SIFT Algorithm Analysis and Optimization

Zhu Daixian

Communication and Information Engineering College, Xi'an University of Science and Technology
Xi'an, China
zh_daixian@126.com

***Abstract*—Due to good invariance of scale, rotation, illumination, SIFT (Scale Invariant Feature Transform) descriptor is commonly used in image matching. The steps of extracting SIFT feature are analyzed in detail, and SIFT Key-point location is optimized. The Chamfer distance is used in this article; it decreases computation time and improves the accuracy of image matching. The experimental results show that the algorithm can reduce time complexity and maintain robust quality at the same time.**

***Keywords*—SIFT, analysis and optimization, Comparability measurement, Chamfer distance, image matching**

I. INTRODUCTION

SIFT (Scale Invariant Feature Transform) algorithm is successful in feature matching research areas. The features extracted by SIFT are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint. Images with the larger differences can be matching by the steady features. But the time complexity of the algorithm is relatively high. At the same time, real-time processing and matching characteristics accuracy of the algorithm are expected to be improved. In image processing field, this problem is still not completely solved.

Several scholars have researched some methods to optimize this algorithm. A simplified algorithm described by Li Liu [1] used a feature point with only 12 dimensions based on a circular window to improve the efficiency of matching, but robustness against noise is weaker than original algorithm. Shen Qian [2] improved the computation speed by dynamically modifying sampling step when computing the gradient histogram of the region around the key point location. Zhu Daixian [3] proposes a method, which used linear combination of city-block distance and chessboard distance in place of Euclidean distance in matching process. The generation process of SIFT is analysed firstly, then The Slant distance is used to measure comparability of feature descriptors. Matching algorithm time is reduced and the accuracy of image matching is also improved.

II. SIFT FEATURE EXTRACTION ANALYSIS AND OPTIMIZATION

There are four steps to extract SIFT feature [4]: Scale-space generation, Detection of Scale-Space Extrema, Feature point location, and SIFT Descriptor Generation.

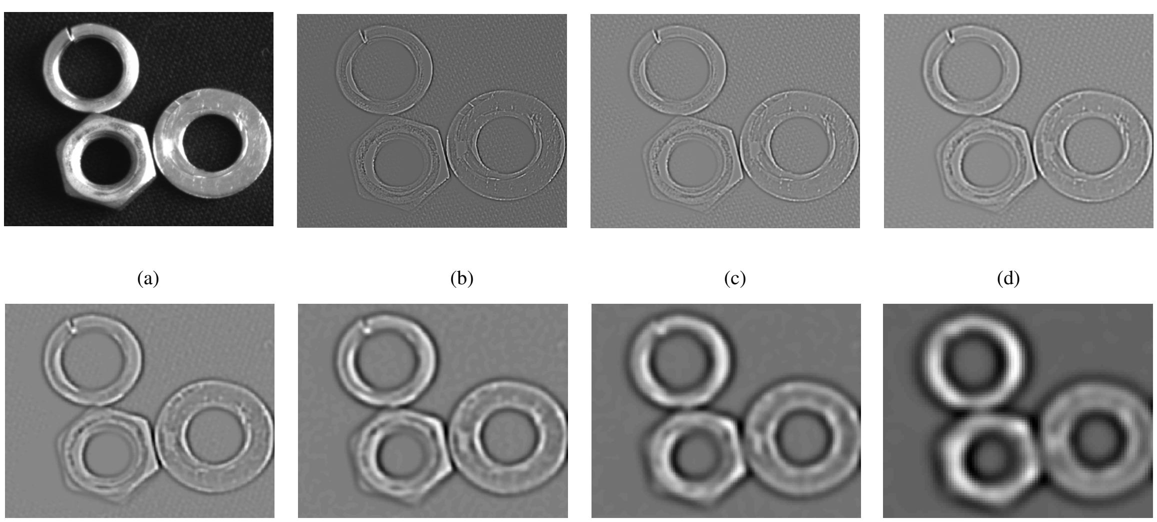
A. Scale-Space Generation

The scale space [5] is generated by different scale Gaussian filters $G(x,y,\sigma)$ with factor K of the original image. The filtered image $L(x,y,k\sigma)$ is called Gaussian pyramid image .

