Image Feature Extraction, Matching And Constrain Estimation Based On SIFT

Zhang Feiyu, *Wuhan University, Wuhan, Hubei, 430079, PRC*
Xie Bingsheng, *Wuhan University, Wuhan, Hubei, 430079, PRC*

Abstract—
This paper describes the principle of Scale Invariant Feature Transform(SIFT) algorithm to achieve image feature extraction and feature matching between the same images with different scales, and it is implemented by Matlab code. At the same time, the constraints of each pair of images are estimated and verified.

Based on the course of Gao Zhi and the seminal paper of David G.Lowe, we summarized this article after practice as the phased learning outcome of the course of computer vision. This paper includes the introduction of SIFT algorithm, SIFT algorithm to achieve image feature extraction and matching principle, based on the previous steps to estimate the constraints of each pair of pictures, as well as implementation and description based on Matlab. See the full code at github.com/SEdges/CV-Big-Assignment.

In his paper, David G.Lowe gave little or no details about the implementation of SIFT algorithm, which brought great difficulties to the implementation. After Mr. Gao Zhi's lecture, combined with handouts and network data, we finally basically completed the implementation of SIFT algorithm main parts, including the application of scale space theory to build a Gaussian difference pyramid, and the key points of exploration, at the same time by fitting three-dimensional quadratic function to accurately determine the location and scale of key points. At the same time, the key points with low contrast and unstable edge response points are removed (because the DoG operator will produce strong edge response) to enhance the matching stability and improve the anti-noise ability. Finally, the characteristics of the key points are described, and the Euclidean distance is used to match the corresponding key points between the same image with different scales.

In the attempt to realize the feature point detection and matching with scale invariance, we considered using the Hessian Matrix with low time complexity to extract the feature points, and then using the non-maximum suppression to determine the feature points initially and accurately locate the feature points. However, due to unknown reasons, the gradient direction of local pixels is not accurate and the matching is not successful. Therefore, we finally chose the most stable SIFT algorithm.

Digital Object Identifier 10.1109/XXX.0000.0000000

BRIEF INTRO TO SIFT

Scale-invariant feature transform (Scale-invariant Feature Transform or SIFT) is a computer vision algorithm used to detect and describe local features in an image. It finds the extreme point in the spatial Scale and extracts its position, scale and rotation invariants. This algorithm was published by David Lowe in 1999 and summarized in 2004. Its applications include object recognition, robot map perception and navigation, image stitching, 3D model building, gesture recognition, image tracking and motion comparison.

The algorithm has a patent, owned by the University of British Columbia.

The essence of SIFT algorithm is to find the key points (feature points) in different scale Spaces, and calculate the direction of the key points. SIFT to find the key points are some very prominent, will not change due to lighting, affine transformation and noise factors, such as corner points, edge points, dark areas and bright areas of dark points.

IMAGE FEATURE EXTRACTION

David G.Lowe decomposed the SIFT algorithm into the following four steps:

1. Scale space extreme value detection: search image positions on all scales. Gaussian differential functions are used to identify potential points of interest that are invariant to scale and rotation.

2. Key point localization: At each candidate location, the location and scale are determined by a finely fitted model. Key points are chosen according to their stability.

3. Direction determination: One or more directions are assigned to each key point location based on the local gradient direction of the image. All subsequent operations on the image data transform with respect to the direction, scale, and position of the key points, thus providing invariance to these transformations.

4. Description of key points: The local gradient of the image is measured on the selected scale in the neighborhood around each key point. These gradients are transformed into a representation that allows for relatively large local shape deformation and illumination changes.

