Manuscript prepared for J. Name
with version 2015/09/17 7.94 Copernicus papers of the LaTeX class copernicus.cls.
Date: 13 February 2017

Response to Referee # 1

'Open-source sea ice drift algorithm for Sentinel-1 SAR imagery using a combination of feature-tracking and pattern-matching'

Stefan Muckenhuber and Stein Sandven

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

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

Dear Referee # 1,

Thank you very much for helping us improving our paper.

Please find here the **answers** to your comments and the corresponding ***changes in manuscript***:

1 General comments

The authors present a new approach for sea ice motion tracking, combining a modified feature tracking algorithm (Muckenhuber, 2016) with a basic pattern matching approach using cross correlation. The authors thereby replace the often used iterative cross correlation approach within an image resolution pyramid by a feature tracking step (which involves a resolution pyramid as well) to predict the search direction for the higher resolution levels of the cross correlation step. A. Berg and L. E. B. Eriksson (2014) presented with their paper on 'Investigation of a Hybrid Algorithm for Sea Ice Drift Measurements Using Synthetic Aperture Radar Images,' based on the combination of pattern matching (cross and/or phase correlation) and feature tracking. In 2014 Komarov and Barber published an algorithm (also referred in this paper), which uses a kind of correlation based feature tracking - since it first identifies characteristic points for the following correlation.

The work from Berg and Eriksson needs to be mentioned and we included

Berg and Eriksson (2014) introduced a hybrid algorithm for sea ice drift retrieval from ENVISAT ASAR data using phase correlation and a feature based matching procedure that is activated if the phase correlation value is below a certain threshold.

Unlike Berg and Eriksson (2014), the feature-tracking step is performed initially and serves as a first guess to limit the computational effort of the pattern-matching step.

To specify the approach from Komarov and Barber 2014, we changed 'pattern-matching' to

***combination of phase/cross-correlation*.**

The idea to combine feature tracking and pattern matching for sea ice drift estimation is tempting and I really like it. It would potentially allow estimating sea ice motion faster and in the case of appropriate feature descriptors even that are rotationally invariant for areas which contain not only translational motion but rotational motion as well. This characteristic can be especially useful in regions like the marginal ice zone, where rotational motion occurs relatively often. However, the devil is in the details.

The idea of study is first step in the direction of a rotational invariant drift algorithm (or at least more robust against rotational motion) for the marginal ice zone and would therefore be worth being published in the Cryosphere after major revisions. However, due to some open questions regarding the implementation of the approach and its validation I cannot recommend its publication at this point. I would like to encourage the authors to continue the work on this interesting idea and resubmit a strongly revised version of this work in the future.

My main concerns are:

1. the suggested logarithmic scaling and its surprising limits (I guess there is something wrong with the calibration routines,)
2. The very vague description of the combination of feature tracking and pattern matching
3. And the slightly irritating validation approach

There is a typo in previous Section 3: the values of the brightness boundaries were given in B and not dB. We corrected that and included a histogram of a representative image pair to illustrate the chosen boundaries. The description of the algorithm has been changed and extended. The validation approach has been changed and the error analysis has been extended. Details are given below.

2 Specific comments

Page 3 Line 62-63 'the resulting vectors are independent of their neighbours [which] is an important advantage ...' - I'm afraid I have to disagree at that point, especially given the implemented feature tracking algorithm. - It has the advantage that it is fast, that it does not get confused by rotational motion and is able to estimate the translational motion even in regions with occurring rotational motion (and that is already great!) but since the employed feature tracking uses a resolution pyramid as well and simply combines all vectors from the different levels of the resolution pyramid, the resulting vectors are neither necessarily all independent nor have the same accuracy (given that some of them are based on a coarser version of the image). Regarding shear and deformation zones, I would claim that a pattern matching algorithm could do the same with an optimised search strategy. Even more problematic, the suggested feature tracking algorithm only identifies a given number of

features for the whole scene. In the worst case, a shear zone or a divergence / convergence zone would not be covered at all, if other features in the scene have a higher score.

With the term 'independent', we wanted to refer to the fact that features are identified without taking the position of other features into account and matched from one image to the other without taking the drift and rotation information from surrounding vectors into account. It is true that features can overlap, the resolution varies due to the resolution pyramid and the independent feature positioning can lead to missing important drift information. We changed the sentences to:

This can be done computationally efficient and the resulting vectors are often independent of their neighbours in terms of position, lengths, direction and rotation, which is an important advantage for resolving shear zones, rotation and divergence/convergence zones. The considered feature-tracking approach identifies features without taking the position of other features into account and matches features from one image to the other without taking the drift and rotation information from surrounding vectors into account (Muckenhuber et al., 2016). However, due to the independent positioning of the features, very close features may share some pixels and since all vectors from the resolution pyramid are combined, the feature size varies among the matches, which implies a varying resolution. In addition, the resulting vector field is not evenly distributed in space and large gaps may occur between densely covered areas, which can eventually lead to missing a shear or divergence/convergence zone.

Page 3 Line 69 'comparable quality estimate for each vector' - I wish there were! There has been a first suggestion by Hollands, Linow and Dierking in 2015 and there is definitely the potential to do so but it is far from being a standard.

We agree and removed this part of the sentence.

Page 3 Line 92 'this data type' - the dual pol version of this data type is only available for the southern part of the Arctic and the Coastal regions and not at all for Antarctica. Since their feature tracking algorithm prefers HV polarisation I wonder if the authors have analysed the results of their algorithm in the case of HH polarisation only to predict a potential performance for the otherwise omitted regions.

The focus of this paper is put on HV, since this polarisation has a better feature-tracking performance and we found a good coverage of this data type in our region of interest. We did not yet analyse the HH performance of the algorithm on a large dataset, but this will certainly be addressed in our future work. We added the following to Section 2:

The introduced algorithm can utilise both HH and HV channel. However, the focus of this paper is put on using HV polarisation, since this channel provides on average four times more feature tracking vectors than HH Muckenhuber et al. (2016), representing a better initial drift estimate

for the combined algorithm.

To further address the polarisation topic, we added the following to Section 5:

The focus of this paper in terms of polarisation was put on the HV channel, since this polarisation provides on average four times more feature tracking vectors than HH and therefore delivers a finer initial drift for the first guess. We found our area of interest well covered with HV images, but other areas in the Arctic and Antarctic reveal a better coverage in HH polarisation. Considering the four representative feature-tracking image pairs from Muckenhuber et al. (2016), the the relatively best HH polarisation performance (i.e. most vectors from HH, while at the same time fewest vectors from HV) was the image pair that showed the least time difference, i.e. 8 h, compared to 31 h, 33 h and 48 h. Therefore, we assume that the HV polarisation provides more features that are better preserved over time. And more consistent features would also favour the performance of the pattern-matching step. However, at this point, this is just an assumption and will be addressed in more detail in our future work.

Utilising the advantage of dual polarisation (HH+HV) is certainly possible with the presented algorithm, but increases the computational effort. A simple approach is to combine the feature tracking vectors derived from HH and HV and produce a combined first-guess. Pattern-matching can be performed based on this combined first-guess for both HH and HV individually and the results can be compared and eventually merged into a single drift product. Having two drift estimates for the same position, from HH and HV pattern-matching respectively, would also allow to disregard vectors that disagree significantly. However, this option would increase the computational effort by two, meaning that the presented Fram Strait example would need about 8 min processing time.

After implementing the presented algorithm into a super-computing facility, we aim to test and compare the respective performance of HV, HH and HH+HV on large datasets to identify the respective advantages.

Page 4 Line 118 'good geolocation accuracy' - I believe I remembered some discussions, that there were some geolocation problems with Nansat earlier, which effected the drift estimation. If I remember correctly: is there a chance that the authors could quantify what 'good' means in this respect?

We discovered drift artefacts in high latitudes between the ground control points before we introduced spline interpolation and reprojection to stereographic. We tested the performance before and after introducing these steps and the artefact disappeared using either one of the steps. To ensure the best possible performance, we apply both steps. The geolocation accuracy depends on the accuracy and amount of ground control points that are delivered in the metadata of the Sentinel-1 scene. At the ground control point the location accuracy should be highest. We cannot give an error value in meter, since we do not have validation

points on the ground and cannot control the accuracy of the Sentinel-1 ground control points. However, from our experience and comparison with buoy drift data, the geolocation accuracy is expected to be in the order of the image resolution.

Page 5 Lines 126 - 135 For a start I would suggest to change the order of the explanation and first mention the conversion from linear to log scale before the authors mention the scaling to integer values between 0 - 255 but this is the easier part. The more difficult part might be that we have a problem if there are no typos in these lines and I understood everything correctly. Log(0.013) = -1.88 dB while log(0.08) = -1.1 dB. If their minimum backscatter values are in dB as well (units missing!), it would mean, that the authors only use the range between -3.25dB – -1.88dB for HV and the range between -2.5dB – -1.1 dB. Could the authors please comment on this and even rephrase this part if I just misunderstood the authors? The problem I see is that their chosen backscatter range only represents a minor part of the backscatter range to be expected for sea ice in the logarithmic scale. If these are the correct numbers, the authors might as well want to check the calibration routines for their data.

We changed the order of explanation and first mention the conversion from linear to log scale before the scaling to integer values. There has been a typo with regards to the units: the values of the brightness boundaries were given in B and not dB. We corrected that, added the units and included a histogram of a representative image pair (Figure 1) to illustrate the chosen boundaries and show the representative image pair after the conversion into the integer range (Figure 2).

Page 6 Line 166 'serves as a quality estimate of the matching performance' - After it has been shown by Hollands, Linow and Dierking (2015) that there is no relation between the matching error and the correlation coefficient I would prefer a proof why the authors can use it as a quality measure. Even their Fig. 7 shows that the authors also dismiss good values, using the correlation coefficient as a quality value. Admittedly there is a group of large error values in their histogram but I wonder if this is significant. A correlation coefficient is only meaningful if the respective texture is characteristic enough. - I suggest to google Anscombe's quartet.

We agree and removed the claim that the cross correlation value could serve as quality estimate. However, using the considered validation data, we found for our data type, time period and area of interest that the probability for large errors decreases with increasing cross correlation value. The following was added to Section 3:

We found that the probability for a large D value (representative for the error) decreases with increasing maximum cross coefficient value MCC. Therefore we suggest to exclude matches with a MCC value below a certain threshold MCC_{min}. This option is embedded into the

algorithm, but can easily be adjusted or turned off by setting $MCC_{min} = 0$***.***

Page 6 Line 173 - 176 'To filter outliers, ... removed' - I have to admit, it would help me, if the authors could describe this outlier removal in more detail - based on the current description it is difficult to evaluate what the authors actually did.

The algorithm description has been changed and more details have been included. A subsection 'II Filter' and Figure 4 have been added to describe and illustrate the filtering process.

Page 6 Line 177 - 181 'The remaining feature vectors ... neighbouring feature tracking vectors' - Just for the better understanding: What happens if there is a large area with no vectors at all framed by a few sparse vectors. Would the authors just triangulate over the whole area (potentially containing deformation or shear zones)?

If the considered area lies in between three feature tracking vectors, we triangulate over the area to provide the first guess. This initial drift estimate however, will then be adjusted by the pattern-matching approach. If the closest feature-tracking vector is far away, we apply the lowest restrictions defined by d_{max}. We found a useful value for d_{max} for our area and time period of interest to be 100 pixels, meaning that the search area is defined by an 8 km radius around the first guess. The lowest restrictions can easily be adjusted according to expected ice conditions and computational performance.

Page 6 Line 181-183 'To provide a drift estimate ... combination of x1 and y1.' - similar to Line 173 - 176 it is hard to say, what the authors actually did. May be the authors could add some details, making it easier to follow.

The algorithm description has been changed and more details have been included. A subsection 'III First guess' and Figure 5 have been added to describe and illustrate the process that leads to the first guess.

Page 6 Line 187- 190 I find it a bit confusing that we have a given size of the window before it is tuned. The same is true for dmin and dmax: It only became clear when I reached section 3.3. I would suggest that the authors mention here that they are going to identify the optimal parameters and may be as well why the authors decided to choose formula (4) for the window size.

We changed the description of the pattern-matching step and adjusted the order according to this comment. We clarify which parameters need to be specified for the introduced pattern-matching procedure and how we find the recommended setting for each parameter.

Page 7 Figure 1 It would be interesting so see a SAR image for the same area and may be a drift vector field. Is it correct that there is land where the distances are low and sea ice where the distance colour scale is saturated- Actually the authors already anticipate a result of their parameter tuning here. That makes it difficult to read. May be the authors should reorganise this part.

We added Figure 2 to illustrate the related SAR images and show the corresponding feature-tracking vectors in Figure 3. The colour scales of the left and middle panel represent the first guess of the end positions on SAR_2 and the colour scale of the right panel indicates the distance to the closest feature tracking vectors, i.e. values of $d = 10$ represent 0-10 pixel distance to the closest feature-tracking vector and values of $d = 100$ represent 100-∞ pixel distance. We split the figure and changed the algorithm description accordingly to make the process better understandable.

Page 7 Line 195 '-beta +beta with step delta beta' - it is confusing that the authors suddenly start to introduce rotation as well since it has not been mentioned beforehand. The authors should have at least introduced it in section 3.2 II.

The algorithm description has been changed accordingly and rotation is introduced in the new subsection 'IV Pattern-matching'.

Section 3.2 page 5-7 Given that this section is meant to be the innovative part of this study I suggest restructuring it, to make it more concise. Right now, it is quite confusing and has varying level of detail and order (e.g. the window size question is a specific cross correlation question. I would urge the authors to state clearly when they introduce a parameter which they want to tune in the later course of the paper. Additionally I would suggest adding a flow chart, highlighting the steps, described in this paper.

We adjusted the algorithm description accordingly and added more details. Figure 3 includes a flow chart and respective example images to illustrate the algorithm steps and the resulting products.

Page 8 Formula 6 Why did the authors choose this distance measure instead of the RMSD in Formula 5?

The $RMSD$ equation (previous Equation 5) and the comparison to the manually drawn vectors have been removed. The distance measure D (previous Equation 6) has been used to get an individual error value for each compared vector pair, consisting of one validation vector and one algorithm vector. Since we found a logarithmic error distribution for the buoy comparison, a mean value as expressed by the $RMSD$ does not represent the found distribution.

Section 4.1 Honestly, I would suggest skipping this section - it is not surprising that the logarithmic scaling leads to a higher number of features since the logarithmic histogram scaling favours the structures in the sea ice which are mainly represented in the shadow and medium backscatter values but hardly in the highlights.

We skipped this Section and briefly mention in the data pre-processing why a logarithmic distribution is used:

Using a logarithmic scaling provides a keypoint distribution for the feature tracking algorithm that depends less on high peak values, while the total number of vectors increases.

Page 10 Section 4.2 / Table 2 I have various questions:

- I understood that the authors tuned their Influence domain parameter dmax based on one image pair over Fram strait as well as the side length for their template but how did the authors tune their Dmin value and the MCCmin value?

The parameter tuning was removed from the manuscript. Instead, useful restrictions that limit the computational effort of the pattern-matching were found and a useful MCC_{min} value was found according to the error distribution from the buoy comparison.

- 70 x 70 pixel for t1 means that their correlation window covers an area of approx. 6.3 x 6.3 km - how does this go along with their claim to resolve deformation and shear zones?

We agree that this resolution is not sufficient and changed the recommended setting to 34×34 pixels in order to be consistent with our goal. We added the following to Subsection 'IV Pattern-matching':

The size of the small template $t_{1s} \times t_{1s}$ defines the considered area that is tracked from one image to the next and hence, affects the resolution of the resulting drift product. In order to be consistent with the resolution of the feature-tracking step and achieve our goal of a sea ice drift product with a spatial scaling of less than 5 km, we use the size of the feature-tracking patch of the pyramid level with the highest resolution to define the size of t_1. That means, we use $t_{s1} = 34$ pixels (2.7 km).

- Since their influence domain influences the size of their search window t2 it would mean that the authors add a degree of freedom of +/-1.8 to +/-11.25 km to their first feature tracking based guess, which would push their 0.5 m/s maximum ice drift limit for the feature tracking to about 0.6 m/s - right? Its contribution would however vary depending on the time span between both images of the scene. For the same constant drift velocity (but speed variations with in the scene), an image pair with a longer time span would then show larger displacement differences with in the scene while having the same maximum degree of freedom of +/- 11.25 km like an image pair that has been acquired at the same day - this might cause a problem, don't the authors think?

Yes, this is a good point and needs to be considered. We added the following to Section 5:

The current setting of the feature-tracking algorithm applies a maximum drift filter of 0.5 m/s. We found this to be a reasonable value for our time period and area of interest. However, when considering extreme drift situations in Fram Strait and a short time interval between image acquisitions, this threshold should be adjusted.

