Application of SIFT algorithm in the sea ice image matching

Zhuo Chen^{1, a}, Shanshan Ma^{2,b}

¹ School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China

² School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China

^achenzhuo_qust@163.com, ^b490415312@qq.com

Abstract

In order to meet the needs of the marine development and maritime disaster reduction work, it is necessary to strengthen the monitoring and forecasting of sea ice. Doing sea ice drift detection can provide accurate field data for the numerical prediction of North Sea and medium-range forecasts, which is an effective way to improve forecast accuracy and reducing ice disasters. Use feature point extraction algorithm and image matching algorithm to study the ice drift speed and direction extraction method, which is the preparation of sea ice drift monitoring technology research. In many image matching algorithms, SIFT algorithm has good scale invariance of rotation, spatial rotation invariance and strong matching ability. SIFT and its extension algorithm has been proved to be the robust in the same descriptor, which is currently a hot research at home and abroad. This paper introduces four main steps of the SIFT algorithm and its application in the sea ice image matching feature points.

Keywords: sea ice; SIFT; scale space; difference of Gaussian; key points; feature descriptor

Introduction

In a variety of image features, the corner feature concentrated many important information of the shape in images, and it also has the advantage of rotation invariant and it is not affected by the lighting conditions. These features make the corners match suited to the actual application needs of sea ice drift velocity flow measurement. Using corner feature to match calculations can reduce the amount of computation without losing the important gray scale information of the image at the same time, and it can greatly improve the matching speed.

Harris etc[1] have taken advantage of local features to achieve the robust corner detection. Using region descriptors with rotation invariant to describe the detected Harris corners, Schmid etc[2] have achieved rotational invariance of focus point matching. However, Harris corner cannot achieve a perfect match. So David Lowe[3] proposed scale invariance of local features to describe the image characteristics, namely SIFT (Scale Invariant Feature Transforms) feature matching. SIFT is a computer vision algorithm used to detect and describe the local characteristics of the image; it looks at the extreme points in spatial scale and extracts its position, scale and rotation invariant.

The idea of SIFT algorithm

SIFT algorithm is based on the idea of image feature scale selection, and it established the multi-scale space of image, then it will detect the same feature point at different scales and determine the position of the feature point with its scale at the same time so that it can reach the purposes of the scale anti-scaling. Eliminating some point as well as low-contrast edge response points and extracts rotation invariant feature descriptor in order to achieve the purpose of anti-affine transformation. The following is the main idea to obtain the image feature points:

1. Detection of scale-space extreme: by Gaussian filtering the original image and establishing a Gaussian pyramid, and then achieving a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation and they are likely to play a role in subsequent image matching feature points.

2. Keypoint localization and scale orientation: the points founded by step 1 could be real keys which are only based on measures of their stability.

3. Keypoint orientation assignment: distribute gradient direction and gradient mode for the key points according to the gradient direction of each key points of neighborhood.

4. Keypoint feature descriptor: the feature points are described by the form of multi-dimensional vector.

SIFT feature point matching algorithm steps

The image SIFT feature vector generation algorithm includes a total of four steps: scale space extrema detection, keypoint localization and scale orientation, keypoint orientation assignment, keypoint feature description. Finally, complete the feature points matching by feature vector.

1. Image scale space construction, extrema detection

First of all, to make an image with scale space invariance, we need to create scale space. SIFT algorithm uses a Gaussian function to establish the scale space, and Gaussian function is defined as shown in Eq. 1:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2 + y^2)} / 2\sigma^2.$$
 (1)

The scale space of an image is defined as shown in Eq. 2:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y).$$
(2)

The symbols: L is a blurred image; G is the Gaussian Blur operator; I is an image; (x,y) are the location coordinates; σ is the "scale" parameter. Think of it as the amount of blur. Greater the value is, greater the blur. The * is the convolution operation in x and y. It "applies" Gaussian blur G onto the image I.To efficiently detect stable keypoint locations in scale space, difference of Gaussian scale space[3] was proposed, which is generated by using different scales of Gaussian differential nuclear and image convolution and is defined as shown in Eq. 3:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$

= $L(x, y, k\sigma) - L(x, y, \sigma)$

(3)

2. Keypoint localization and scale orientation

The key points consist of the local extreme points of the DOG space. Preliminary exploration of the key points is completed through the DOG comparison between two adjacent images in the same group. By fitting three-dimensional quadratic function location and scale of key points (up to sub-pixel accuracy), while removing low-contrast edges and unstable critical point response points to enhance the matching stability and improve noise immunity.

A poorly defined peak in the difference-of-Guassian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal curvatures can be computed from a 2×2 Hessian matrix, H, computed at the location and scale of the keypoint. The derivatives are estimated by taking differences of neighboring sample points. Let α be the eigenvalue with the largest magnitude and β be the smaller one. Let γ be the ratio between the largest magnitude eigenvalue and the smaller one, so that $\alpha = \gamma\beta$, and the derivative can be defined as shown in Eq. 4, which depends only on the ratio of the eigenvalues rather than their individual values.