SCALE SPACE EXTREME VALUE DETECTION

Lowe used the Gaussian difference pyramid approximate LoG operator to detect the key points of stability in the scale space. The scale space of an image $L(x, y, \sigma)$ is defined as the convolution of a scaled Gaussian $G(x, y, \sigma)$ with the original image $I(x, y)$.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1}$$

Where

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

In 2002, Mikolajczyk found that the maximum and minimum values of the scale normalized Gaussian Laplacian function $\sigma^2 \bigtriangledown^2 G$ can produce the most stable image features in a detailed experimental comparison. The Difference of Gaussian (DOG) operator is very close to the scale-normalized Gaussian Laplace function, where

$$\sigma \bigtriangledown^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma} \tag{3}$$

So

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \bigtriangledown^2 G \tag{4}$$

Where k-1 is a constant. Therefore, the Gaussian difference operator can be calculated as

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{5}$$

In the actual calculation, the pyramid construction method is as follows :1. The image is blurred by Gaussian at different scales; 2. Down-sample the image (sampling at intervals). The Gaussian difference image is obtained by subtracting the images of the upper and lower adjacent layers of the Gaussian pyramid in each group.

The keypoints consist of local extreme points in the DOG space. Each pixel is compared with all its neighbors (26 points in total, 8 points at the same scale and 9×2 points at the upper and lower adjacent scales). Due to the comparison between adjacent scales, for the Gaussian difference pyramid with four layers in each group, the extreme point detection of two scales can only be performed in the middle two layers, and the other scales can only be performed in different groups. In order to detect S scale extreme points in each group, DOG pyramid needs S+2 layers of images for each group, while DOG pyramid is obtained by subtracting two adjacent layers of Gaussian pyramid, and Gaussian pyramid needs S+3 layers of images for each group, and S is between 3 and 5 in actual calculation. The parameters that need to be determined to build the scale space are:

σ-Scale space coordinates.

O-Octave.

S-The number of layers in one octave. Set to 3.

Where the relation between σ and O, S is as follows.

$$\sigma(o, s) = \sigma_0 2^{\sigma + \frac{s}{S}} o \in [0, ..., O-1], s \in [0, ..., S+2] \tag{6}$$

The above formula can be used to determine the scale coordinates of the key-points. Where σ_0 is the base layer scale of 1.6, o is the index of octave, and s is the index of the inner layer of the group.

The number of levels of the pyramid is jointly determined according to the original size of the image and the size of the image at the top of the tower. In addition, in (4)

$$k = 2^{\frac{1}{S}} \tag{7}$$

When constructing the Gaussian pyramid, the scale coordinates of each level within the octave are calculated as follows:

$$\sigma(s) = \sqrt{(k^s \sigma_0)^2 - (k^{s-1}\sigma_0)^2} \tag{8}$$

To calculate the scale of a layer in an octave, the following formula is used directly:

$$\sigma_o ct(s) = \sigma_0 2^{\frac{s}{S}} s \in [0, ..., S+2] \tag{9}$$

Therefore, the equation can be denoted as

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,\sigma(s+1)) - G(x,y,\sigma(s))) * I(x,y) \\ &= L(x,y,\sigma(s+1)) - L(x,y,\sigma(s)) \end{aligned} \tag{10}$$

KEY POINT LOCALIZATION

The extreme points detected by the above methods are the extreme points of the discrete space. The extreme points of the discrete space are not really extreme points. The method of using known discrete spatial points to obtain continuous spatial extreme points is called Sub-pixel Interpolation.

In order to improve the stability of key points, curve fitting of the scale space DoG function is required. The Taylor expansion (fitting function) of the DoG function in scale space is as follows.

$$D(X) = D + \frac{\partial D^T}{\partial X}X + \frac{1}{2}X^T\frac{\partial^2 D}{\partial X^2}X \tag{11}$$

Where

$$X = (x,y,\sigma)^T \tag{12}$$

The offset of the equation is obtained by taking the derivative, and the precise position of the feature point can be obtained by several iterations.