During a KV Svalbard cruise in summer 2016, we deployed three GPS tracker in Fram Strait that recorded their positions with a temporal resolution of 5-30 min between 8^{th} July until 9^{th} September 2016 in an area covering 75° N to 80° N and 4° W to 14° W. Considering the displacements with 30 min interval, we found velocities above 0.5 m/s on a few occasions, when the tidal motion adds to an exceptionally fast ice drift.

The GPS data from the hovercraft expedition FRAM2014-2015 (https://sabvabaa.nersc.no), that was collected with a temporal resolution of 10 s between 31^{st} August 2014 until 6^{th} July 2015, did not reveal a single 30 min interval during which the hovercraft was moved by ice drift more than 0.45 m/s. The hovercraft expedition started at 280 km south from the North Pole towards the Siberian coast, crossed the Arctic Ocean towards Greenland and was picked up in the north-western part of Fram Strait.

In case the estimated drift from feature-tracking reaches velocities close to 0.5 m/s, the pattern-matching step might add an additional degree of freedom of up to 8 km, which could eventually lead to a higher drift result than 0.5 m/s, depending on the time interval between the acquisitions. The smaller the time difference, the larger is the potentially added velocity. In order to be consistent when combining the drift information from several image pairs with different timings, one should apply a maximum drift filter on the final drift product of the presented algorithm that has the same maximum velocity as the feature-tracking filter. The corresponding function is implemented in the distributed open-source algorithm.

Page 10 Section 4.3 line 249: 'on a grid with 8 km spacing' - I suggest to summarize the information of their resulting product somewhere. It is not necessarily obvious to find the information on their grid spacing in the Parameter tuning and Computational Efficiency Section.

The considered grid is not meant as a given parameter of the resulting product, but serves only to provide an estimate for the computational efficiency of the presented approach. The points of interest, given in longitude and latitude, represent the input for the algorithm. This can be the position of a ship, the grid of a model or an evenly spaced grid with any wanted resolution. The algorithm includes a routine that can derive points of interest in lon-lat on an evenly spaced grid. We hope that the changed algorithm description improved the explanation regarding points of interest and considered grid.

Page 10 Section 4.3 Given the resolution of 8 x 8 km even pattern matching only based algorithms show a similar performance or even better. But I admit that the robustness to rotational motion is very useful in the marginal ice zone, where many of the pure pattern matching algorithms fail.

We changed the resolution of the example and put more focus on the rotational motion.

Page 10 Section 4.4 line 261: What does the size of 34 pixel mean- Is the feature described as a patch of 34 x 34 side length- May be the authors should add a short explanation to their feature tracking part on page 5.

We removed this Section and added a description of the considered feature patch sizes to Subsection 'I Feature-tracking'.

Page 12 Line 268-271: Why do the authors choose a minimum Cross Correlation Coefficient of 0.35? If the authors found a logarithmic function their distance distribution seems to follow, the authors could name it. Otherwise less strict term would be that the distance distribution seems to show a logarithmic behaviour or something like this. A peak at 300m is not necessarily meaningful (e.g. what would be the peak without their Cross Correlation Threshold? How many drift vectors form a peak?) but even if the authors have a peak, it does only represent the systematic component of the error and not the random one. In order to identify the distribution I would suggest smoothing the histogram and fitting a distribution to it.

We smoothed the histogram and fitted a logarithmic normal distribution to it. We found the chosen minimum cross correlation coefficient $MCC_{min} = 0.4$ by plotting MCC values against distance D, that represents the error. This is now shown in Figure 11.

Page 14 Table 4: I would think that it is not the best approach to validate an algorithm based on the drift vectors I tuned it to. For a real validation the authors need at least another independent image pair with an independent set of manually derived drift vectors. I would strongly encourage the authors to change this! The authors compare apple with oranges if the authors compare an algorithm tuned to this specific scene with algorithms like the one from CMEMS. Additionally it would be great, if the authors could quantify both systematic and random error.

We removed the parameter tuning and do not compare our results against the manually drawn vectors anymore. Figure 11 is included to illustrate systematic and random error.

Page 14 Line 290: 'To further estimate the accuracy of the algorithm ...' - here it would be interesting to see, how the other algorithms perform as well. Additionally it would be great, if the authors could quantify both systematic and random error. The authors might want to check the regular validation document for the CMEMS ice drift as a start: http:myocean.met.noSIW-TACdocmyo-wp14-siw-dtu-icedrift-glob-obs-validation_latest.pdf The peak of a distribution is no error value!

We illustrate the error in Figure 11 according to this suggestion. We removed the comparison with CMEMS and simple feature tracking, since we don't have drift results of these two algorithms at the buoy locations.

Page 14 Line 302-303: 'Hence, ... image resolution' I agree there are various factors influencing the result of the algorithm and thereby influencing the validation but I cannot agree with this statement. It might be but the authors have not shown this yet!

We removed these two sentences.

3 Technical corrections

Page 1 Line 5: 'respective advantages of the two approaches' - the authors should emphasise in more detail what the advantages are, since this is the basic justification for this paper and this not only in the abstract but in the introduction/motivation as well

The feature-tracking and pattern-matching description in Section 1 has been improved and the corresponding part in the abstract has been changed to:

Feature-tracking produces an initial drift estimate and limits the search area for the consecutive pattern-matching, that provides small to medium scale drift adjustments and normalised cross coefficient values. The algorithm is designed to combine the two approaches in the most meaningful way in order to benefit from the respective advantages. The main advantages of the considered feature-tracking approach are the computational efficiency and the independence of the vectors in terms of position, lengths, direction and rotation. Pattern-matching on the other side allows better control over vector positioning and resolution.

Page 3 Line 37 'covers the Arctic every week with a spatial resolution of 5 km' - I'm not sure but the authors might want to check it: as far as I know the, RGPS covers a large part of the Western Arctic Ocean but not the entire Arctic, due to the acquisition area of Radarsat. Up to my knowledge, the 5 x 5 km spatial resolution is a gridded drift field, which does not necessarily represent the actual spatial resolution, given that the RGPS searches features in a 10 or 25 km grid respectively. See also the RGPS Data User-s Handbook (Fig. 1 and Fig. 2)

We changed the sentence to:

The geophysical processor system from Kwok et al. (1990) has been used to calculate sea ice drift fields in particular over the Western Arctic (depending on SAR coverage) once per week with a spatial resolution of 10-25 km for the time period 1997–2012. This extensive dataset makes use of SAR data from Radarsat-1 and ENVISAT (Environmental Satellite).

Page 3 Line 73 'respective advantages' - If possible, be clearer about the respective advantages and summarise them here together with the disadvantages the authors still have and those the authors bypass with their approach.

The feature-tracking and pattern-matching description in Section 1 has been improved and in addition, we added to Section 1:

The main advantages of the considered feature-tracking approach are the computational efficiency and the independence of the vectors in terms of position, lengths, direction and rotation. Pattern-matching on the other side allows better control over vector positioning and resolution, which is a necessity for computing divergence, shear and total deformation.

Page 2 Line 44 'pattern-marching and feature tracking respectively' - even terms are somehow flexible: I would claim, that Komarov and Barber do somehow a basic feature tracking as well, since they identify features, with certain characteristics before the correlate them - in that way, they have implemented the search for descriptors in a way. The use of correlation does not necessary mean that the approach is a pattern matching approach, since the correlation itself is the distance measure only, that is used to assess how similar a feature or a pattern is, compared to the reference. It might be a bit pedantic, but the authors might still want to give it a second thought.

We changed the sentence to:

Komarov and Barber (2014) and Muckenhuber et al. (2016) have evaluated the sea ice drift retrieval performance of dual-polarisation SAR imagery using a combination of phase/cross-correlation and feature-tracking based on corner detetction respectively.

Page 2 Line 52-55 'Making use ... Copernicus.eu).' - I agree, that it is an important product, which should definitely be mentioned in the frame of this article but I think, the statement does not really fit there where it is right now because it interrupts their motivation.

We moved the sentence into the paragraph above.

Page 6 Line 185 -186 'Figure 2 shows...' - I would suggest moving the sentence a few sentence down to Line 195 after '...correlation value is returned

The method description has been restructured taking this comment into account. We refer to this Figure at a later point in the description.

Page 10 Section 4.3 line 252-254: 'NB: The vectors near ... treated with caution' - I completely agree but it is no question of computational efficiency

This sentence has been removed.

Page 10 Section 4.4 line 256: Strictly speaking the authors should compare their estimated drift vectors to their manually derived vectors and not the other way round and the authors estimate a drift vector and do not calculate it but this is a minor technical issue I guess.

The comparison with the manually drawn vectors have been removed and we took this comment into account, when describing the comparison to the GPS buoy dataset.

Page 11 Table 3: Is it correct, that their drift estimation is only based on HV polarisation- I guess the authors should state it somewhere in the beginning. Given their experience with dual pol motion tracking, I assumed that the authors used both polarisations here as well- I suggest being clearer about it from the beginning, if this is the case.

Yes, the considered drift estimates in this work are based on HV. We added the following to Section 2:

The introduced algorithm can utilise both HH and HV channel. However, the focus of this paper is put on using HV polarisation, since this channel provides on average four times more feature tracking vectors than HH (Muckenhuber et al., 2016), representing a better initial drift estimate for the combined algorithm.

Page 15 Line 311: 'The parameters can easily be varied...' - a short tabular overview on the range for the individual parameters and their effect on the algorithm performance would be nice even though probably difficult.

We updated the algorithm description taking this comment into account. The parameters and their effect on the drift result are now explained more in detail for a better understanding of eventual changes from the recommended setting. The possible range of the parameters t_{s1}, t_{s2}, β, $\Delta\beta$, d_{min} and d_{max} is not limited.

Page 15 Line 329: 'the real sea ice velocity' - the velocity the authors observe is not wrong, they might underestimate the speed and its variation as well as the variation of the drift direction but velocity is defined as distance per time, and the resulting velocity vector, being a sum of velocity vector variations over the observation interval is the resulting velocity vector. A higher temporal resolution is interesting but it is as interesting and influences the 'realness' of their velocity vector the same way higher spatial resolution does. It would be great if the authors could give this phrase a second thought.

We removed the term 'real sea ice velocity' and changed it to *sea ice displacements with higher temporal resolution*, that *reveal more details e.g. rotational motion due to tides*. Section 5 has been updated accordingly.

Thanks again for your comments. We are looking forward to your reply!

Best regards,

S. Muckenhuber and S. Sandven

Manuscript prepared for J. Name
with version 2015/09/17 7.94 Copernicus papers of the LaTeX class copernicus.cls.
Date: 13 February 2017

Response to Referee # 2

'Open-source sea ice drift algorithm for Sentinel-1 SAR imagery using a combination of feature-tracking and pattern-matching'

Stefan Muckenhuber and Stein Sandven

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

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

Dear Referee # 2,

Thank you very much for helping us improving our paper.

Please find here the **answers** to your comments and the corresponding ***changes in manuscript***:

1 Major comments

Introduction, P2: The manuscript should include some additional background information on the sea ice drift in the study area (with possible references): what are the magnitudes of typical ice drift in the study area and whole Arctic (e.g. cm/s and daily) and in which areas they are located and which are their causes?

We added references and the following to Section 1:

Early work from Nansen (1902) established the rule-of-thumb that sea ice velocity resembles 2 % of the surface wind speed with a drift direction of about 45° to the right (Northern Hemisphere) of the wind. This wind driven explanation can give a rough estimate for instantaneous ice velocities. However, the respective influence of wind and ocean current strongly depends on the temporal and spatial scale. Only about 50 % of the long-term (several months) averaged ice drift in the Arctic can be explained by geostrophic winds, whereas the rest is related to mean ocean circulation. This proportion increases to more than 70 % explained by wind, when considering shorter time scales (days to weeks). The wind fails to explain large-scale ice divergence patterns and its influence decreases towards the coast (Thorndike and Colony, 1982).

Using GPS drift data from the International Arctic Buoy Program (IABP), Rampal et al. (2009) analysed the general circulation of the Arctic sea ice velocity field and found that the fluctuations follow the same diffusive regime as turbulent flows in other geophysical fluids. The monthly mean drift using 12 h displacements was found to be in the order of 0.05 to 0.1 m/s and showed a

strong seasonal cycle with minimum in April and maximum in October. The IABP dataset also revealed a positive trend in the mean Arctic sea ice speed of +17 % per decade for winter and +8.5 % for summer considering the time period 1979–2007. This is unlikely to be the consequence of increased external forcing. Instead, the thinning of the ice cover is suggested to decrease the mechanical strength which eventually causes higher speed given a constant external forcing (Rampal et al.; 2009b).

Fram Strait represents the main gate for Arctic ice export and high drift velocities are generally found in this area with direction southward. Based on moored Doppler Current Meters mounted near 79° N 5° W, Widell et al. (2003) found an average southward velocity of 0.16 m/s for the period 1996–2000. Daily averaged values were usually in the range 0–0.5 m/s with very few occasions above 0.5 m/s.

Method/Feature tracking, P5: It is mentioned that 'The best match is accepted if the ratio of the two shortest Hamming Distances is below 0.75.'. Explain why this is done and how the threshold was selected. Probably to reduce possibilty of similarization errors? What is magnitude of typical Hamming distances? If they are small, then 0.75 has quite different meaning than for larger values. I assume that the ratio is the ratio of the shortest and second shortest Hamming distance (also write this in the text).

The Hamming distances are embedded in the feature-tracking algorithm and are not returned during application of the algorithm. This makes the evaluation of the value distribution difficult. However, based on visual interpretation of drift results using different Hamming distances, Muckenhuber et al. (2016) found a suitable value for our time period and area of interest. We added the following to Section 3:

The best match is accepted if the ratio of the shortest and second shortest Hamming distances is below a certain threshold. Given a suitable threshold, the ratio test will discard a high number of false matches, while eliminating only a few correct matches.

Muckenhuber et al. (2016) found the most suitable parameter setting for our area and time period of interest, including a Hamming distance threshold of 0.75, ...

Method/Combination, P6: To filter outliers each vector is simulated using two functions which are LS solutions... This need more explanation. Why third degree polynomial has been used and which data are used in the LS fit? Also in the extrapolation is also performed using a LS solutions. Also describe this in more detail. How is the traingulation constructed (Delauney?) in interpolation?

Section 2 has been changed according to this comment. We included more detailed descriptions of LS solutions and triangulation. Equations were added to specify the procedure. The

Parameter Tuning/Validation, P10: It is not exlpicitly mentioned which data were used for the parameter tuning. Were all the vlaidation data used for this? Then the validation with this data set is not fair as the algorithm has been tuned for this data. Then only the buoy data can be used for independent validation. Or if separate sets are used for parameter tuning and validation, indicate this in the manuscript.

The Parameter Tuning Section has been removed and validation is now only done against buoy data.

2 Detailed comments

P1L2: 'computanional' -> 'computationally'

Agree, we changed the manuscript accordingly.

P2L33: '90s' -> '90's'

Agree, has been changed.

P2L38: In the case of ENVISAT, rather give the name of the instrument i.e. ENVISAT ASAR, could also mention that RADARSAT was an instrument of CSA and ENVISAT ASAR of ESA.

Agree, has been changed.

P3L88: '((' -> '('

Agree, has been changed.

P3L88: '...dual polarization support...' '..also in wide swath mode'. Also earlier instruments had a possibility to measure multiple polarizations but the covered area was small. This has been changed by RADARSAT-2 and SENTINEL-1.

We changed the sentence to:

The mission includes two identical satellites, Sentinel-1A (launched in April 2014) and Sentinel-1B (launched in April 2016), each carrying a single C-band SAR with a centre frequency of 5.405 GHz and dual-polarisation support (HH+HV, VV+VH) also for wide swath mode.

P3L90: Give also the acronyms for the mode i.e. EW GRDM (thes are generally used by ESA in documentation and file names).

Agree, has been added.

P4L125: You can remove 'of 93m range x 87m azimuth', this information has already been given earlier.

Agree, has been removed.

P6 eq. 3: Here You give the formula for NCC. Also give the drift (dx,dy) detection as a formula, something like: $(dx, dy) = argmax_(k,l)inWNCC(x+k, y+l)$ Is NCC computed according to this equation or by applying FFT and IFFT (which has been applied in many algorithms to fasten the computation)?