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

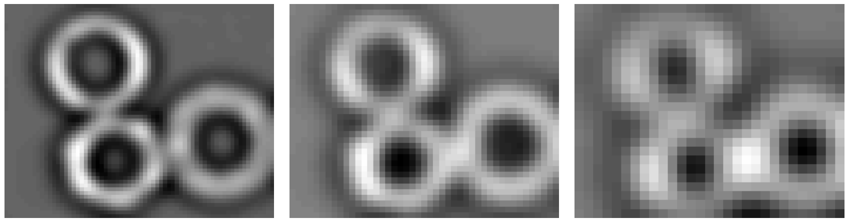
$$L(x,y,k\sigma)=G(x,y,k\sigma)*I(x,y) \tag{2}$$

Scale space images are showed in Fig.1.



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

(e) (f) (g) (h)



(i) (j) (k)

Fig.1 Scale space images

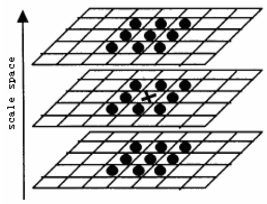


Fig.2 Scale-space extrema detection

B. Detection of Scale-Space Extrema

This is implemented efficiently by using a DOG (difference-of-Gaussian) function to identify potential interest points which are invariant to scale and orientation in scale space with the image $D(x, y, \sigma)$, which called DOG scale space and can be computed from the difference of two nearby scales.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{3}$$

Each pixel in the middle layer (the bottom and the top level exception) compared with eight adjacent pixels in the same layer, and compared with various nine adjacent pixels of its superior layer and the lower layer in DOG scale space, there is 26 points total, as can be seen from Fig.2, to ensure local extrema in the scale space and the space of two-dimensional images. The location and scale of the extremum is recorded as $D(x_0, y_0, \sigma)$.

C. Feature Point location Optimization

Once a key-point candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale. Located key-points accurately at the location and scale of the central sample point uses the Taylor expansion (up to the quadratic terms) of the scale-space function $D(x, y, \sigma)$, shifted so that the origin is at the sample point $D(x_0, y_0, \sigma)$ [4]:

$$D(x, y, \sigma) = D(x_0, y_0, \sigma_0) + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X \tag{4}$$

In (4), $X = (x, y, \sigma)^T$, $\frac{\partial D}{\partial X} = \begin{bmatrix} \frac{\partial D}{\partial x} \\ \frac{\partial D}{\partial y} \\ \frac{\partial D}{\partial \sigma} \end{bmatrix}$,

$$\frac{\partial^2 D}{\partial X^2} = \begin{bmatrix} \frac{\partial^2 D}{\partial x^2} & \frac{\partial^2 D}{\partial x \partial y} & \frac{\partial^2 D}{\partial x \partial \sigma} \\ \frac{\partial^2 D}{\partial y \partial x} & \frac{\partial^2 D}{\partial y^2} & \frac{\partial^2 D}{\partial y \partial \sigma} \\ \frac{\partial^2 D}{\partial \sigma \partial x} & \frac{\partial^2 D}{\partial \sigma \partial y} & \frac{\partial^2 D}{\partial \sigma^2} \end{bmatrix},$$

and we can deduct other expressions using pixel differences, they are

$$\frac{\partial D}{\partial x} = (D_n^{x+1,y} - D_n^{x-1,y})/2 \quad , \frac{\partial D}{\partial \sigma} = (D_{n+1}^{x,y} - D_{n-1}^{x,y})/2 ,$$

$$\frac{\partial D}{\partial y} = (D_n^{x,y+1} - D_n^{x,y-1})/2 ,$$

$$\frac{\partial^2 D}{\partial y^2} = [(D_n^{x,y+1} - D_n^{x,y}) - (D_n^{x,y} - D_n^{x,y-1})]/4 \quad ,$$

$$\frac{\partial^2 D}{\partial \sigma^2} = [(D_{n+1}^{x,y} - D_n^{x,y}) - (D_n^{x,y} - D_{n-1}^{x,y})]/4$$

$$\frac{\partial^2 D}{\partial x^2} = [(D_n^{x+1,y} - D_n^{x,y}) - (D_n^{x,y} - D_n^{x-1,y})]/4,$$

$$\frac{\partial^2 D}{\partial x \partial y} = [(D_n^{x+1,y+1} - D_n^{x-1,y+1}) - (D_n^{x+1,y-1} - D_n^{x-1,y-1})]/4$$

$$\frac{\partial^2 D}{\partial y \partial x} = [(D_n^{x+1,y+1} - D_n^{x+1,y-1}) - (D_n^{x-1,y+1} - D_n^{x-1,y-1})]/4 \quad ,$$