DIRECTION DETERMINATION

In order to make the descriptor rotation-invariant, the local features of the image should be used to assign a reference orientation to each key-point. The stable orientation of local structure is obtained by image gradient method. For the key points detected in the DOG pyramid, the gradient and orientation distribution characteristics of the pixels in the 3σ neighborhood window of the Gaussian pyramid image where they are located are collected. The magnitude and direction of the gradient are as follows:

$$\begin{aligned} m(x,y) &= \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \\ \theta(x,y) &= \tan^{-1}((L(x,y+1) - L(x,y-1))/L(x+1,y) - L(x-1,y)) \end{aligned} \tag{13}$$

L is the scale space value of the key-point, according to Lowe's proposal, the gradient magnitude m(x,y) is added according to the Gaussian distribution of $\sigma = 1.5\sigma_o ct$, according to the scale sampling principle of 3σ, the neighborhood window radius is $3 \times 1.5\ Sigma_o ct$.

After completing the gradient calculation of the key-point, the histogram is used to count the gradient and orientation of the pixels in the neighborhood. The gradient histogram divides the range of 0-360 degrees into 36 bins of 10 degrees each.

At this point, the detected key-points containing position, scale, and orientation are the SIFT feature points of the image.

DESCRIPTION OF KEY POINTS

Lowe proposed that the descriptor use gradient information in eight directions computed in a 4*4 window within the key-point scale space, for a total of 4*4*8=128 dimensional vector representation. The representation steps are as follows:

1. Determine the image region needed to compute the descriptor.The feature descriptors are related to the scale of the feature points, so the gradient should be computed on the Gaussian image corresponding to the feature points. The neighborhood around the keypoint is divided into d*d(Lowe suggests d=4) sub-regions, and each sub-region is used as a seed point, and each seed point has eight directions.

2. Each sub-region is assigned a rectangular area of side length to sample.

3. The sampling points in the neighborhood are assigned to the corresponding sub-region, the gradient values in the sub-region are assigned to eight directions, and their weights are calculated.

The coordinates of the rotated sample points are assigned to $d \times d$ sub-regions within a circle of radius, and the gradients and orientations of the sample points affecting the sub-regions are calculated, assigned to eight directions. Interpolation Calculates gradients in eight directions for each seed point.

5. After the formation of feature vectors, in order to remove the influence of illumination changes, they need to be normalized. For the overall drift of image gray value, the gradient of each point in the image is obtained by subtracting the neighborhood pixels, so it can also be removed. The resulting descriptor vector is $H = (h1, h1, ..., h128)$, the normalized feature vector is $L = (l1, l2, ..., l128)$.

6. Descriptor vector threshold. Nonlinear illumination and the change of camera saturation cause the gradient value of some directions to be too large, but the influence on the direction is weak. Therefore, the threshold value (usually 0.2 after vector normalization) is set to cut off large gradient values. Then, another normalization process is performed to improve the discriminative power of the features.

7. Sort the feature description vectors by the scale of the feature points.

IMAGE FEATURE MATCHING

As for the key-point matching problem, it has now been transformed into the KPD matching problem, and the similarity degree of two KPDs is calculated using the Euclidean distance. Let two KPDs be $R = (r1, r2, ...$ and $S = (s1, s2, ..., s128)$, the Euclidean distance between R and S is calculated as follows.

$$d = \sqrt{(r1 - s1)^2 + (r2 - s2)^2 + ... + (r128 = s128)^2} \tag{14}$$

So, to find the correspondences (key-points that have a relationship) between two images at different scales, we do the following:

1.Detect the key-points of the two images respectively, and calculate the KPD of each key-point to obtain two KPD sets SET1 and SET2.

2.For each KPD in SET1, find the best match from SET2 (that is, the one with the smallest Euclidean distance is the best match), and then conversely, for each KPD in SET2, find the best match from SET1; only those KPD pairs that are considered to be the best match are the corresponding points.

3.To improve the accuracy of matching, we can set a threshold, and those pairs whose Euclidean distance is greater than this threshold will not be considered. To make the algorithm more efficient, we can use KD-tree and Random Sample Consensus (RANSAC).