The matrix NCC is computed according to the new Equation 8 and FFT and IFFT are not applied. The used python function is matchTemplate from OpenCV (http://docs.opencv.org/2.4/modules/imgproc/doc/object_detection.html). We changed the pattern-matching description according to this comment and added the following:

The highest value in the matrix NCC, i.e. the the maximum normalised cross coefficient value MCC***, represents the location of the best match and the corresponding location adjustment is given by*** dx ***and*** dy***.***

$$(\frac{1+t_{s2}-t_{s1}}{2}+dx, \frac{1+t_{s2}-t_{s1}}{2}+dy) = argmax(\textbf{\textit{NCC}}(x,y) \quad (1)$$

P6 Eq. 4: Define 'side' in the text.

This phrase has been removed and replaced by t_{1s} **and** t_{2s}**.**

P7 Fig 1 and Fig 2. Use a, b, and c for the subfigures and to refer to them.

We added titles to the subfigures to refer to them and make the algorithm description easier understandable.

P7 L195: Explain here what is denoted by 'beta'. It is is also in Fig. 2 caption.

The algorithm description has been changed according to this comment and the following has been added:

To account for rotation adjustment, the matrix NCC is calculated several times: template t_1 ***is rotated around the initially estimated rotation*** α ***from*** $\alpha-\beta$ ***to*** $\alpha+\beta$ ***with step size*** $\Delta\beta$***. The angle*** β ***is the maximum additional rotation and represents therefore the rotation restriction. The NCC matrix with the highest cross coefficient value*** MCC ***is returned.***

P7 Fig.2 (and text): Why rectangular/square templates has been used? A circular template would be much easier (symmetric) to rotate. Consider using a circular templates instead.

We agree with this comment. Regarding t_1 **however, the current version of the used OpenCV function matchTemplate does not allow circular templates and work-arounds would influence the result and the computational efficiency. We hope that a later version of matchTemplate will allow to use masks. Regarding** t_2**, we included a circular mask for the matching result to**

limit the search area to a circle rather than a square.

Logarithmic scaling P8-9: I think logarithmic scale is the typical presentation of SAR sigma0 and often a fixed scaling to gray tone imagery is used for SAR imagery, e.g. scaling between -30dB -> 0 dB. You could mention this fact on the manuscript. This also leads to the question if any other 'scaling' would produce even better results, e.g. applying some king of histogram derived image mapping (e.g. simple histogram equlization etc.). This could be one topic for further development.

We agree. The corresponding section has been removed and the logarithmic scaling description has been moved to Section 3 and adjusted according to this comment. Muckenhuber et al. (2016) tested different scalings procedures on four representative image pairs to retrieve the best possible feature-tracking results. We apply the same scaling for pattern-matching for both computational efficiency and because we assume that a scaling that is preferable for feature-tracking is also preferable for pattern-matching. This assumption however, has not been proven and is certainly a topic for further development.

P10/Computational efficiency: You give a time of less than 3.5 minutes here. Is this a typical execution time or just execution time for a randomly selected example. Could you give average execution times and deviations or maybe estimate for the worst case? Does the execution time increase linearly as a function of the number of vectors or is there some other kind of relationship?

The given time is representative for an image pair with large overlap, good coverage with feature-tracking vectors and the given resolution of the final product. We adjusted and extended the Section Computational efficiency according to this comment. The step 'II Pattern-matching and III Combination' is proportional to the number of chosen points of interest, i.e. the number of drift vectors of the final product. The first two steps can be seen representative for all Sentinel-1 image pairs with 400×400 km coverage. We added a corresponding analysis of the different steps and the influencing parameters.

P12 L270-271: also give the average D. 'peak' is not a correct word here, the histogram/distribution has many peaks, possibly You could use 'mode' here and also in the caption of Fig. 7.

We agree. The error estimation has been changed accordingly. We now fit a logarithmic normal distribution to the histogram and found a median $e^{\mu} = 341.9$ m.

P12 L276-277: The DTU method has not been documented very well in any publications I think. Also the reference given does not say much. I suppose there is not better reference for this?

The comparison with the DTU drift field in Fram Strait has been removed. The DTU product however, is still mentioned in Section 1. We did not find any better reference than

Pedersen et al. (2015), http://www.seaice.dk/ and http://marine.copernicus.eu.

P12 L279: '...used the nearest neighbors...' -> '...used the nearest neighbors (NN's)...' then NN can be used in Table 4.

We agree. However, the corresponding comparison using image pair Fram Strait has been removed.

P13 Fig. 6: Would it be possible to indicate the location of the detail in the coarse-scale image (without causing too much damage for the image)?

We agree. However, the corresponding image has been removed.

P14 Table 4: 'Average distance' -> 'Average NN distance' or something like that. Are the values after +- sign standard deviations or some multiples of standard deviation or something else? Include this information in the table or caption.

The +- sign indicated one standard deviation. However, the corresponding comparison using image pair Fram Strait has been removed.

Discussion: What is the possible error magnitude of the manually estimated drift (is it assumed to be sub-pixel, one pixel or more and what kind of possible error sources these vectors include?)?

The estimated error is in the order of several 100 m. However, the corresponding comparison using image pair Fram Strait has been removed.

P15 L319-320: Also ESA is going to improve their thermal noise removal by including more measurements along the azimuth direction. Probably this also could be mentioned. If necessary you can get more information on this from Nuno Miranda at ESA (nuno.miranda@esa.int).

Thank you for this information. We contacted Nuno Miranda from ESA and added the following to Section 5:

The European Space Agency is also in the process of improving their thermal noise removal for Sentinel-1 imagery. Noise removal in range direction is driven by a function that takes measured noise power into account. Until now, noise measurements are done at the start of each data acquisition, i.e. every 10-20 minutes, and a linear interpolation is performed to provide noise values every 3 seconds. The distribution of noise measurements showed a bimodal shape and it was recently discovered that lower values are related to noise over ocean while higher values are related to noise over land. This means, that Sentinel-1 is able to sense the difference of the earth surface brightness temperature similar to a passive radiometer. When the data acquisition includes a transition from ocean to land or vice versa, the linear interpolation fails to track the noise variation. The successors of Sentinel-1A/B are planned to include more frequent noise

measurements. Until then, ESA wants to use the 8-10 echoes after the burst that are recorded while the transmitted pulse is still travelling and the instrument is measuring the noise. This will provide noise measurements every 0.9 seconds and allows to track the noise variations in more detail. In addition, ESA is planning to introduce a change in the data format during 2017 that shall remove the noise shaping in azimuth. These efforts are expected to improve the performance of the presented algorithm significantly.

We thank Nuno Miranda in the Acknowledgement for the provided informations.

Thanks again for your comments. We are looking forward to your reply!

Best regards,

S. Muckenhuber and S. Sandven

Manuscript prepared for J. Name
with version 2015/09/17 7.94 Copernicus papers of the LaTeX class copernicus.cls.
Date: 13 February 2017

Response to Referee # 3

'Open-source sea ice drift algorithm for Sentinel-1 SAR imagery using a combination of feature-tracking and pattern-matching'

Stefan Muckenhuber and Stein Sandven

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

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

Dear Referee # 3,

Thank you very much for helping us improving our paper.

Please find here the **answers** to your comments and the corresponding ***changes in manuscript***:

General comment

A general impression after reviewing this manuscript is that it requires more work and provision of additional details before being ready for publication in TC. The authors are thus invited to revise their manuscript before a new version is submitted. Specifically, the following items should be addressed.

We increased the level of detail and added new figures to improve the manuscript.

1 Description of the algorithm

The 'pattern-matching' step is not well enough described and many questions are still open at the end of section 3.2.

The pattern-matching description has been rewritten and more details have been added.

1.a) The ordering of the sub-sections (I. Feature-Tracking, II. Pattern-matching, III. Combination) is maybe not optimal as you spend some of Section III to describe the rotation by angle beta (that should really go into II). Maybe it would be easier to follow if the sub-section followed the steps of the algorithms (feature-matching, fitting of polynomial for first-guess, filtering, patter-matching, etc...).

We changed to order according to this comment. The new subsections are: 'I Feature-tracking', 'II Filter', 'III First guess', 'IV Pattern-matching' and 'V Final Product'. We added Figure 3 incl. flow chart to illustrate the steps and the respective products.

1.b) It is unclear if your pattern-matching step features a series of x,y shifts to maximize the cross-correlation in addition to the rotation by beta, or not. If you combine both x, y, and beta shifts, what is the relative order and does it matter?

The pattern-matching description has been changed according to this comment. The matrix NCC(x,y), containing all normalised cross coefficient values for all possible x,y shift, is calculated several times: one for each rotation β. The highest cross coefficient value is found considering all NCC matrizes.

1.c) As you recall in I. 'Feature-Tracking', the ORB algorithms also gives an information about the rotation angle (delta between centroid-based orientation of the matched features). Is this feature-matching first-guess of the rotation used at all? If yes, how; and if no, why not?

This is a good point and we adjusted the algorithm according to this comment. We included the usage of the feature-tracking rotation: a filtered rotation field based on the rotation found for the individual features serves now as initial rotation for the pattern-matching step.

1.d) What is 'the initial rotation between the two Sentinel-1 image' (line 194) and how is it computed? Is it the same value across the image?

The 'initial rotation between two Sentinel-1 images' was derived as angle between the left edges of the images. It was calculated by re-projecting the left edge of the second image onto the projection of the first image. This is the same value for the entire scene. However, after including the feature-tracking rotation (see above), the algorithm is not using this rotation anymore, but rather a rotation field, that varies over the image (see α in Figure 5), based on the rotation of the individual features.
The 'initial rotation between two Sentinel-1 images' is still calculated since it allows to exclude the different projections of the two scenes and derive the actual rotation of the sea ice at each point of interest.

1.e) In subsection II. 'Pattern-matching' you write the NCC formula for 'two equally sized windows'. But later you seem to use two unequally sized windows (size t1 in SAR1, size t2 in SAR2). What is the NCC formula do you then use? Of is size t2 related to the size of the search window while t1 is the size of the pattern? The questions above are mostly to give an impression of the level of details expected when you re-formulate this section. Your first manuscript contained quite some details on the methodology, and this new one requires at least as many details.

We changed the pattern-matching description according to this comment and included a more detailed formulation of the NCC equation.

2 Validation against GPS data

2.a) The choice of validation metric (the distance between the end points of the reference and estimated vectors) is not peculiar. Virtually all other studies use the RMSE along two components (e.g. u and v). And the logarithmic distribution of the errors is not discussed or exploited. Please also discuss the RMSDs in u and v components and compare your results with that of other investigators.

We changed the validation procedure and fitted a logarithmic normal distribution to the histogram. We did not see any specific pattern when considering u and v component separately, but we added plots to further investigate the systematic and random error (Figure 11). To our knowledge, we are currently the only ones using this GPS dataset for validation. It is hard to compare these results with other drift products, since they resemble a different resolution and we don't have drift estimates at the buoy locations. However, we tried to make our validation procedure similar to the regular validation of the CMEMS ice drift to improve the possibility for future comparison.

2.b) The N-ICE campaign deployed many buoys, but very much in the vicinity of the vessel Lance. How many different buoys enter your validation database, and what is the average distance between them? Are we sampling more than few kilometres in each SAR pair?

The Norwegian Polar Institute provided us with data from 32 buoys. Based on that dataset, we automatically searched for fitting Sentinel-1 image pairs that provided more than 300 feature-tracking vectors and had a time differences of less than three days. We added a map with the resulting buoy trajectories (Figure 8) to illustrate location, spread and drift distance.

2.c) N-ICE data should offer the possibility to discuss the accuracy when inside the pack versus at the marginal ice zone. Please see if you can segment your validation database to cover this. As you point out yourself, the added value of rotation should be most visible in the marginal ice zone.

To describe the ice conditions during the collection of the validation data, we added the following to Section 2:

The ice conditions during the N-ICE2015 expedition are describe on the project website (http://www.npolar.no/en/projects/n-ice2015.html) as challenging. The observed ice pack, mainly consisting of 1.3-1.5 m thick multiyear and first-year ice, drifted faster than expected and was very dynamic. Closer to the ice edge, break up of ice floes has been observed due to rapid ice drift and the research camp had to be evacuated and re-established four times. This represents a good study field, since these challenging conditions are expected in our area and time period of

interest.

The automatic search algorithm, that allows to perform the validation on a high number of image pairs, is only comparing location and timing of buoy and satellite data and does not include any information on ice condition. To segment the validation dataset according to ice condition, we would need to describe the ice conditions for each validation vector individually. Unfortunately However, future work will cover experiments of the algorithm performance in different ice conditions.

2.d) Can you convince the reader (and the reviewer!) that the value of the maximum NCC indeed constitutes a quality measure (your Abstract)? Are matchups with lower NCC values really father away from GPS truth, than those with high NCC? Hollands et al. (2015) did not find any relation between the two. Is your threshold at 0.35 related to a significant drop in the documented accuracy against the buoy drift? (Hollands, T. , Linow, S. and Dierking, W. (2015): Reliability Measures for Sea Ice Motion Re?trieval From Synthetic Aperture Radar Images , IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8 (1), pp. 67-75 . doi: 10.1109/JSTARS.2014.2340572)

We removed the term 'quality measure' throughout the manuscript. However, we found that the probability for a high error decreases with increasing maximum cross coefficient value (Figure 11) and added the following to the validation section:

We found that the probability for a large D ***value (representative for the error) decreases with increasing maximum cross coefficient value*** MCC***. Therefore we suggest to exclude matches with a*** MCC ***value below a certain threshold*** MCC_{min}***. This option is embedded into the algorithm, but can easily be adjusted or turned off by setting*** $MCC_{min} = 0$***. Based on the findings shown in Figure 11, we recommend a cross coefficient threshold*** $MCC_{min} = 0.4$ ***for our time period and area of interest.***

A corresponding statement was added to the method section.

After changing the recommended size of the smaller template t_1 **to** 34×34 **pixels (to be consistent with the feature-tracking resolution and the aimed accuracy of the drift product), we also adjusted the cross coefficient threshold to 0.4.**

2.e) You use a maximum velocity of 0.5 m/s for your feature-based results (line 171). Is this limit high-enough in view of your validation dataset in the Fram Strait region?

To discuss the maximum velocity limit of 0.5 m/s, we added a general drift assessment to the Introduction and the following to Section 5:

The current setting of the feature-tracking algorithm applies a maximum drift filter of 0.5 m/s. We found this to be a reasonable value for our time period and area of interest. However, when considering extreme drift situations in Fram Strait and a short time interval between image

acquisitions, this threshold should be adjusted.

As mentioned above, we deployed three GPS tracker in Fram Strait and they recorded their positions with a temporal resolution of 5-30 min between 8^{th} July until 9^{th} September 2016 in an area covering 75° N to 80° N and 4° W to 14° W. Considering the displacements with 30 min interval, we found velocities above 0.5 m/s on a few occasions, when the tidal motion adds to an exceptionally fast ice drift.

The GPS data from the hovercraft expedition FRAM2014-2015 (https://sabvabaa.nersc.no), that was collected with a temporal resolution of 10 s between 31^{st} August 2014 until 6^{th} July 2015, did not reveal a single 30 min interval during which the hovercraft was moved by ice drift more than 0.45 m/s. The hovercraft expedition started at 280 km south from the North Pole towards the Siberian coast, crossed the Arctic Ocean towards Greenland and was picked up in the north-western part of Fram Strait.

We removed the validation procedure with the considered image pair over Fram Strait, even though it did not include velocities above 0.5 m/s.

Finally, it would be good if the revision of the paper could include a thorough discussion of the robustness of the combined method to the success of the feature-matching step (not in terms of computation cost, but of introduction of artefacts).

We did not find any artefacts in the test images that we considered so far. However, we would like to increase the number of image pairs significantly and produce large drift field datasets (and corresponding divergence, shear and total deformation datasets) to further evaluate the algorithm performance and investigate its robustness in terms of artefacts. To do that, we recently established a cooperation with TU Wien to embed our algorithm into their super-computing facility and learn from their experience with handling large Sentinel-1 datasets. The aim of this paper is mainly the presentation of the methodology and our next goal is the application on large datasets for further testing. To specify our next steps, we added the following to Section 5:

Our next step is to embed the algorithm into a super-computing facility to further test the performance in different regions, time periods and ice conditions. The goal is to deliver large ice drift datasets and open-source operational sea ice drift products with a spatial resolution of less than 5 km.

Thanks again for your comments. We are looking forward to your reply!

