015.7326403, 2015.
- Rosin, P. L.: Measuring corner properties, *Comput. Vis. Image Und.*, 73, 291–307, 1999.
- Rosten, E. and Drummond, T.: Machine learning for high-speed corner detection, in: *Computer Vision – ECCV 2006*, 9th European Conference on Computer Vision, Proceedings, Part I, 7–13 May 2006, Graz, Austria, 430–443, doi:10.1007/11744023_34, 2006.
- Rublee, E., Rabaud, V., Konolige, K., and Bradski, G.: ORB: an efficient alternative to SIFT or SURF, *IEEE I. Conf. Comp. Vis. (ICCV)*, 6–13 November 2011, Barcelona, Spain, 2564–2571, doi:10.1109/ICCV.2011.6126544, 2011.
- Schubert, A., Small, D., Meier, E., Miranda, N., and Geudtner, D.: Spaceborne Sar Product Geolocation Accuracy: A Sentinel-1 Update, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 13–18 July 2014, Quebec City, QC, 2675–2678, doi:10.1109/IGARSS.2014.6947025, 2014.
- Thomas, M., Geiger, C. A., and Kambhamettu, C.: High resolution (400 m) motion characterization of sea ice using ERS-1 SAR imagery, *Cold Reg. Sci. Technol.*, 52, 207–223, 2008a.
- Thomas, M., Geiger, C. A., Kannan, P., and Kambhamettu, C.: Streamline Regularization for Large Discontinuous Motion of Sea Ice, in: *2008 IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS 2008)*, edited by: Aksoy, S. and Younan, N. H., International Association for Pattern Recognition, Institute of Electrical and Electronics Engineers, Inc., Tampa, FL, 1–4, doi:10.1109/PRRS.2008.4783171, 2008b.