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r}$$
(4)

3. Keypoint orientation assignment

The idea is to collect gradient directions and magnitudes around each keypoint. Then we figure out the most prominent orientation(s) in that region. And we assign this orientation(s) to the keypoints. Any later calculations are done relative to this orientation. This ensures rotation invariance. Gradient magnitudes and orientations are calculated using Eq. 5 and Eq. 6, and L is the scale used for each point where it is in:

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^{2} + (L(x, y + 1) - L(x, y - 1))^{2}}$$
(5)

$$\theta(x, y) =$$

$$\tan^{-1}((L(x, y + 1) - L(x, y - 1))/L(x + 1, y) - L(x - 1, y)))$$
(6)

4. Keypoint feature descriptors

First, consider the axis of rotation as the direction of the key points to ensure rotational invariance. Each keypoint contains three kinds information: location, scales and direction. After getting the main direction of the neighborhood of the point, the key point is described as feature vectors to make the key points be numeric .In the neighborhood of the keypoints, the size in units of 4×4 as a seed

point, after using the histogram method the seed point is described as 8-dimensional vector, that is, an interval of 45 degrees each. Thus the key points around each with 2×2 seed points, each seed point is repressed as 8-dimensional vector. These key seed points can be represented as 32-dimensional vector.

5. SIFT feature vectors matching

After the two images SIFT feature vector generation, the next step requires the use of the Euclidean distance of image 1 is nearest to image 2. In these two key points, if the closest distance divided by time close distance of less than the proportion of the threshold, the acceptance of this pair of matching points. Reduce this ratio threshold, the number of SIFT matching points will be reduced, but more stable. In order to exclude the key block and the background because the image generated chaotic no match relationship, Lowe proposed a comparison with the second nearest neighbor distance of the nearest neighbor distance ratio is less than a threshold value considered to be the correct match.

Experimental results and analysis

In this experiment, with Matlab7.0 in Intel (R) Core (TM) CPU 2.2GHz, 2G of memory, the operating system for computer programming on windows7.

1. SIFT algorithm extracts ice images feature points



Fig.1 Ice image SIFT feature points detection algorithm results

The ice images feature points result extracted by SIFT algorithm are shown as Fig.1(a) and (b). Among them, the size of the original image Fig.1 (a) is 2 times of the size of original Fig.1 (b). Obviously, SIFT algorithm to extract the SIFT feature vector for scale to maintain a certain degree of stability scaling.

2. Two ice images SIFT algorithm matching results

Two ice images SIFT algorithm matching results with different threshold ratio are shown as Fig.2(a), (b) and (C).



(a) threshold ratio is 0.8 (b) threshold ratio is 0.6 (c) threshold ratio is 0.4

Fig.2 Ice image SIFT algorithm matching results with different threshold ratio

Table	1	Ice	image	feature	point	detection	and	matching	comparison	of statistical
results										

Image name	threshold ratio	feature points number	feature point matching	t matching time
			number	
Image1. jpg	0.8	450	196	0.197 s

Image2. jpg		205		
Image2. jpg	0.6	450	136	0.168 s
Image2. jpg		205		
Image1. jpg	0.4	450	93	0.140 s
Image2. jpg		205		

As can be seen from Fig. 2 and Table 1, threshold radio affects the numbers of images SIFT matching points and the feature point matching time. If reduce this ratio threshold, the numbers of SIFT matching point and feature point matching time will be significantly reduced. For error matching, due to high dimensional feature space, there are probably huge quantities of other similar distance error matching, thus its ratio value is high. Lowe [3] recommended that the ratio threshold value is 0.8, although it will remove about 5% of the correct match, but it will remove approximately 90% of false matches. However, the best ratio to take in the sea ice images matching still need to go through experiment for further verification.

Conclusion

SIFT algorithm is suitable for fast and accurate matching in the database with mass characteristics, but the matching accuracy is very low when it is under the similar circumstances or symmetrical environment. Harbert Bay has proposed a feature exaction algorithm[4] which is an accelerated version of the SIFT algorithm and it has performance advantages that the SIFT never gets in the speed space. Chen Yu has proposed a method to match and position with the connection of the SIFT algorithm and Mahalanobis Distance [5]. Yang Xiaomin and others have proved that the initial feature matching of SIFT using the nearest neighbor algorithm will still give a relative nearest neighbor as its matching points[6]. In this paper ,the author has studied the SIFT algorithm and then apply it to the sea ice image feature point matching .The experimental data only has some experimental reference, so we will need further improvement in order to get some better result.

Acknowledgements

This work was financially supported by the Shandong Province Higher Educational Science and Technology Program(J14LN74), Open Research Fund of Shanghai Natural Science Foundation of Shandong Provincial Key Laboratory of Marine Ecology and Environment & Disaster Prevention and Mitigation (2012006).

References

- Harris C, Stephens M. A combined corner and edge detector[C], Proceedings of Fourth Alvey Vision Conference, Manchester, UK(1988), p.147–151.
- [2] Schmid C, Mohr R. in: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19(1997), p. 530–534.
- [3] Lowe D G., in: International Journal of Computer Vision, Vol.60 (2004), p. 91-110.
- [4] Herbert Bay Andreas Ess., in: Computer Vision and Image Understanding, Vol. 110(2008), p. 346-359.
- [5] CHEN Y., in: Geomatics & Spatial Information Technology, Vol. 32(2009), p. 51-52.
- [6] YANG X M., in: Optics and precision engineering, Vol.17 (2009), p.2279-2280.