Best regards,

S. Muckenhuber and S. Sandven

Manuscript prepared for J. Name
with version 2015/09/17 7.94 Copernicus papers of the LaTeX class copernicus.cls.
Date: 13 February 2017

Open-source sea ice drift algorithm for Sentinel-1 SAR imagery using a combination of feature-tracking and pattern-matching

Stefan Muckenhuber and Stein Sandven

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

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

Abstract. An open-source sea ice drift algorithm for Sentinel-1 SAR imagery is introduced based on the combination of feature-tracking and pattern-matching. ~~A computational efficient feature-tracking algorithm~~ Feature-tracking produces an initial drift estimate and limits the search area for the consecutive pattern-matching, that provides small to medium scale drift adjustments and normalised cross ~~correlation valuesas quality measure~~coefficient values. The algorithm is designed to ~~utilise the respective advantages of the two approaches and allows drift calculation at user defined locations~~combine the two approaches in the most meaningful way in order to benefit from the respective advantages. The main advantages of the considered feature-tracking approach are the computational efficiency and the independence of the vectors in terms of position, lengths, direction and rotation. Pattern-matching on the other side allows better control over vector positioning and resolution. The pre-processing of the Sentinel-1 data has been optimised to retrieve a feature distribution that depends less on SAR backscatter peak values. ~~A recommended parameter set for the algorithm has been found using a representative image pair over Fram Strait and 350 manually derived drift vectors as validation.~~ Applying the algorithm with ~~this~~ the recommended parameter setting, sea ice drift retrieval with a vector spacing of ~~8~~4 km on Sentinel-1 images covering 400 km x 400 km, takes ~~less than 3.5~~about 4 minutes on a standard 2.7 GHz processor with 8 GB memory. The corresponding recommended patch size for the pattern-matching step, that defines the final resolution of each drift vector is 34×34 pixels (2.7×2.7 km). For validation, calculated drift results from 241 Sentinel-1 image pairs have been compared to buoy GPS data, collected in 2015 between 15^{th} January and 22^{nd} April and covering an area from ~~81~~80.5° N to 83.5° N and 12° E to 27° E~~, have been compared to calculated drift results from 261 corresponding Sentinel-1 image pairs~~. We found a logarithmic normal distribution of the error with a ~~peak at 300~~median at 341.9 m. All software requirements necessary for applying the presented sea ice drift algorithm are open-source to ensure free implementation and easy distribution.

1 Introduction

Sea ice drift has a strong impact on sea ice distribution on different temporal and spatial scales. Motion of sea ice due to wind and ocean currents causes convergence and divergence zones, resulting in formation of ridges and opening/closing of leads. On large scales, ice export from the Arctic and Antarctic into lower latitudes, where the ice eventually melts away, contributes to a strong seasonality of total sea ice coverage (IPCC, 2013). Due to a lack of ground stations in sea ice covered areas, satellite remote sensing represents the most important tool for observing sea ice conditions on medium to large scales. Despite the strong impact of sea ice drift and the opportunities given by latest satellite remote sensing techniques, there is a lack of extensive ice drift data sets providing sufficient resolution for estimating sea ice deformation on a spatial scaling of less than 5 km.

Our main regions of interest are the ice covered seas around Svalbard and East of Greenland. Characteristic for this area are a large variation of different ice types (Marginal Ice Zone, First Year Ice, Multi Year Ice etc.), a strong seasonality of ice cover and a wide range of drift velocities. Focus was put on the winter/spring period, since the area of interest experiences the highest ice cover during this time of the year.

Early work from Nansen (1902) established the rule-of-thumb that sea ice velocity resembles 2 % of the surface wind speed with a drift direction of about 45° to the right (Northern Hemisphere) of the wind. This wind driven explanation can give a rough estimate for instantaneous ice velocities. However, the respective influence of wind and ocean current strongly depends on the temporal and spatial scale. Only about 50 % of the long-term (several months) averaged ice drift in the Arctic can be explained by geostrophic winds, whereas the rest is related to mean ocean circulation. This proportion increases to more than 70 % explained by wind, when considering shorter time scales (days to weeks). The wind fails to explain large-scale ice divergence patterns and its influence decreases towards the coast (Thorndike and Colony, 1982) .

Using GPS drift data from the International Arctic Buoy Program (IABP), Rampal et al. (2009) analysed the general circulation of the Arctic sea ice velocity field and found that the fluctuations follow the same diffusive regime as turbulent flows in other geophysical fluids. The monthly mean drift using 12 h displacements was found to be in the order of 0.05 to 0.1 m/s and showed a strong seasonal cycle with minimum in April and maximum in October. The IABP dataset also revealed a positive trend in the mean Arctic sea ice speed of +17 % per decade for winter and +8.5 % for summer considering the time period 1979–2007. This is unlikely to be the consequence of increased external forcing. Instead, the thinning of the ice cover is suggested to decrease the mechanical strength which eventually causes higher speed given a constant external forcing (Rampal et al. , 2009b) .

Fram Strait represents the main gate for Arctic ice export and high drift velocities are generally found in this area with direction southward. Based on moored Doppler Current Meters mounted near 79° N 5° W, Widell et al. (2003) found an average southward velocity of 0.16 m/s for the period

1996–2000. Daily averaged values were usually in the range 0–0.5 m/s with very few occasions above 0.5 m/s.

Space-borne Synthetic Aperture Radar (SAR) are delivering systematic acquisitions of sea ice covered oceans since the early ~~90s~~ 1990's and Kwok et al. (1990) showed that sea ice displacement can be calculated from consecutive SAR scenes. SAR is an active imaging sensor operating in the microwave spectrum and produces data regardless of solar illumination and cloud cover. The geophysical processor system from Kwok et al. (1990) has been used to calculate sea ice drift fields ~~for the entire Arctic every~~ in particular over the Western Arctic (depending on SAR coverage) once per week with a spatial resolution of ~~5~~10-25 km for the time period 1997–2012. This extensive dataset makes use of SAR data from ~~Radarsat and~~ RADARSAT-1 operated by the Canadian Space Agency, and from ENVISAT (Environmental Satellite) ASAR (Advanced Synthetic Aperture Radar) operated by ESA (European Space Agency). A high-resolution sea ice drift algorithm for SAR images from ERS-1 (European Remote-sensing Satellite from ESA) based on pattern-matching was introduced by Thomas et al. (2008), allowing drift calculation up to 400 m resolution. The work on this algorithm has been continued by Hollands and Dierking (2011), who derived sea ice drift from ENVISAT ASAR data. Berg and Eriksson (2014) introduced a hybrid algorithm for sea ice drift retrieval from ENVISAT ASAR data using phase correlation and a feature based matching procedure that is activated if the phase correlation value is below a certain threshold. Komarov and Barber (2014) and Muckenhuber et al. (2016) have evaluated the sea ice drift retrieval performance of dual-polarisation SAR imagery using ~~pattern-matching~~ a combination of phase/cross-correlation and feature-tracking based on corner detetction respectively. Muckenhuber et al. (2016) has shown that feature-tracking provides on average around four times as many vectors using HV polarisation compared to HH polarisation. Making use of Sentinel-1 SAR data, an operational sea ice drift product with 10 km resolution is provided by the Danish Technical University (Pedersen et al. (2015) , http://www.seaice.dk/) as part of the Copernicus Marine Environment Monitoring Service (CMEMS, http://marine.copernicus.eu).

After the successful start of the Sentinel-1 mission in early 2014, high-resolution SAR images are delivered for the first time in history within a few hours after acquisition as open-source data to all users. This introduced a new era in SAR Earth observation with great benefits for both scientists and other stack holders. The sea ice covered oceans in the European Arctic Sector represent an important area of interest and with Sentinel-1 having a revisit time of less than one day in the Arctic (ESA, 2012), our area of interest is monitored on a daily basis. ~~Making use of Sentinel-1 data, an operational sea ice drift product with 10km resolution is provided by the Danish Technical University (DTU) as part of the Copernicus Marine Environment Monitoring Service (CMEMS, http://marine.copernicus.eu).~~ Muckenhuber et al. (2016) published an open-source feature-tracking algorithm to derive computationally efficient sea ice drift from Sentinel-1 data. This paper follows up the work from Muckenhuber et al. (2016) and aims to improve the feature-tracking approach by

combining it with pattern-matching. Unlike Berg and Eriksson (2014) , the feature-tracking step is performed initially and serves as a first guess to limit the computational effort of the pattern-matching step.

Contemporary algorithms for deriving displacement vectors between two consecutive images are based either on feature-tracking or pattern-matching.

Feature-tracking detects distinct patterns (features) in both images and tries to connect similar features in a second step without the need for knowing the locations. This can be done computationally efficient and the resulting vectors are often independent of their neighbours ~~,~~ in terms of position, lengths, direction and rotation, which is an important advantage for resolving shear zones, rotation and divergence/convergence zones. ~~However, the~~ The considered feature-tracking approach identifies features without taking the position of other features into account and matches features from one image to the other without taking the drift and rotation information from surrounding vectors into account (Muckenhuber et al., 2016) . However, due to the independent positioning of the features, very close features may share some pixels and since all vectors from the resolution pyramid are combined, the feature size varies among the matches, which implies a varying resolution. In addition, the resulting vector field is not evenly distributed in space and large gaps may occur between densely covered areas~~(Muckenhuber et al., 2016) .~~ , which can eventually lead to missing a shear or divergence/convergence zone.

Pattern-matching, on the other hand, takes a small template from the first image at the starting location of the vector and tries to find a match on a larger template from the second image. Despite a considerable computational effort, this approach is widely used, since it allows to define the vector positions~~and delivers a comparable quality estimate for each vector~~. For practical reasons, a pyramid approach is generally used to derive high-resolution ice drift. This speeds up the processing, but limits the independence of neighbouring vectors, since they depend on a lower resolution estimate (Thomas et al., 2008).

The objective of this paper is to combine the two approaches in the most meaningful way in order to benefit from the respective advantages. The main advantages of the considered feature-tracking approach are the computational efficiency and the independence of the vectors in terms of position, lengths, direction and rotation. Pattern-matching on the other side allows better control over vector positioning and resolution, which is a necessity for computing divergence, shear and total deformation.

The presented algorithm, all necessary software requirements (python incl. Nansat, openCV and SciPy) and the satellite data from Sentinel-1 are open-source. A free and user friendly implementation shall support an easy distribution of the algorithm among scientists and other stakeholders.

The paper is organised as follows: The used satellite products and buoy data are introduced in Section 2. The algorithm description including data pre-processing is given in Section 3, together with tuning and validation methods. Section 4 presents the pre-processing, parameter tuning and

validation results and provides a recommended parameter setting. The discussion including outlook can be found in Section 5.

2 Data

The Sentinel-1 mission is a joint initiative of the European Commission and the European Space Agency (ESA) and represents the Radar Observatory for the Copernicus Programme, a European system for monitoring the Earth with respect to environmental and security issues. The mission includes two identical satellites, Sentinel-1A (launched in April 2014) and Sentinel-1B (launched in April 2016), each carrying a single C-band SAR with a centre frequency of 5.405 GHz and dual-polarisation support ((HH+HV, VV+VH) also for wide swath mode. Both satellites fly in the same near-polar, sun-synchronous orbit and the revisit time is less than 1 day in the Arctic (ESA, 2012). The main acquisition mode of Sentinel-1 over sea ice covered areas is "Extra Wide Swath Mode Ground Range Detected with Medium Resolution " (EW GRDM) and the presented algorithm is built for processing this data type. The considered images have a resolution of 93m range×87m azimuth with residual planimetric distortions within 10m (**?**) . The The covered area per image is 400 km × 400 km and the data are provided with a pixel spacing of 40 m × 40 m in both HH and HV polarisation. The introduced algorithm can utilise both HH and HV channel. However, the focus of this paper is put on using HV polarisation, since this channel provides on average four times more feature tracking vectors than HH (Muckenhuber et al., 2016) , representing a better initial drift estimate for the combined algorithm.

For parameter tuning, we used the image pair 'Fram Strait', including 350 manually derived drift vectors as validation, from Muckenhuber et al. (2016) To illustrate the algorithm performance and explain the individual steps, we use an image pair acquired over Fram Strait. The acquisition times of the two consecutive images are 2015-03-28 07:44:33 (UTC) and 2015-03-29 16:34:52 (UTC), and the covered area including validation vectors are is shown in Figure **??**3. This image pair covers a wide range of different ice conditions (multiyear ice, first-year ice, marginal ice zone etc.) and the ice situation is representative for our area and time period of interest.

To evaluate suitable search limitations and validate the algorithmresults, we used , we use GPS data from drift buoys that have been set out in the ice covered waters north of Svalbard as part of the Norwegian Young Sea Ice Cruise (N-ICE2015) project of the Norwegian Polar Institute (Spreen and Itkin, 2015). The considered drift ice conditions during the N-ICE2015 expedition are describe on the project website (http://www.npolar.no/en/projects/n-ice2015.html) as challenging. The observed ice pack, mainly consisting of 1.3-1.5 m thick multiyear and first-year ice, drifted faster than expected and was very dynamic. Closer to the ice edge, break up of ice floes has been observed due to rapid

ice drift and the research camp had to be evacuated and re-established four times. This represents a good study field, since these challenging conditions are expected in our area and time period of interest. The considered GPS data have been collected in 2015 between 15^{th} January and 22^{nd} April, and cover an area ranging from ~~81~~80.5° N to 83.5° N and 12° E to 27° E. The buoys recorded their positions either hourly or every three hours. In the later case, the positions have been interpolated for each hour.

3 Method

3.1 Data pre-processing

To process Sentinel-1 images within Python (extraction of backscatter values and corresponding geolocations, reprojection, resolution reduction etc.), we use the open-source software Nansat (Korosov et al., 2016). Nansat is a scientist-friendly Python toolbox for 2-D satellite Earth observation data, and builds on the Geospatial Data Abstraction Library (http://www.gdal.org). As done in Muckenhuber et al. (2016), we change the projection of the the provided ground control points (latitude/longitude values given for certain pixel/line coordinates) to stereographic and use spline interpolation to calculate geographic coordinates. This provides a good geolocation accuracy also at high latitudes. The pixel spacing of the image is changed by averaging from 40 m to 80 m, which is closer to the sensor resolution of 93 m range $\times$ 87 m azimuth, and decreases the computational effort.

For each pixel p, the Sentinel-1 data file provides a digital number DN_p and a normalisation coefficient A_p, from which the normalised radar cross section σ^0_{raw} is derived by the following equation:

$$\sigma^0_{\text{raw}} = DN_p^2 / A_p^2 \tag{1}$$

The ~~pixel spacing of the image is changed by averaging from 40m to 80m, which is closer to the sensor resolution of 93m range×87m azimuth, and decreases the computational effort~~normalised radar cross section σ^0_{raw} reveals a logarithmic distribution and the structures in the sea ice are mainly represented in the low and medium backscatter values rather than in the highlights. Therefore, we change the linear scaling of the raw backscatter values σ^0_{raw} to a logarithmic scaling and get the backscatter values $\sigma^0 = 10 * \lg(\sigma^0_{\text{raw}})$ [dB]. A representative backscatter distribution over sea ice is shown in Figure 1. Using a logarithmic scaling provides a keypoint distribution for the feature tracking algorithm that depends less on high peak values, while the total number of vectors increases.

To apply the feature-tracking algorithm from Muckenhuber et al. (2016), the SAR backscatter values σ^0 have to be converted into intensity values i with $0 \leq i \leq 255$ for $i \in \mathbb{R}$. ~~Before the conversion , we change the linear scaling of the raw backscatter values σ^0_{raw} to a logarithmic scaling and get the~~

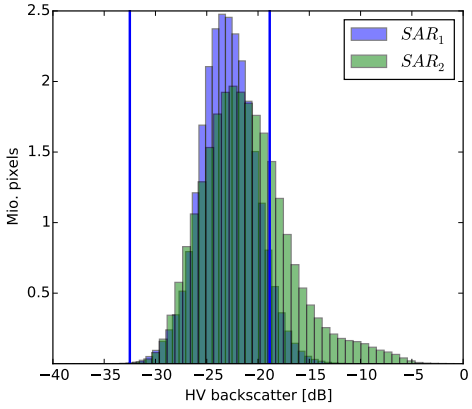

Figure 1. Histogram of HV backscatter values σ^0 from image pair Fram Strait. The lower and upper brightness boundaries $\sigma^0_{\text{min}} = -32.5\,\text{dB}$ and $\sigma^0_{\text{max}} = -18.86\,\text{dB}$ are shown with blue lines and illustrate the domain for the intensity values i.

~~backscatter values $\sigma^0 = 10 * \lg \sigma^0_{\text{raw}}$. The conversion~~ This conversion is done by using Eq. (2) and setting all values outside the domain to 0 and 255.

$$i = 255 \cdot \frac{\sigma^0 - \sigma^0_{\text{min}}}{\sigma^0_{\text{max}} - \sigma^0_{\text{min}}} \tag{2}$$

The upper brightness boundary σ^0_{max} is set ~~to the logarithm of the~~ according to the recommended value from Muckenhuber et al. (2016), i.e. ~~log(0.013) and log(0.08)~~ -18.86 dB and -10.97 dB for HV and HH respectively. The lower boundary σ^0_{min} was chosen to be ~~-3.25~~ -32.5 dB (HV) and ~~-2.5~~ -25 dB (HH), since this was found to be a reasonable range of expected backscatter values. Figure 2 shows the image pair Fram Strait after the conversion into intensity values.