$$\frac{\partial^2 D}{\partial \sigma \partial y} = [(D_{n+1}^{x,y+1} - D_{n-1}^{x,y+1}) - (D_{n+1}^{x,y-1} - D_{n-1}^{x,y-1})]/4$$

$$\frac{\partial^2 D}{\partial y \partial \sigma} = [(D_{n+1}^{x,y+1} - D_{n+1}^{x,y-1}) - (D_{n-1}^{x,y+1} - D_{n-1}^{x,y-1})]/4 ,$$

$$\frac{\partial^2 D}{\partial x \partial \sigma} = [(D_{n+1}^{x+1,y} - D_{n+1}^{x-1,y}) - (D_{n-1}^{x+1,y} - D_{n-1}^{x-1,y})]/4$$

$$\frac{\partial^2 D}{\partial \sigma \partial x} = [(D_{n+1}^{x+1,y} - D_{n-1}^{x+1,y}) - (D_{n+1}^{x-1,y} - D_{n-1}^{x-1,y})]/4$$

here, n denote currently layer, $n-1$ denote superior layer and $n+1$ denote lower layer.

The location of the extremum, X_M, is determined by taking the derivative of (4)with respect to **x** and setting it to zero. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge, to enhance matching stability and to improve the resistance to noise. As suggested by Brown, the Hessian and derivative of the scale-space function $D(x, y, \sigma)$ are approximated by using differences of neighboring sample points. The eigenvalue of **H** is proportional to the principal curvatures of $D(x, y, \sigma)$, Reference [5] indicated the confine of the ratio, if the $ratio \geq 12.1$, the key-point is eliminated.

$$ratio = \frac{\left(\frac{\partial^2 D}{\partial x^2} + \frac{\partial^2 D}{\partial y^2}\right)^2}{\frac{\partial^2 D}{\partial x^2}\frac{\partial^2 D}{\partial y^2} - \left(\frac{\partial^2 D}{\partial x \partial y}\right)^2} \tag{5}$$

Described as above, computing first three the second derivative of the local extremum points to determine whether to retain this point or not. For the reserved points, computing second the X_M, with taking the derivative of (4) , we can deduct (5):

$$X_M = -\left(\frac{\partial^2 D}{\partial^2 X}\right)^{-1} \frac{\partial D}{\partial X} \tag{6}$$

In this step, only nine remainder derivatives need to be computed, if the first three second derivatives saved. As suggested by Lowe, The function value at the extremum, $D(X_M)$, is useful for rejecting unstable extrema with low contrast. This can be obtained by substituting (6) into (4), giving

$$D(X_M) = D(x_0, y_0, \sigma_0) + \frac{1}{2}\frac{\partial D^T}{\partial X} X_M \tag{7}$$

If $\left|D(X_M)\right| \geq 0.03$, the extrema should be retained. With the optimization of Key-point orientation accurately, the computing time of Key-point orientation is shorter than Lowe's. The located feature points are showed in Fig. 3.