CONSTRAINTS ESTIMATION

SEE GITHUB

REFERENCES

1. David G.Lowe Distinctive Image Features from Scale-Invariant Keypoints. January 5, 2004.
2. David G.Lowe Object Recognition from Local Scale-Invariant Features. 1999
3. Matthew Brown and David Lowe Invariant Features from Interest Point Groups. In British Machine Vision Conference, Cardiff, Wales, pp. 656-665.
4. PETER J. BURT, MEMBER, IEEE, AND EDWARD H. ADELSON, The Laplacian Pyramid as a Compact Image Code. IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. COM-3l, NO. 4, APRIL 1983
5. W.-K. Chen, *Linear Networks and Systems*. Belmont, CA, USA: Wadsworth, 1993, pp. 123–135. (Book)
6. S. P. Bingulac, "On the compatibility of adaptive controllers," in *Proc. 4th Ann. Allerton Conf. Circuits Syst. Theory*, 1994, pp. 8–16. (Conference proceedings)
7. K. Elissa, "An overview of decision theory," unpublished. (Unpublished manuscript)
8. R. Nicole, "The last word on decision theory," *J. Comput. Vis.*, submitted for publication. (Pending publication)
9. C. J. Smith and J. S. Smith, Rocky Mountain Research Laboratories, Boulder, CO, USA, private communication, 1992. (Private communication)

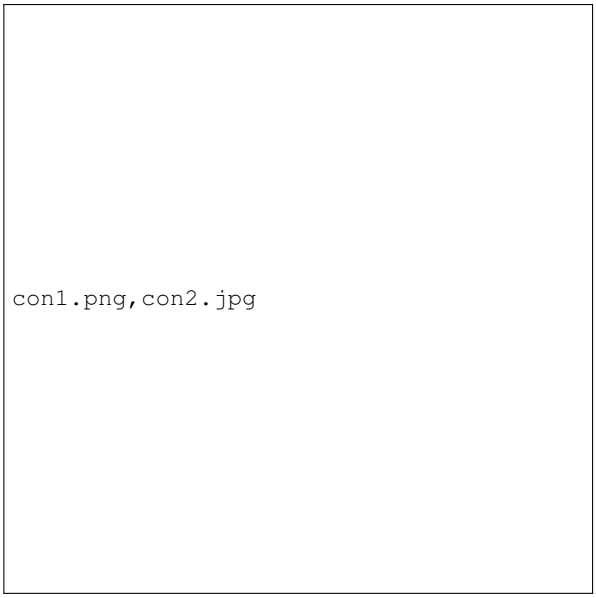

FIGURE 1. Constraints estimation of img3, 4

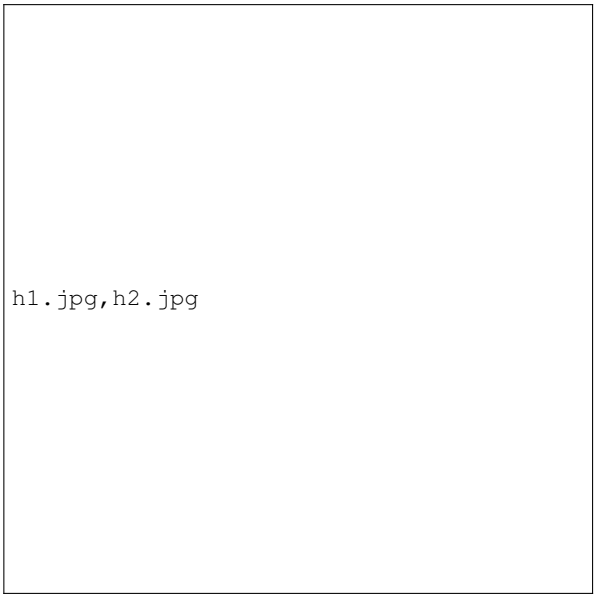

FIGURE 2. H matrix of img3