3.2 Sea ice drift algorithm

The presented sea ice drift algorithm is based on a combination of feature-tracking and pattern-matching, and is designed to utilise the respective advantages of the two approaches. ~~Computational~~ Computationally efficient feature-tracking is used to derive a first estimate of the drift field. The provided vectors serve as initial search position for pattern-matching, that provides accurate drift vectors at each given location including rotation ~~and quality estimate . The~~ estimate and maximum cross coefficient value. As illustrated in the flowchart in Figure 3, the algorithm consists of ~~three~~ five main steps: I Feature tracking, II Filter, III First guess, IV Pattern matching and V Final drift product.

I Feature-tracking

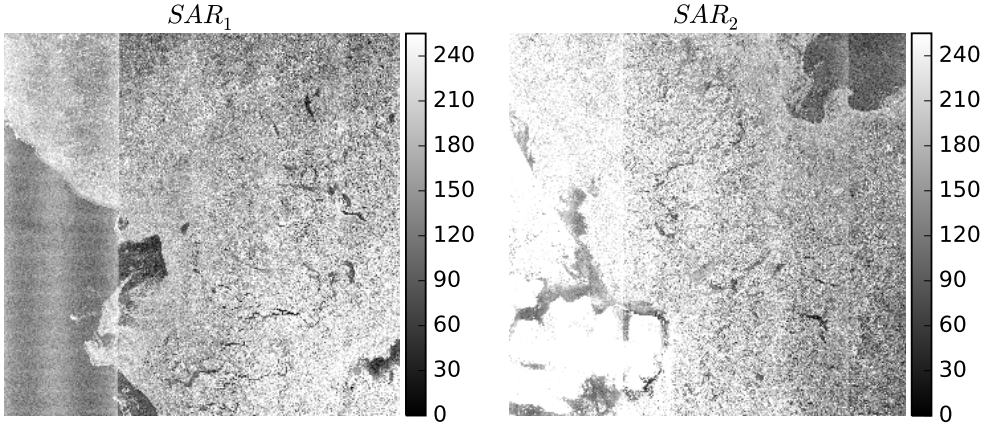

Figure 2. Image pair Fram Strait in HV polarisation after conversion (Equation 2) from backscatter values σ^0 into intensity values with range $0 \leq i \leq 255$ using lower and upper brightness boundaries $\sigma^0_{\text{min}} = -32.5$ dB and $\sigma^0_{\text{max}} = -18.86$ dB.

The feature-tracking algorithm used in this work is ~~adopted~~ an adjusted version from Muckenhuber et al. (2016), who introduced a computationally efficient sea ice drift algorithm for Sentinel-1 based on the ORB (Oriented FAST and Rotated BRIEF) algorithm from Rublee et al. (2011). ORB uses the concept of the FAST keypoint detector (Rosten and Drummond, 2006) to find corners on several resolution levels. The patch around each corner is then described using an modified version of the binary BRIEF descriptor from Calonder et al. (2010). To ensure rotation invariance, the orientation of the patch is calculated using the intensity-weighted centroid. Muckenhuber et al. (2016) applies a Brute Force matcher that compares each feature from the first image to all features in the second image. The comparison of two features is done using the Hamming distance, that represents the number of positions in which the two compared binary feature vectors differ from each other. The best match is accepted if the ratio of the ~~two shortest~~ shortest and second shortest Hamming distances is below ~~0.75.~~ a certain threshold. Given a suitable threshold, the ratio test will discard a high number of false matches, while eliminating only a few correct matches.

~~**II Pattern-matching**~~

~~The used pattern-matching approach is based on the maximisation of the normalised cross correlation. The normalised cross correlation of two equally sized windows g and h is defined as:~~

$$NCC(g,h) = \frac{\sum_{i,j}(g_{ij} - \bar{g})(h_{ij} - \bar{h})}{\sqrt{\sum_{i,j}(g_{ij} - \bar{g})^2 \sum_{i,j}(h_{ij} - \bar{h})^2}}$$

~~with g_{ij} (h_{ij}) representing the value of g (h) at the location i, j and $\bar{g}$ ($\bar{h}$) the mean value of g (h) (Hollands , 2012) . Considering a window g from a SAR image and a window h that is moved~~

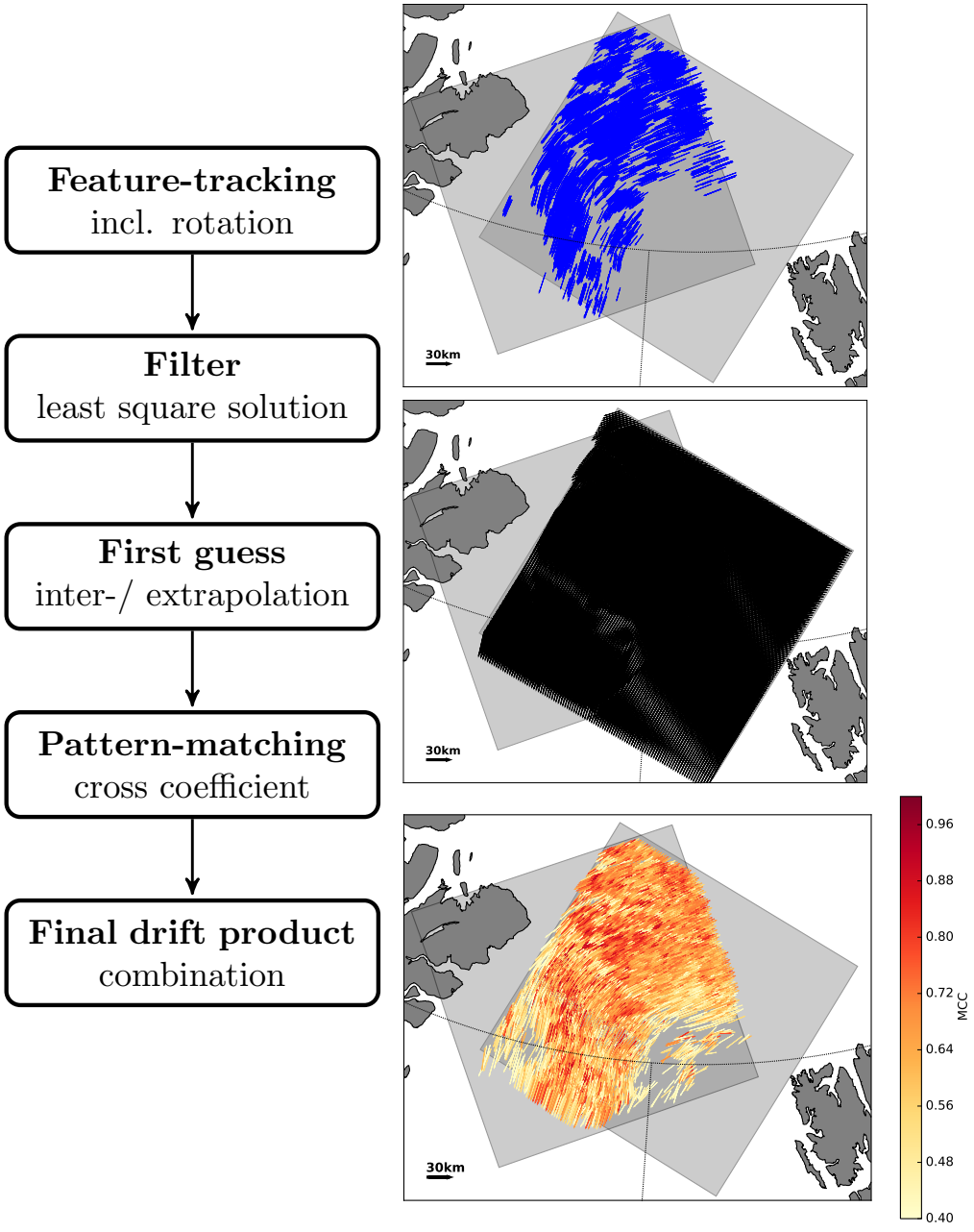

Figure 3. The flowchart on the left depicts the five main steps of the algorithm. The right column illustrates the evolution of the drift results using image pair Fram Strait in HV polarisation and a grid with 4 km spacing. Blue vectors are derived applying an adjusted version of the feature tracking algorithm from Muckenhuber et al. (2016) . Black vectors indicate the initial drift estimate (first guess) based on filtered feature-tracking vectors. The final drift product (yellow to red vectors) are derived from combining the first guess with pattern-matching adjustment and applying a minimum cross coefficient value. In this example, a total of 4725 vectors have been found with a MCC value above 0.4 in 4 min.

~~with step size 1~~Muckenhuber et al. (2016) found the most suitable parameter setting for our area and time period of interest, including a Hamming distance threshold of 0.75, a maximum drift filter of 0.5 ~~pixel over a quadratic area of a consecutive SAR image results in a matrix with NCC values . The highest value in this matrix,~~ m/s, a patch size of 34×34 pixels and a resolution pyramid with 7 steps combined with a scaling factor of 1.2. Due to the resolution pyramid, the considered feature area varies from 2.7×2.7 km to 9.8×9.8 km and the resulting drift field represents a resolution mixture between these boundaries.

We adjust the algorithm from Muckenhuber et al. (2016) by applying a logarithmic scaling for the SAR backscatter values σ_0 instead of the previous used linear scaling (Section 3.1). In addition, we extract for each vector the rotation information α, i.e. ~~the maximum cross correlation MCC, represents the location of the best match and serves as a quality estimate of the matching performance.~~ how much the feature rotates from the first to the second image.

~~III Combination~~

~~After data pre-processing as described above, the~~ Applying the adjusted feature-tracking algorithm ~~from Muckenhuber et al. (2016) is applied with a maximum drift filter of 0.5. This~~ provides a number of un-evenly distributed vectors (e.g. blue vectors in Figure 3) with start positions ~~x_1, y_1~~ x_{1f}, y_{1f} on the first image (~~SAR_1)and end positions x_2, y_2~~ SAR_1), end positions x_{2f}, y_{2f} on the subsequent image (~~SAR_2) . To filter outliers, the starting point of each vector is simulated using two functions $f_{x_1}(x_2, y_2)$ and $f_{y_1}(x_2, y_2)$~~SAR_2) and corresponding rotation values α_f. The index f represents a feature-tracking vector and ranges from 1 to F, with F being the total number of derived feature-tracking vectors.

II Filter

To reduce the impact of potentially erroneous feature-tracking vectors on the following steps, outliers are filtered according to drift and rotation estimates derived from least squares solutions using a third degree polynomial function. Considering a matrix $\mathbf{A}$, that contains all end positions x_{2f}, y_{2f} in the following form

$$\mathbf{A} = \begin{pmatrix} 1 & x_{21} & y_{21} & x_{21}^2 & y_{21}^2 & x_{21} * y_{21} & x_{21}^3 & y_{21}^3 \\ 1 & x_{22} & y_{22} & x_{22}^2 & y_{22}^2 & x_{22} * y_{22} & x_{22}^3 & y_{22}^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{2F} & y_{2F} & x_{2F}^2 & y_{2F}^2 & x_{2F} * y_{2F} & x_{2F}^3 & y_{2F}^3 \end{pmatrix} \quad (3)$$

, we derive three vectors $\boldsymbol{b}_{x_1}$, $\boldsymbol{b}_{y_1}$ and $\boldsymbol{b}_{\alpha}$, that represent the ~~least-squares solutions between x_1, y_1 and the~~ least squares solutions for $\mathbf{A}$ and $\boldsymbol{x}_1 = (x_{11}, ..., x_{1F})$, $\boldsymbol{y}_1 = (y_{11}, ..., y_{1F})$ and $\alpha = (\alpha_1, ..., \alpha_F)$ respectively. The starting position x_{1f}, y_{1f} and the rotation α_f of each vector can then be simulated

using a third degree polynomial ~~of x_2 and y_2. Vectors that have a start positionx_1 or y_1 further~~ function $f(x_{2f},y_{2f},\boldsymbol{b})$ depending on the end position x_{2f}, y_{2f} and the corresponding least squares solution $\boldsymbol{b} = (b_0,b_1,b_2,b_3,b_4,b_5,b_6,b_7)$.

$$f(x_{2f},y_{2f},\boldsymbol{b}) = b_0 + b_1x_{2f} + b_2y_{2f} + b_3x_{2f}^2 + b_4y_{2f}^2 + b_5x_{2f}y_{2f} + b_6x_{2f}^3 + b_7y_{2f}^3 \quad (4)$$

If the simulated start position, derived from $f(x_{2f},y_{2f},\boldsymbol{b})$, deviates from the feature-tracking start position x_{1f}, y_{1f} by more than 100 pixels~~(8~~, the vector is deleted. The same accounts for rotation outliers. If the simulated rotation deviates from the feature-tracking rotation α_f by more than 60°, the vector is deleted. We found a third degree polynomial function to be a good compromise between allowing for small to medium scale displacement and rotation discontinuities, and excluding very unlikely vectors, that eventually would disturb the following steps. The parameters for the filter process, i.e. 100~~) away from the simulated point are removed.~~ pixels (displacement) and 60° (rotation), have been chosen according to visual interpretation using several representative image pairs. Figure 4 illustrates the filter process by depicting the results from image pair Fram Strait.

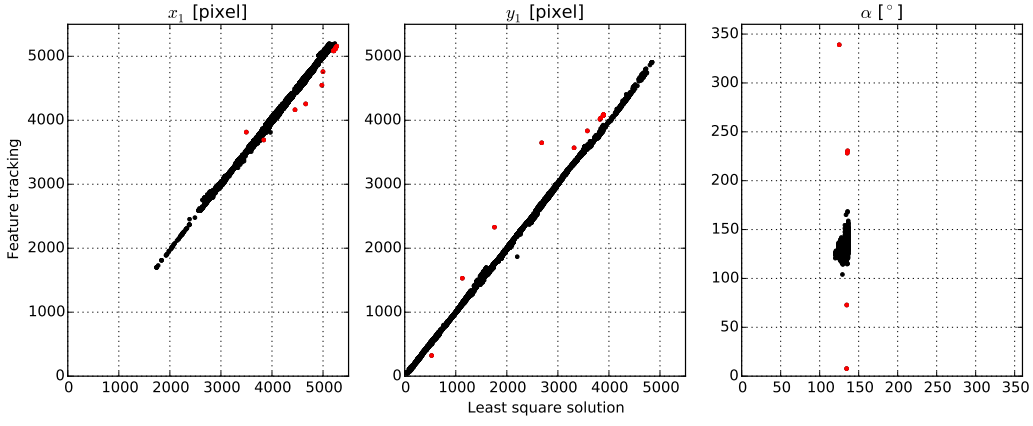

Figure 4. Filter process applied on image pair Fram Strait. The x-axis represent the simulated start position and rotation, derived from $f(x_{2f},y_{2f},\boldsymbol{b})$ and the y-axis represent the feature-tracking start position x_{1f}, y_{1f} and rotation α_f. Red points were identified as outliers and deleted.

III First guess

The remaining feature-tracking vectors are used to estimate the drift incl. rotation on the entire first image, i.e. estimated x_2~~and~~ , y_2 and α values are provided for each pixel on SAR_1 (~~left and middle panel in Figure ??). The interpolation is~~ Figure 5).