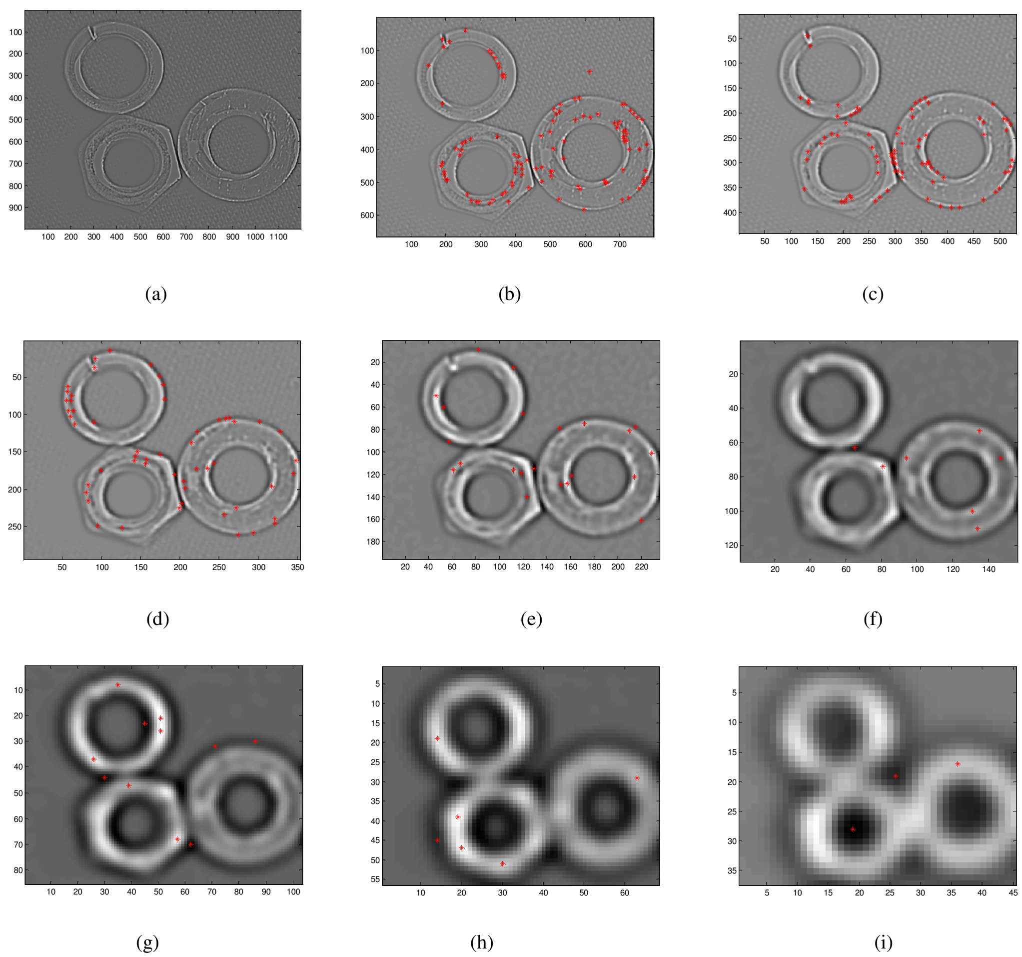


Fig.3 Feature points location

D. The SIFT Descriptor

By assigning a consistent orientation to each key point based on local image properties, the key-point descriptor can be represented relative to this orientation and achieve invariance to image rotation. For each key-point, an orientation histogram is formed from the gradient orientations of sample points within a region round the key-point. The orientation histogram has 36 bins covering the 360 degree range of orientations. Peaks in the orientation histogram correspond to dominant directions of local gradients. The highest peak in the histogram is detected, and then any other local peak that is within 80% of the highest peak is used to also create a key-point with that orientation.

The local image gradients are measured at the selected scale in the region around each key-point. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

In order to achieve orientation invariance, the coordinate of the descriptor and the gradient orientations are rotated relative to the key-point orientation. Select a 16×16 rectangular block pixels around a key-point, and divided it into 16 4×4 sub-regions. the gradient magnitude and orientation of each 4×4 sub-regions pixels are computed using pixel differences, These samples are then accumulated into orientation histograms summarizing the contents over 4×4 sub-regions, These are weighted by a Gaussian window, indicated by the overlaid circle. each orientation histogram is distributed into 0,45,90,135,180,225,270,315 angle eight direction projection, the length of each angle histogram projection is figure with arrow corresponding to the sum of the gradient magnitudes near that direction within the region. The key-point is expressed by 128 dimensions feature vector showed in Fig.4. Finally, the feature vector is normalized to unit length to reduce the effect of illumination change.

The key-point descriptors showed in Fig.5. are highly distinctive, which allows a single feature to find its correct match with good probability in a large database of features.

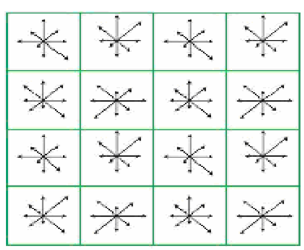


Fig.4 128- dimension feature vector

III. SIFT FEATURE MATCHING IMPROVEMENT

The best candidate match for each key-point is found by identifying its nearest neighbor in the database of key-points from training images. The nearest neighbor is usually defined as the key-point with the minimum Euclidean distance. Calculating Euclidean distance [6], it needs n times multiplication computing and one times square root computing, so the time of matching is large. Distance Transform image of Chamfer Distance is similar with Euclidean distance, and the computational of Chamfer Distance is much smaller than that of Euclidean distance. The Chamfer distance is comparability measurement of SIFT feature matching in this article. Distance Transform showed in Fig.6.