Between the feature-tracking vectors, estimated values are constructed by triangulating ~~between the start positions on SAR_1~~ the input data and performing linear barycentric interpolation on each

triangle~~to find x_2 and y_2 based on the~~ . That means, the estimated values represent the weighted mean of the three neighbouring feature-tracking ~~vectors .~~ values. The interpolated value v_p at any pixel p inside the triangle is given by Equation 5, where v_1, v_2, v_3 represent the feature-tracking values at the corners of the triangle and A_1, A_2, A_3 are the areas of the triangle constructed by p and the two opposite corners, e.g. A_1 is the area between p, and the corners with value v_2 and v_3.

$$v_p = \frac{A_1 v_1 + A_2 v_2 + A_3 v_3}{A_1 + A_2 + A_3} \tag{5}$$

To provide a ~~drift estimate~~ first guess for the surrounding area, ~~we extrapolate x_2~~ values are estimated based on the least squares solutions using a linear combination of x_1 and y_1. Considering a matrix $\mathbf{C}$, that contains all start positions x_{1f}, y_{1f} in the following form

$$\mathbf{C} = \begin{pmatrix} 1 & x_{11} & y_{11} \\ 1 & x_{12} & y_{12} \\ \vdots & \vdots & \\ 1 & x_{1F} & y_{1F} \end{pmatrix} \tag{6}$$

, we derive three vectors $\boldsymbol{d}_{x_2}$, $\boldsymbol{d}_{y_2}$ and ~~y_2 using two functions $f_{x_2}(x_1, y_1)$ and $f_{y_2}(x_1, y_1)$, that are derived from the least-squares solutions between~~ $\boldsymbol{d}_\alpha$, that represent the least squares solutions for $\mathbf{C}$ and $\boldsymbol{x}_2 = (x_{21}, ..., x_{2F})$, $\boldsymbol{y}_2 = (y_{21}, ..., y_{2F})$ and $\alpha = (\alpha_1, ..., \alpha_F)$ respectively. The estimated end position x_2, y_2 and rotation α at any location can then be simulated using the linear function $f(x_1, y_1, \boldsymbol{c})$ depending on the start position x_1, y_1 and the corresponding least squares solution $\boldsymbol{d} = (d_0, d_1, d_2)$.

$$f(x_1, y_1, \boldsymbol{d}) = d_0 + d_1 x_1 + d_2 y_1 \tag{7}$$

An example for the resulting first guess, i.e. estimated values for x_2, y_2 and ~~a linear combination of x_1 and y_1.~~ α on SAR$_1$, is shown in Figure 5 and corresponding vectors are shown in black in Figure 3. Note that rotation α includes both the relative image rotation from SAR_1 to SAR_2 and the actual rotation of the feature itself. The introduced algorithm provides also the relative image rotation by projecting the left corners of SAR_2 onto SAR_1 and calculating the angle between the left edges of SAR_1 and SAR_2. The actual rotation of the features can easily be obtained by subtracting the relative image rotation from α.

~~This initial drift estimate is used to perform efficient~~ **IV Pattern-matching**

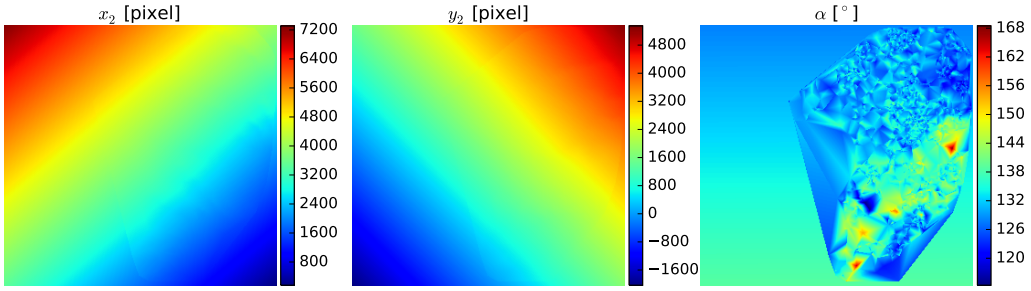

Figure 5. ~~Estimated~~ Example of estimated drift and rotation (first guess) based on filtered feature-tracking ~~, and distance d to the nearest feature-tracking vector~~ vectors using ~~the~~ image pair ~~'~~Fram Strait~~' from Muckenhuber et al. (2016)~~. The ~~left and middle panel~~ three panels show the ~~two~~ components x_2 ~~and~~ , y_2 of the estimated end positions ~~on~~ and the ~~second image (SAR_2)~~ estimated rotation α for each pixel on the first image (SAR_1).~~The right panel shows the distribution of d on SAR_1 with a lower and upper threshold d_{min}, d_{max}.~~

The estimated drift field derived from feature-tracking provides values for x_2, y_2 and α at any location on SAR_1. The uncertainty of this estimate however, increases with distance d to the closest feature-tracking vector. Therefore, small to medium scale adjustments of the estimates are necessary, depending on the distance d. We apply pattern-matching ~~based on normalised cross correlation on a pre-defined grid or~~ at chosen points of interest ~~. Figure 6 shows a pattern-matching example from image pair 'Fram Strait' used in Muckenhuber et al. (2016) .~~ to provide more accurate drift vectors and adjust the rotation estimate at these specific locations.

~~Pattern-matching using initial drift estimate from feature-tracking: Small template t_1 (left) around point of interest on SAR_1 is rotated from $-\beta$ to $+\beta$ and matched with large template t_2 (middle) from SAR_2, that has its centre at estimated end position x_2, y_2. The right contour plot shows the normalised cross correlation matrix of the rotation β^* that provided the highest maximum cross correlation $MCC(\beta^*)$. The estimated end position x_2, y_2 of this example has to be adjusted by -21pixels, +32pixels to fit with the location of $MCC(\beta^* = 2°) = 0.71$. NB: X and Y -axis represent pixel coordinates.~~

~~A~~ The used pattern-matching approach is based on the maximisation of the normalised cross coefficient. Considering a small template t_1 ~~with a given size is taken~~ around the point of interest from ~~SAR_1. A~~ SAR_1 with size $t_{1s} \times t_{1s}$ and a larger template t_2 ~~with centre at~~ around the location x_2, y_2 (defined by the corresponding ~~drift estimate from feature-tracking is taken from SAR_2. The size of t_2 is defined by the distance d to the nearest feature-tracking vector with a lower and upper threshold (d_{min}, d_{max}) :~~

$$\text{side}(t_2) = \text{side}(t_1) + 2 * d$$

first guess) from SAR_2 with size $t_{2s} \times t_{2s}$, the normalised cross coefficient matrix **NCC** is defined as (Hollands , 2012) :

$$\mathbf{NCC}(x,y) = \frac{\sum_{x',y'}(t'_1(x',y')t'_2(x+x',y+y'))}{\sqrt{\sum_{x',y'} t'_1(x',y')^2 \sum_{x',y'} t'_2(x+x',y+y'))^2}} \tag{8}$$

$$t'_1(x',y') = t_1(x',y') - \frac{1}{t_{1s}^2} \sum_{x'',y''} t_1(x'',y'') \tag{9}$$

$$t'_2(x+x',y+y') = t_2(x+x',y+y') - \frac{1}{t_{1s}^2} \sum_{x'',y''} t_2(x+x'',y+y'') \tag{10}$$

with ~~$d_{min} \leq d \leq d_{max}$ for $d \in \mathbb{N}$ (example of d distribution in right panel of Figure ??). The two templates~~ $t_1(x',y')$ and $t_2(x',y')$ representing the value of t_1 and t_2 ~~are matched using maximisation of normalised cross correlation. Template~~ at location x',y'. The summations are done over the size of the smaller template, i.e. x', y', x'' and y'' go from 1 to t_{1s}. Template t_1 is moved with step size 1 pixel over template t_2 both in horizontal (x) and vertical (y) direction and the cross coefficient values for each step are stored in the matrix **NCC** with size $(1+t_{s2}-t_{s1}) \times (1+t_{s2}-t_{s1})$. The highest value in the matrix **NCC**, i.e. the the maximum normalised cross coefficient value MCC, represents the location of the best match and the corresponding location adjustment is given by dx and dy.

$$(\frac{1+t_{s2}-t_{s1}}{2} + dx, \frac{1+t_{s2}-t_{s1}}{2} + dy) = argmax(\mathbf{NCC}(x,y) \tag{11}$$

To restrict the search area t_{2s} to a circle, we set all values of **NCC** that are further than $t_{2s}/2$ away from the centre position to zero.

To account for rotation adjustment, the matrix **NCC** is calculated several times: template t_1 is rotated ~~starting with the initial rotation between the two Sentinel-1 images and going from $-\beta$ to $+\beta$ with step~~ around the initially estimated rotation α from $\alpha - \beta$ to $\alpha + \beta$ with step size $\Delta\beta$. The ~~result~~ angle β is the maximum additional rotation and represents therefore the rotation restriction. The **NCC** matrix with the highest cross ~~correlation value~~ coefficient value MCC is returned.

~~In the last step, the small to medium scale displacement adjustments derived from~~ To illustrate the pattern-matching ~~are added to the estimated drift from feature-tracking. The maximum cross correlation values serve as individual quality measure for each drift vector and vectors that have a MCC value below the threshold MCC_{min} are removed.~~ process, an example, taken from image pair Fram Strait, is shown in Figure 6.

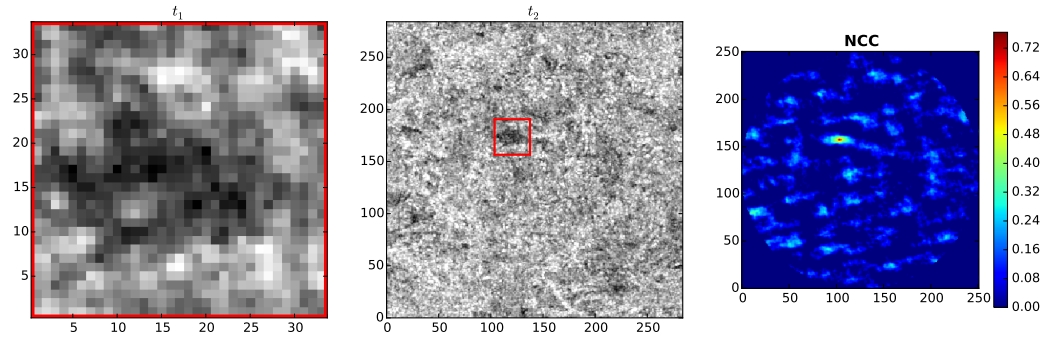

Figure 6. Pattern-matching using initial drift estimate from feature-tracking: The small template t_1 (left) around the point of interest on SAR_1 is rotated from $\alpha - \beta$ to $\alpha + \beta$ and matched with the large template t_2 (middle) from SAR_2, that has its centre at the estimated end position x_2, y_2. The right contour plot shows the normalised cross coefficient matrix **NCC** of the rotation β^* that provided the highest maximum cross coefficient MCC. The estimated end position x_2, y_2 of this example has to be adjusted by $dx = -21$ pixels, $dy = 32$ pixels to fit with the location of $MCC = 0.71$. Rotation adjustment β^* was found got be 3°. NB: X and Y-axis represent pixel coordinates.

3.3 ~~Parameter tuning~~

The described process demands the specification of four parameters: t_{1s}, t_{2s}, β and $\Delta\beta$.

The size of the small template $t_{1s} \times t_{1s}$ defines the considered area that is tracked from one image to the next and hence, affects the resolution of the resulting drift product. In order to be consistent with the resolution of the feature-tracking step and achieve our goal of a sea ice drift product with a spatial scaling of less than 5 km, we use the size of the feature-tracking patch of the pyramid level with the highest resolution to define the size of t_1. That means, we use $t_{s1} = 34$ pixels (2.7 km).

The size of ~~template t_1 and t_2 are crucial for a reliable drift result and for limiting~~ the larger template $t_{2s} \times t_{2s}$ restricts the search area on SAR_2, i.e. how much the first guess can be adjusted geographically, and the angle β restricts the rotation adjustment of the first guess α. The three parameter t_{2s}, β and $\Delta\beta$ have a strong influence on the computational efficiency of the drift algorithm. Meaning that an increase of t_{2s}, β and a decrease of $\Delta\beta$ increase the computational effort ~~. As shown in Equation ??,~~ of the pattern-matching step. Based on visual interpretation of several representative image pairs, we found $\Delta\beta = 3°$ to be a good compromise between matching performance and computational efficiency.

Since the uncertainty of the ~~maximum size of t_2 is limited by the upper threshold d_{max}~~ first guess increases with distance d (Figure 7) to the closest feature-tracking vector, the search restrictions t_{2s} and β should increase with d. To find ~~the most useful values for d_{max} and the size of t_1, we varied the two parameters within the domains $20 \leq \text{side}(t_1) \leq 140$ with $\Delta\text{side}(t_1) = 2$, and $50 \leq d_{max} \leq 200$ with $\Delta d_{max} = 10$ (values given in pixels). For each combination, we calculated~~

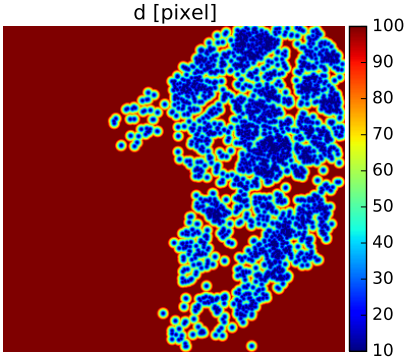

Figure 7. Example to illustrate the distribution of distance d to the closest feature-tracking vector using image pair Fram Strait. Values outside the range $d_{min} \leq d \leq d_{max}$ are set to $d_{min} = 10$ and $d_{max} = 100$. The points with value d_{min} represent the start positions x_1, y_1 of the feature-tracking vectors on SAR_1.

~~the drift on image pair 'Fram Strait' at the starting locations of the 350 manually derived validation vectors and compared the results using root mean square distance $RMSD$:~~ useful restrictions for t_{2s} and β, we calculated drift vectors using very high values for t_{2s} and β, i.e. being computationally more demanding than we anticipate, and compared the results with the GPS drift buoy dataset from the N-ICE2015 expedition. Based on the results (Section 4) we found the following functions to represent useful restrictions for our area and time period of interest.

$$t_{2s}(d) = t_{1s} + 2d \qquad d_{min} \leq d \leq d_{max} \qquad d \in \mathbb{N} \tag{12}$$

~~The index i represents a vector pair consisting of a calculated vector and a validation vector at the same location.The eastward and northward drift components of the calculated vector are u_i and v_i. The validation vector has the corresponding drift components U_i and V_i. N is the total number of vector pairs.~~

$$\beta(d) = \begin{cases} 9 & \text{if } d < d_{max} \\ 12 & \text{if } d \geq d_{max} \end{cases} \tag{13}$$

The values for d_{min}, d_{max}, β and $\Delta\beta$ can easily be varied in the algorithm to adjust for e.g. different areas, drift conditions or a different compromise between matching performance and computational efficiency.

~~During parameter tuning, the minimum value of the normalised cross correlation~~ **V Final drift product**

In the last step, the small to medium scale displacement adjustments from pattern-matching are added to the estimated first guess derived from feature-tracking. Using buoy comparison, we found that the probability for large displacement errors decreases with increasing MCC value (Section 4). Therefore, vectors that have a MCC value below the threshold MCC_{min} ~~was set to zero. We applied rotation on t_1 ranging from $-10°$ $(-\beta)$ to $+10°$ $(+\beta)$ with step size $\Delta\beta = 1°$. The lower threshold d_{min} was set to 20pixels to allow for small scale drift adjustments close to the locations of feature-tracking vectors~~are removed. We found $MCC_{min} = 0.4$ to be a good filter value, but this value can easily be adjusted in the algorithm depending on the sought compromise between amount of vectors and error probability. The algorithm returns the final drift vectors in longitude, latitude, the corresponding first guess rotation α and the rotation adjustment β in degrees and the maximum cross coefficient value MCC. An example for the final product is depicted with yellow to red coloured vectors in Figure 3. The colour scale refers to the MCC value, indicating the probability for an erroneous vector.

3.3 Comparison with buoy data

Sentinel-1 image pairs have been selected automatically according to position and timing of the ~~buoy data.~~ GPS buoy data from the N-ICE2015 expedition. Each pair yielded more than 300 drift vectors using the feature-tracking algorithm from Muckenhuber et al. (2016) and had a time difference between the two acquisitions of less than three days. Drift vectors have been calculated with the presented algorithm starting at the buoy GPS position with the least time difference to the acquisition of the first satellite image. The distance D between the calculated end position on the second image and the buoy GPS position with the least time difference to the second satellite acquisition has been calculated using the following equation:

$$D = \sqrt{(u-U)^2 + (v-V)^2} \tag{14}$$

where u and v represent eastward and northward drift components of the displacement vector derived by the algorithm, and U and V the corresponding drift components of the buoy.

4 Results

4.1 ~~Logarithmic scaling of σ^0~~

~~The start and end positions of the feature-tracking vectorsare tied to keypoints that are found during an initial processing step in the drift algorithm from Muckenhuber et al. (2016) .Looking at the distribution of the intensity values among the matched keypoints~~

4.1 Search restrictions evaluation

To find suitable values for restricting the size of the search window t_{2s} and the rotation range defined by β, we calculated drift vectors, that can be compared to the considered GPS buoy dataset, using restrictions that are computationally more demanding than we anticipate for the recommended setting, i.e. $t_{2s} = 434$ pixels and $\beta = 18°$. These values corresponds to a possible pattern-matching adjustment of up to 200 pixels (16 km) and 18° in any direction independent of the distance d to the closest feature-tracking vector.

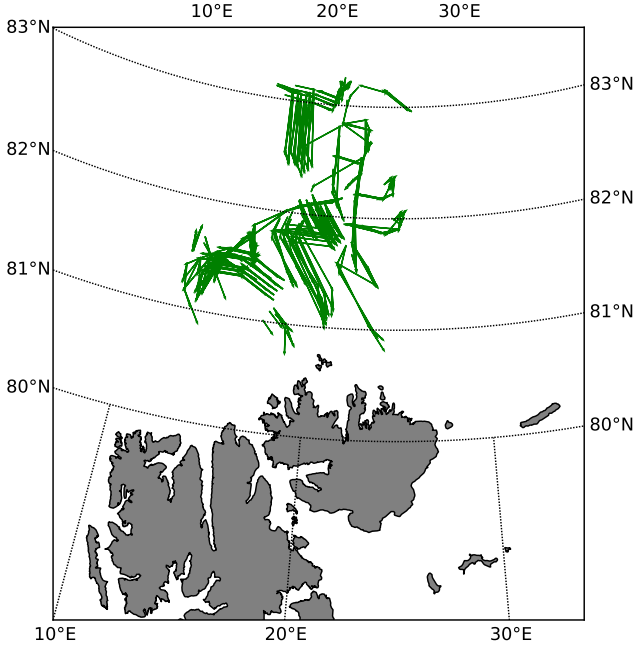

Figure 8. Considered buoy trajectories from the N-ICE2015 expedition that were used for comparison with algorithm results.

Based on an automatic search, we found ~~a strong peak at maximum intensity 255, meaning that most matched keypoints are recognised at very high backscatter values (NB: the number of vectors is equal to the number of matched keypoints divided by two). Before the conversion of the backscatter values σ^0 into the intensity values i (Equation 2), the scaling of σ^0 can be changed from linear ($\sigma^0 = \sigma^0_{\text{raw}}$)to logarithmic ($\sigma^0 = 10 * \lg \sigma^0_{\text{raw}}$). Table **??** and Figure **??** show the intensity value distribution of matched keypoints from image pair 'Fram Strait'. Using a logarithmic instead of a linear scaling provided a keypoint distribution that depends less on high peak values (the number of keypoints with intensity value 255 decreased from 5378 to 2513), while the total number of vectors increased from 7042 to 10034.~~ 240 matching Sentinel-1 image pairs (consisting of 110 images), that allowed for comparison with 689 buoy vectors (Figure 8). The distance D (Equation 14) between the buoy location at the time of the second image SAR_2 and the corresponding algorithm result,

represents the error estimate for one vector pair. To identify algorithm results that are more likely erroneous, vector pairs with a value D above 1000 m are marked with red dots in Figure 9 and Figure 10. Vector pairs with $D < 1000$ m are plotted with black dots.

~~Linear and logarithmic scaling of the backscatter values σ^0 in HV and HH polarisation: range, i.e. lower and upper brightness boundaries as used in Equation 2, and number of matched keypoints using image pair 'Fram Strait' (matched keypoints with maximum pixel value 255 in brackets). Linear scaling ($\sigma^0 = \sigma^0_{\text{raw}}$) Logarithmic scaling ($\sigma^0 = 10 * \lg \sigma^0_{\text{raw}}$)HV range σ^0_{min}~~Figure 9 and Figure 10 show the resulting pattern-matching adjustment of location (dx, ~~σ^0_{max}0, 0.013-3.25; $\log(0.013)$HH range σ^0_{min}, σ^0_{max}0, 0.08-2.5; $\log(0.08)$HV matched keypoints11244 (4653) 15614 (2196) HH matched keypoints2840 (725) 4454 (317)~~ dy) and rotation ($d\beta$) using the computationally demanding restrictions. The values are plotted against distance d to the next feature tracking vector in order to identify the dependence of the parameters on d. The blue lines in Figure 9 and Figure 10 indicate the recommended restrictions. This represents a compromise between computational efficiency and allowing the algorithm to adjust the first guess as much as needed for our time period and area of interest. The corresponding functions for $t_{2s}(d)$ and $\beta(d)$ are given in Equation 12 and Equation 13 and the recommended boundary values for distance d are $d_{min} = 10$ and $d_{max} = 100$.

~~Number of matched keypoints for each intensity value in image pair 'Fram Strait' using linear (HV: black, HH: gray) and logarithmic scaling (HV: darkgreen, HH: lightgreen) for the backscatter values σ^0. Total numbers, and matches at intensity maximum 255 are shown in Table ??.~~

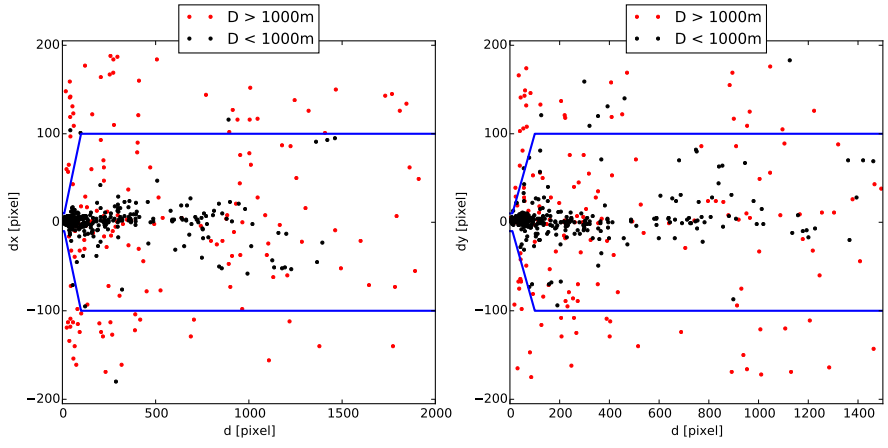

Figure 9. Pattern-matching location adjustment dx and dy in x and y direction versus distance d to closest feature tracking vector. D represents the difference between buoy GPS position and algorithm result. The blue lines indicate the recommended setting for t_{2s} (Equation 12) with $d_{min} = 10$ and $d_{max} = 100$.

4.2 ~~Parameter tuning~~

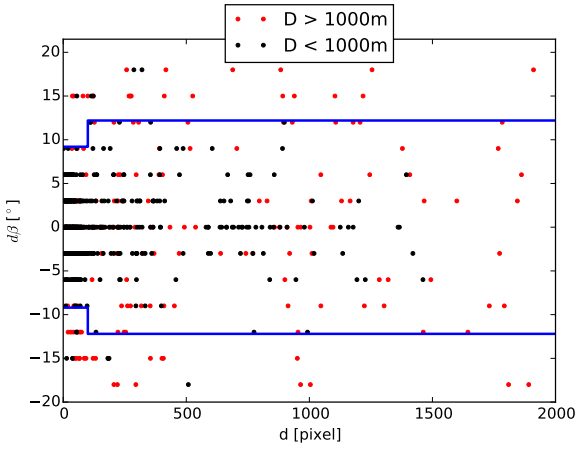

Figure 10. Pattern-matching rotation adjustment $d\beta$ versus distance d to closest feature tracking vector. D represents the difference between buoy GPS position and algorithm result. The blue lines indicate the recommended setting for β (Equation 13) with $d_{min} = 10$ and $d_{max} = 100$.

~~Figure ?? shows the $RMSD$ (Equation ??) calculated for image pair 'Fram Strait' using HH (left panel) and HV polarisation (right panel). Based on this~~

4.2 Validation

Using the recommended search restrictions from above, the algorithm has been validated against the N-ICE2015 GPS buoy data set (Figure 8). The automatic search provided 241 image pairs (consisting of 110 images) and 714 vectors for comparison for the considered time period (15^{th} January to 22^{nd} April) and area (80.5° N to 83.5° N and 12° E to 27° E). NB: this is a higher number of vectors than found for the evaluation of the search restrictions, since the used search windows t_2 are smaller and vectors closer to the SAR edge may be included.

The results of the conducted validation are shown in Figure 11. We found that the probability for a large D value (representative for the error) decreases with increasing maximum cross coefficient value MCC. Therefore we suggest to exclude matches with a MCC value below a certain threshold MCC_{min}. This option is embedded into the algorithm, but can easily be adjusted or turned off by setting $MCC_{min} = 0$. Based on the findings shown in Figure 11, we recommend a cross coefficient threshold $MCC_{min} = 0.4$ for our time period and area of interest. Using the suggested threshold reduces the number of vector pairs from 714 to 565.

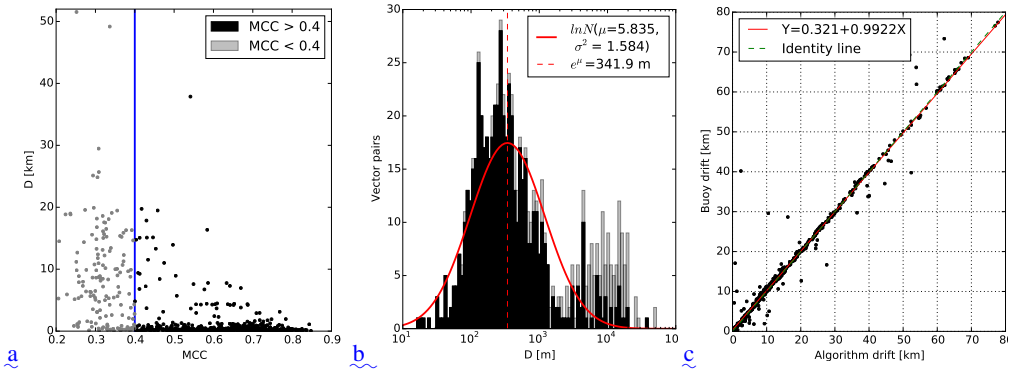

Figure 11. Calculated ice drift using recommended search restrictions compared to buoy GPS data. Light grey represents vectors with maximum cross coefficient values $MCC < 0.4$ and results after using the suggested threshold $MCC_{min} = 0.4$ are shown in black. (a) MCC values against distance D (Equation 14) between algorithm and buoy end position. The blue line indicates the recommended setting for $MCC_{min} = 0.4$. (b) Logarithmic histogram of distance D with 100 bins between 10 m and 10^5 m including a logarithmic normal distribution (solid red line) that was fitted to the results with $MCC > 0.4$. (c) Comparison of drift distance derived from algorithm against buoy displacement.

The conducted validation also reveals a logarithmic normal distribution of the distance D (Equation 14) that can be expressed by the following probability density function (solid red line in Figure 11):

$$lnN(D;\mu,\sigma) = \frac{1}{\sigma D\sqrt{2\pi}} e^{-\frac{(\ln D-\mu)^2}{2\sigma^2}} \tag{15}$$

with μ and σ being the mean and standard deviation of the variable's natural logarithm. We found the mean and variance of the distribution lnN to be $\mu = 5.835$ and $\sigma^2 = 1.584$. The median of the logarithmic normal distribution is $e^{\mu} = 341.9\,m$ (dashed red line in Figure 11).

4.3 Recommended parameter setting

Based on the restriction evaluation, our experience with the algorithm behaviour, and considering a good compromise between computational efficiency and high quality of the resulting vector field, we recommend the parameter setting shown in Table 1. The ~~following testing and validation process is conducted using this parameter setting.~~ corresponding recommended values for $t_{2s}(d)$ and $\beta(d)$ are given in Equation 12 and Equation 13.

~~$RMSD$ (Equation **??**) calculated for image pair 'Fram Strait' using different values for d_{max} (X-axis) and for the side length of template t_1 (Y-axis). Left panel shows HH polarisation and right panel HV polarisation. $RMSD$ values are given in meters, X and Y-axis represent pixel values. Black crosses mark recommended parameter setting: $d_{max} = 125$pixels and side$(t_1) = 70$pixels.~~

Table 1. Recommended parameter setting for sea ice drift retrieval from Sentinel-1 using the presented algorithm.

Parameter	Meaning	Recommended setting
$[\sigma^0_{\min}, \sigma^0_{\max}]$ (HH)	Brightness boundaries for HH channel	[~~-2.5, log(0.08)~~ -25 dB, -10.97 dB]
$[\sigma^0_{\min}, \sigma^0_{\max}]$ (HV)	Brightness boundaries for HV channel	[~~-3.25, log(0.013)~~ -32.5 dB, -18.86 dB]
~~side(t_1)~~ t_{1s}	Size of template t_1	~~70~~34 pixels (2.7 km)
$[d_{min}, d_{max}]$	~~Influence domain for size of t_2~~ Boundaries for distance d	[~~20~~10 pixels, ~~125~~100 pixels]
MCC_{min}	Threshold for cross ~~correlation~~ coefficient	~~0.35~~0.4
~~$-\beta, +\beta,$~~ $\Delta\beta$	Rotation angle ~~and~~ increment	~~-10~~3°~~, +10°, 2°~~

4.4 Computational efficiency

The processing time depends on the parameter setting and the chosen vector distribution. Using the recommended parameter setting from Table 1, allows high-resolution sea ice drift retrieval from a Sentinel-1 image pair within a few minutes. Figure ~~??~~ 3 depicts calculated ice drift vectors for the image pair ~~'Fram Strait '~~ Fram Strait on a grid with ~~8~~4 km (~~100~~50 pixels) spacing. The corresponding processing times are shown in Table 2. The calculations have been done using a MacBook Pro from early 2013 with a 2.7 GHz Intel Core i7 processor and 8 GB 1600 MHz DDR3 memory. The total processing time for ~~1145~~ 4725 vectors with a normalised cross ~~correlation value above 0.35, is less than 3.5~~coefficient value above 0.4, is about 4 minutes. ~~NB: The vectors near Svalbard are located in the marginal ice zone. This is a very challenging area for drift algorithms based on consecutive images and the results have to be treated with caution~~This can be considered a representative value for an image pair with large overlap, good coverage with feature-tracking vectors and 4 km grid spacing.

The initial process in Table 2 'Create Nansat objects from Sentinel-1 image pair and read matrixes' takes the same amount of computational effort for all image pairs consisting of Sentinel-1 images with 400x400 km coverage.

The process 'I Feature-tracking' depends on the setting of the feature-tracking algorithm and varies strongly with the chosen number of features. Using the recommended setting from Muckenhuber et al. (2016) , that includes the number of features to be 100000, the presented computational effort can be considered representative for all image pairs, independent of chosen points of interest and overlap of the SAR scenes.

The last process 'II Pattern-matching and III Combination' however, depends on the considered image pair and the chosen drift resolution. The computational effort is proportional to the number of chosen points of interest. Given a evenly distributed grid of points of interest, the computational effort increases with overlapping area of the SAR scenes, since pattern-matching adjustments are only calculated in the overlapping area. The effort potentially decreases with a higher number of

~~Processing time for sea ice drift retrieval from image pair 'Fram Strait' on a grid with 8km (100pixels) spacing using HV polarisation (Figure ??).~~

Table 2. Processing time for sea ice drift retrieval from image pair Fram Strait on a grid with 4 km (50 pixels) spacing using HV polarisation (Figure 3). Representative for an image with large overlap and good coverage with feature-tracking vectors.

Process	Time [s]
~~Create Nansat objects from~~ Create Nansat objects from Sentinel-1 ~~image pair~~image pair	~~21~~
~~Read matrixes from Nansat objects~~and read matrixes	~~49~~ 70
I Feature-tracking	66
II ~~Pattern-matching and III Combination~~Pattern-matching and III Combination	~~65~~ 107
$\sum$ Sea ice drift retrieval	~~201~~ 243

well distributed feature-tracking vectors, since the size of the search windows t_2 (and slightly the range of the angle β) increases with distance d to the closest feature-tracking vector.

~~Sea ice drift from image pair 'Fram Strait' on a grid with 8km spacing using HV polarisation. Black vectors indicate the initial drift estimate from feature-tracking. Coloured vectors are derived from combining feature-tracking and pattern-matching with a minimum cross correlation value $MCC_{min} = 0.35$. The colour indicates the maximum cross correlation MCC. A total of 1145 vectors have been found with a MCC value above 0.35.~~

4.5 ~~Validation~~

~~The manually derived vectors from the image pair 'Fram Strait' have been compared with calculated drift vectors at the same locations (Figure ??) using $RMSD$ from Equation ?? and the recommended parameters from Table 1. 335 vectors with a normalised cross correlation value above 0.35 could be used for the comparison and the resulting $RMSD$ value is 540m. NB: nine vectors close to the image border could not be used for comparison. The templates t_1 and t_2 are larger then the patch size used for simple feature-tracking (34pixels) and hence, the restricted area at the image border is slightly increased.~~

~~Sea ice drift derived from image pair 'Fram Strait' using HV polarisation: manually drawn validation vectors (green), initial drift estimate from feature-tracking (black) and vectors from combined feature-tracking and pattern-matching (colour according to maximum cross correlation MCC). The right panel shows the entire scene and the left panel depicts the algorithm procedure on an enlarged area.~~

~~To compare the drift results from the algorithm with GPS positions from the N-ICE2015 buoy data set, 261 Sentinel-1 image pairs have been selected automatically for the considered time period~~

(15thJanuary to 22ndApril) and area (81°N to 83.5°N and 12°E to 27°E). Each pair yielded more than 300 drift vectors using the feature-tracking algorithm from Muckenhuber et al. (2016) and had a time difference between the two acquisitions of less than three days. The satellite and buoy data sets provide 633 possible displacement pairs for comparison. Using the suggested threshold for cross correlation $MCC_{min} = 0.35$ reduces the number of vector pairs to 540. The results of the comparison are shown in Figure **??**. We found a logarithmic distribution of the distance D (Equation 14) with a peak at 300m (3.75pixels).