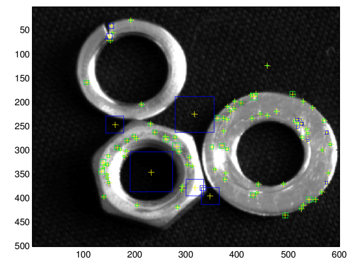
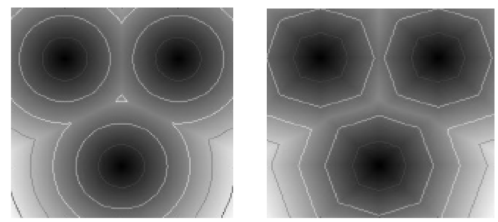


Fig.5 Key-point descriptor



(a) Euclidean distance (b) Chamfer distance

Fig.6 Distance transform

Our key-point descriptor has a 128-dimensional feature vector. The exhaustive search algorithm works particularly well for this problem in that we only consider matches in which the nearest neighbor is less than 0.8 time the distance to the second-nearest neighbor.

IV. EXPERIMENT RESULTS

The above algorithm is achieved with program language MATLAB in this paper. Fig. 7.show an example of two images matching, two images matching time used Lowe’s method is 2950ms, but the time used in the proposed method in this article is 1020 ms.

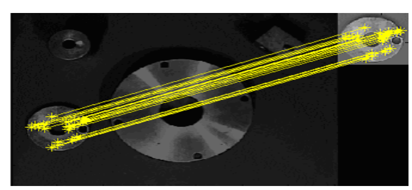
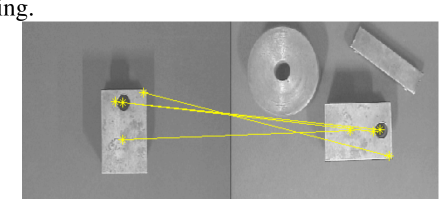
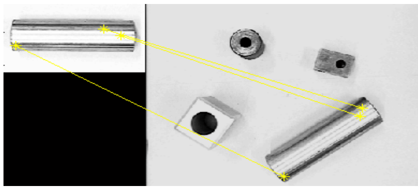


Fig.7 An example of two images matching

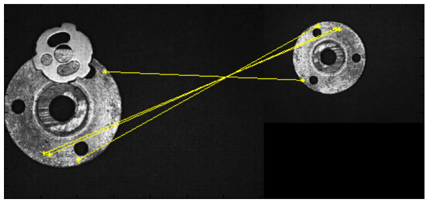
The effect of part image matching is showed in Fig.8, which screened in the different zoom scale, lighting conditions and rotate angle and Binocular vision matching.



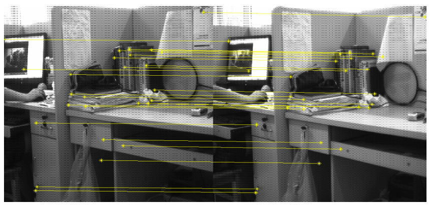
(a) Zoom scale and rotation image matching



(b) Illumination image matching



(c) Rotation image matching



(d) Binocular image matching

Fig.8 matching image.

V. CONCLUSIONS

SIFT image matching algorithm is analyzed profoundly in this paper. The analysis of feature extraction can be used in the intelligent computing of this algorithm. Chamfer distance in place of Euclidean distance reduces the time of the algorithm in key-point matching. Real-time quality and stability of algorithm are enhanced.

REFERENCES

[1] Liu Li, Peng Fuyuan, Zhao Kun, “Simplified SIFT algorithm for fast image matching,” Infrared and Laser Engineering, vol.37, pp.181–184,Jan. 2008.

[2] Qian Shen, Zhu Jianying, “ Improved SIFT-based bi-directional image matching algorithm,” Mechanical Science and Technology for Aerospace Engineering, Vol.26,no.9,pp. 1179-1182, Sep. 2007.

[3] Zhu Daixian, Wang Xiaohua, “A method of Improving SIFT algorithm matching efficiency,” in Proc.CISP’ 09, 2009, vol.2, p.891.

[4] Lowe D.G. “Instinctive image features from scale-invariant key-points,” International Journal of Computer Vision. 2004, p. 91~110.

[5] Lindeberg.T, “Scale-space theory: A basic tool for analyzing structures at different scales,” Journal of Applied Statistics, Vol.21,no.2,pp. 224–270, Feb.1994.

[6] Hua Luo-geng. An introduction of higher mathematics, other issues, Beijing, China: Science Press, 1984