Buoy GPS data compared to calculated ice drift. Logarithmic histogram of distance D (Equation 14) with 100bins between 10m and 10^5m. Light grey shows the unfiltered results ($MCC_{min} = 0$) and dark grey shows the results after using the suggested threshold for cross correlation $MCC_{min} = 0.35$. The peak of the distribution is marked with a red line at 300m.

5 Discussion and outlook

Comparison of presented algorithm (combined feature-tracking + pattern-matching), simple feature-tracking as done in Muckenhuber et al. (2016) , and CMEMS data using image pair 'Fram Strait' and 350 manually derived drift vectors as validation. $RMSD$ is the root mean square distance from Equation **??**. The vector pairs refers to the number of used vector pairs for comparison, i.e. vector pairs with maximum 5km distance. The average distance is measured between the start positions of the validation vectors and the corresponding nearest neighbour vectors from the algorithm. Algorithm $RMSD$ mvector pairs Average distance mFeature-tracking + pattern-matching540 335 0 Feature-tracking563 314 1702±1325 CMEMS (pattern-matching) 1690 201 3440±1105

Muckenhuber et al. (2016) compared their drift results (based on simple feature-tracking) and drift vectors from the Copernicus Marine Environment Monitoring Service (CMEMS) with the same manually drawn vectors as we use in Section 4. The CMEMS product is provided by the Technical University of Denmark (DTU), has a resolution of 10km and is based on pattern-matching techniques (**?** , http://www.seaice.dk/). Since the start locations of the drift vectors from these two algorithms do not coincide with the validation vectors, Muckenhuber et al. (2016) used the nearest neighbours for comparison and a maximum distance of 5km. Table **??** shows the validation results from simple feature-tracking as done in Muckenhuber et al. (2016) , pattern-matching as done by DTU for CMEMS, and the presented combined feature-tracking + pattern-matching algorithm. The combined algorithm provides the highest accuracy (represented by $RMSD = 540$m) and the highest number of vectors pairs that can be used for comparison (335). Unlike simple feautre-tracking, the combined algorithm allows to choose the positions of the drift vectors, which makes it possible to place them at the same locations as the validation vectors. This is represented by an average distance of 0m. As discussed in Muckenhuber et al. (2016) , the manually drawn vectors cannot be

~~considered as perfect validation, since there might be an error introduced during the manual ice drift identification. Hence, we expect the error originating from the combined algorithm to be less than 540m.~~

To ~~further~~ estimate the accuracy of the introduced algorithm, we compared ~~the~~ drift results from ~~261~~ 240 Sentinel-1 image pairs with corresponding GPS positions from the ~~N-ICE~~ N-ICE2015 buoy data set. We found a logarithmic error distribution with a ~~peak at 300~~median at 341.9 m (Figure ~~??~~11). The derived error values represent a combination of the following error sources:

- Timing: Buoy GPS data were collected every 1-3 hours and the timing does not necessarily match with the satellite acquisition time.
- Resolution: The algorithm returns the drift of a pattern (recommended size = ~~70~~34 pixels, see Table 1), whereas the buoy measures the drift at a single location.
- Conditions: The ice conditions around the buoy is not known well enough to exclude the possibility that the buoy is floating in a lead. In this case, the buoy trajectory could represent a drift along the lead rather then the drift of the surrounding sea ice.
- actual error of the algorithm.

~~Hence, the actual error of the presented algorithm is expected to be even lower than 300m. This means, that the algorithm accuracy is in the scale of the satellite image resolution.~~

A main advantage of the combined algorithm compared to simple feature-tracking, is the user defined positioning of the drift vectors. The current algorithm setup allows the user to choose whether the drift vectors should be positioned at certain points of interest or on a regular grid with adjustable spacing. Constricting the pattern-matching process to the area of interest minimises the computational effort according to the individual needs.

The recommended parameters shown in Table 1 are not meant as a fixed setting, but should rather give a suggestion and guideline to estimate the expected results and the corresponding computational effort. The parameters can easily be varied in the algorithm setup and should be chosen according to availability of time, computational power, number of image pairs, needed accuracy, area of interest and expected ice conditions (e.g. strong rotation).

The presented combination of feature-tracking and pattern-matching can be applied to any other application that aims to derive displacement vectors computationally efficient from two consecutive images. The only restriction is that images need to depict edges, that can be recognised as keypoints for the feature-tracking algorithm, and the conversion into intensity values i (Equation 2) needs to be adjusted according to the image type.

The remote sensing ~~of sea ice~~ group at NERSC is currently developing a new pre-processing step to remove thermal noise on HV images over ocean and sea ice. First tests have shown a significant

improvement of the sea ice drift results using this pre-processing step before applying the presented algorithm. This is ongoing work and will be included into a future version of the algorithm.

~~Having a computational~~ The European Space Agency is also in the process of improving their thermal noise removal for Sentinel-1 imagery. Noise removal in range direction is driven by a function that takes measured noise power into account. Until now, noise measurements are done at the start of each data acquisition, i.e. every 10-20 minutes, and a linear interpolation is performed to provide noise values every 3 seconds. The distribution of noise measurements showed a bimodal shape and it was recently discovered that lower values are related to noise over ocean while higher values are related to noise over land. This means, that Sentinel-1 is able to sense the difference of the earth surface brightness temperature similar to a passive radiometer. When the data acquisition includes a transition from ocean to land or vice versa, the linear interpolation fails to track the noise variation. The successors of Sentinel-1A/B are planned to include more frequent noise measurements. Until then, ESA wants to use the 8-10 echoes after the burst that are recorded while the transmitted pulse is still travelling and the instrument is measuring the noise. This will provide noise measurements every 0.9 seconds and allows to track the noise variations in more detail. In addition, ESA is planning to introduce a change in the data format during 2017 that shall remove the noise shaping in azimuth. These efforts are expected to improve the performance of the presented algorithm significantly.

Having a computationally efficient algorithm with adjustable vector positioning allows not only to provide near-real time operational drift data, but also the investigation of sea ice drift over large areas and long time periods. Our ~~nest task is to~~ next step is to embed the algorithm into a super-computing facility to further test the performance in different regions, time periods and ice conditions. The goal is to deliver large ice drift datasets and open-source operational sea ice drift products with a spatial resolution of less than 5 km.

This work is linked to the question how to combine the different timings of the individual image pairs in a most useful way. ~~This task is linked to the question how sea ice displacement relates to real sea ice velocity.~~ Having more frequent satellite acquisitions, as we ~~will~~ get with the Sentinel-1 satellite constellation, enables to derive displacements for shorter time gaps and the calculated vectors ~~are getting closer to the real sea ice velocity~~will reveal more details e.g. rotational motion due to tides. As part of a scientific cruise with KV-Svalbard in July 2016, we deployed three GPS trackers on loose ice floes and pack-ice in Fram Strait. The trackers send their position every ~~30~~5-30 min to deliver drift information with high temporal resolution. This efforts shall help to gain a better understanding of short-term drift variability and by comparison with calculated sea ice drift, we will investigate how ~~displacements~~ displacement vectors from subsequent satellite images relate to ~~real sea ice velocity~~sea ice displacements with higher temporal resolution.

The focus of this paper in terms of polarisation was put on the HV channel, since this polarisation provides on average four times more feature tracking vectors than HH and therefore delivers a finer

initial drift for the first guess. We found our area of interest well covered with HV images, but other areas in the Arctic and Antarctic reveal a better coverage in HH polarisation. Considering the four representative feature-tracking image pairs from Muckenhuber et al. (2016) , the the relatively best HH polarisation performance (i.e. most vectors from HH, while at the same time fewest vectors from HV) was the image pair that showed the least time difference, i.e. 8 h, compared to 31 h, 33 h and 48 h. Therefore, we assume that the HV polarisation provides more features that are better preserved over time. And more consistent features would also favour the performance of the pattern-matching step. However, at this point, this is just an assumption and will be addressed in more detail in our future work.

Utilising the advantage of dual polarisation (HH+HV) is certainly possible with the presented algorithm, but increases the computational effort. A simple approach is to combine the feature tracking vectors derived from HH and HV and produce a combined first-guess. Pattern-matching can be performed based on this combined first-guess for both HH and HV individually and the results can be compared and eventually merged into a single drift product. Having two drift estimates for the same position, from HH and HV pattern-matching respectively, would also allow to disregard vectors that disagree significantly. However, this option would increase the computational effort by two, meaning that the presented Fram Strait example would need about 8 min processing time.

After implementing the presented algorithm into a super-computing facility, we aim to test and compare the respective performance of HV, HH and HH+HV on large datasets to identify the respective advantages.

The current setting of the feature-tracking algorithm applies a maximum drift filter of 0.5 m/s. We found this to be a reasonable value for our time period and area of interest. However, when considering extreme drift situations in Fram Strait and a short time interval between image acquisitions, this threshold should be adjusted.

As mentioned above, we deployed three GPS tracker in Fram Strait and they recorded their positions with a temporal resolution of 5-30 min between 8^{th} July until 9^{th} September 2016 in an area covering 75° N to 80° N and 4° W to 14° W. Considering the displacements with 30 min interval, we found velocities above 0.5 m/s on a few occasions, when the tidal motion adds to an exceptionally fast ice drift.

The GPS data from the hovercraft expedition FRAM2014-2015 (https://sabvabaa.nersc.no), that was collected with a temporal resolution of 10 s between 31^{st} August 2014 until 6^{th} July 2015, did not reveal a single 30 min interval during which the hovercraft was moved by ice drift more than 0.45 m/s. The hovercraft expedition started at 280 km south from the North Pole towards the Siberian coast, crossed the Arctic Ocean towards Greenland and was picked up in the north-western part of Fram Strait.

In case the estimated drift from feature-tracking reaches velocities close to 0.5 m/s, the pattern-matching step might add an additional degree of freedom of up to 8 km, which could

eventually lead to a higher drift result than 0.5 m/s, depending on the time interval between the acquisitions. The smaller the time difference, the larger is the potentially added velocity. In order to be consistent when combining the drift information from several image pairs with different timings, one should apply a maximum drift filter on the final drift product of the presented algorithm that has the same maximum velocity as the feature-tracking filter. The corresponding function is implemented in the distributed open-source algorithm.

Appendix A: Open-source distribution

The presented sea ice drift retrieval method is based on open-source satellite data and software to ensure free application and easy distribution. Sentinel-1 SAR images are distributed by ESA for free within a few hours of acquisition under https://scihub.esa.int/dhus/. The algorithm is programmed in Python (source code: https://www.python.org) and makes use of the open-source libraries Nansat, openCV and SciPy. Nansat is a scientist friendly Python toolbox for processing 2-D satellite Earth observation data (source code: https://github.com/nansencenter/nansat). OpenCV (Open Source Computer Vision) is a computer vision and machine learning software library and can be downloaded under http://opencv.org. SciPy (source code: https://www.scipy.org) is a Python-based ecosystem of software for mathematics, science, and engineering. The presented sea ice drift algorithm is distributed as open-source software under https://github.com/nansencenter/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). We thank ~~the developer group of the Earth observation toolbox Nansat and Anton Korosov for his inputs. We also thank~~ Polona Itkin and Gunnar Spreen for providing us the buoy GPS data that were collected as part of the N-ICE2015 project with support by the Norwegian Polar Institute's Centre for Ice, Climate and Ecosystems (ICE) and its partner institutes. The used satellite data were provided by the European Space Agency. We thank Nuno Miranda for information on ESA's de-noising efforts for Sentinel-1. We also thank the developer group of the Earth observation toolbox Nansat and Anton Korosov for his inputs.

References

Berg, A. and Eriksson L.E.B.: Investigation of a Hybrid Algorithm for Sea Ice Drift Measurements Using Synthetic Aperture Radar Images IEEE Transactions on Geoscience and Remote Sensing, Vol. 52, No. 8, 5023–5033, 2014.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P.: BRIEF: Binary Robust Independent Elementary Features, CVLab, EPFL, Lausanne, Switzerland, 2010.

ESA: Sentinel-1 ESA's Radar Observatory Mission for GMES Operational Services, ESA Communications, SP-1322/1, ISBN: 978-92-9221-418-0, ISSN: 0379-6566, 2012.

Spreen G. and Itkin P.: N-ICE2015 buoy data, Norwegian Polar Institute, https://data.npolar.no/dataset/6ed9a8ca-95b0-43be-bedf-8176bf56da80, 2015.

Hollands, T.: Motion tracking of sea ice with SAR satellite data, dissertaiton, Section 2: Estimation of motion from images, University Bremen, 2012.

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 – Intergovernmental Panel on Climate Change: Climate Change 2013: The Physical Science Basis, Fifth Assessment Report, AR5, 317–382, 323–335, 2013.

Komarov, A.S., and Barber, D.G.: Sea Ice Motion Tracking From Sequential Dual-Polarization RADARSAT-2 Images, IEEE Transactions on Geoscience and Remote Sensing, Vol. 52(1), No. 1, 121–136, doi: 10.1109/TGRS.2012.2236845, 2014.

Korosov A.A., Hansen W.M., Dagestad F.K., Yamakawa A., Vines A., Riechert A.: Nansat: a Scientist-Orientated Python Package for Geospatial Data Processing, Journal of Open Research Software, 4: e39, DOI: http:// dx.doi.org/10.5334/jors.120, 2016

Kwok, R., Curlander J.C., McConnell R., and Pang S.: An Ice Motion Tracking System at the Alaska SAR Facility, IEEE Journal of Oceanic Engineering, Vol. 15, No. 1, 44–54, 1990.

Muckenhuber S., Korosov A.A., and Sandven S. (2016): Open-source feature-tracking algorithm for sea ice drift retrieval from Sentinel-1 SAR imagery, The Cryosphere, 10, 913?925913-925, doi:10.5194/tc-10-913-2016, 2016

Nansen, F.: The Oceanography of the North Polar Basin. Scientific Results, Vol. 3, 9, Longman Green and Co., Kristinania, Norway, 1902.

Pedersen, L.T., Saldo, R. and Fenger-Nielsen, R.: Sentinel-1 results: Sea ice operational monitoring, Geoscience and Remote Sensing Symposium (IGARSS), IEEE International, 2828–2831, doi=10.1109/IGARSS.2015.7326403, 2015

Rampal, P., Weiss, J., Marsan, D. and Bourgoin M.: Arctic sea ice velocity field: General circulation and turbulent-like fluctuations, Journal of Geophysical Research: Oceans, Vol. 114, Nr. C10014, doi=10.1029/2008JC005227, 2009.

Rampal, P., Weiss, J. and Marsan, D.: Positive trend in the mean speed and deformation rate of Arctic sea ice 1979–2007, Journal of Geophysical Research: Oceans, Vol. 114, Nr. C5013, doi=10.1029/2008JC005066, 2009b.

Rosten, E. and Drummond, T.: Machine learning for high-speed corner detection, in European Conference on Computer Vision, ISBN 978-3-540-33833-8, 430–443, doi: 10.100711744023_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), ISBN: 978-1-4577-1101-5, 2564–2571, doi: 10.1109ICCV.2011.6126544, 6–13 Nov, 2011.

Schubert, A. , Small, D., Meier, E., Miranda, N., and Geudtner, D.: Spaceborne Sar Product Geolocation Accuracy: A Sentinel-1 Update, Geoscience and Remote Sensing Symposium (IGARSS), IEEE International, 2675–2678, doi=10.1109

Spreen, G. and Itkin, P.: N-ICE2015 buoy data, Norwegian Polar Institute, https:/IGARSS.2014.6947025, 2014 /data.npolar.no/dataset/6ed9a8ca-95b0-43be-bedf-8176bf56da80, 2015.

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, 2008.

Thorndike, A. S. and Colony, R.: Sea ice motion in response to geostrophic winds, Journal of Geophysical Research: Oceans, Vol. 87, Nr. C8, 5845–5852, doi=10.1029/JC087iC08p05845, 1982.

Widell, K., Østerhus, S. and Gammelsrød, T.: Sea ice velocity in the Fram Strait monitored by moored instruments, Geophysical Research Letters, Vol. 30, Nr. 19, doi=10.1029/2003GL018119, 2003.