Performance of SIFT by using different Dimensions of Images

Sawinder Singh Mangat
P.G Student
Department of Electronics and Communication
Chandigarh University, Gharuan, Mohali

Harpreet Kaur
Assistant Professor
Department of Electronics and Communication
Chandigarh University, Gharuan, Mohali

ABSTRACT
In this paper, SIFT algorithm is discussed for feature extraction in image processing Accuracy of feature extraction depends on accurate feature detection and matching. In this research, practical SIFT on different images of different dimensions and compare consequences of that images according to time taken for results.

Keywords
Digital image, multidimensional images, SIFT

1. INTRODUCTION
Filter is exceptionally normal and for the most part utilized calculation for highlight extraction as a part of picture preparing. Filter is fundamentally utilized for Keypoint extraction. [1] These separated keypoints can be utilized for coordinating reason as a part of duplicate move picture phony. Lowe exhibited SIFT for extricating particular invariant components from pictures that can be scrabbled, pivoted and any loud picture. Filter is superior to anything different calculations as far as distinguishing more number of elements from a picture. Filter is first powerful Keypoint-based calculation which is utilized with various sorts of bends. SURF (Speed Up Robust Feature) is propelled adaptation of SIFT. SURF expends less time than SIFT yet SIFT give us more precise keypoints. In SURF, hessian grid is utilized for highlight identification. Both SIFT and SURF are valuable for loud and scaled pictures. [2]

1.1 SIFT Algorithm basically have Four Stages
1. Extrema detection with scale space
2. Localization of keypoints
3. Orientation assignment
4. Description generator

1.1.1 Extrema Detection with Scale Space
Distinguish area and Keypoint utilizing scale space extrema as a part of DOG (Difference-Of-Gaussian). Scale space is particular as the assignment-

$$v(x, y, \sigma) = p(x, y, k\sigma) * S(x, y)$$

Where,
v = Gaussian function
K = Constant factor
*= Convolution
S(x, y) = input image
P(x, y) = variable-scale Gaussian

`

There are different strategies for distinguishing stable keypoints area in scale space. Pooch (Difference-Of-Gaussian is one of them I have utilized here. [4] Gaussian pyramid and DOG pyramid is appeared in fig-1. Scale space extrema equation is given by

$$h(x, y, \sigma) = j(x, y, k\sigma) - j(x, y, \sigma)$$

Where,

$h(x, y, \sigma)$ =scale space extrema

K = scaling value

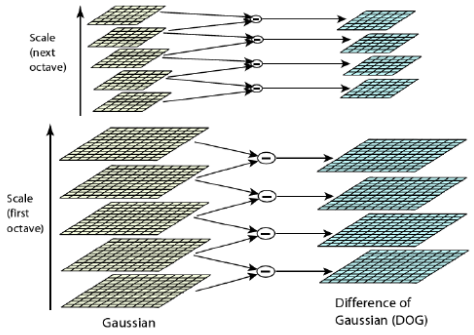


Fig 1: Gaussian pyramid (left side) and Difference-of-Gaussian pyramid (right side) [3]

Get scale space extrema by registering the distinction between two pictures, one with scale (k) with other. To identify nearby maxima and minima of$h(x, y, \sigma)$ each point is compared with its 8 neighbors. If value of $h(x, y, \sigma)$ is minimum or maximum then this point is considered as extrema. [5]

1.1.2 Localization of Keypoints
In this step, many keypoints are removed from slant of keypoints. These eliminations depend on contrast and localization. Only keypoints having poor contrast and having poorly localized on edges. Keypoint localization helps in improving keypoints. Position of extrema is specified by-

$$z = -\frac{\partial^2 D^{-1}}{\partial x^2}\frac{\partial D}{\partial x}$$

On the off chance that estimation of Z is underneath a limit worth when that point is avoided. This worth expels extrema with low complexity. Taylor arrangement extension is utilized to get more exact area of extrema. In the event that power at this extrema is beneath edge esteem then it is rejected. Pooch has high reaction on edges in this way, edges should be evacuated. Harris corner identification is utilized for edge expulsion.

1.1.3 Orientation Assignment
Region around each Keypoint is picked and effects of scaling and rotation is removed. In this step, magnitude and orientation also defined by using these equations.

For magnitude

$$(x,y) = \sqrt{\left(p(x+1,y) - p(x-1,y)\right)^2 + (p(x,y+1) - p(x,y-1))^2}$$

For orientation

$$\theta(x,y) = \tan\left(\frac{\{p(x,y+1)\} - (p\{X,Y-1\})}{p\{x+1,y\} - p\{x-1,y\}}\right)$$

An introduction is allocated to each Keypoint to accomplish constant to picture pivot. An area is taken around the Keypoint area relying upon the scale, and the heading is ascertained in that locale. An introduction histogram with 36 receptacles covering 360 degrees is made. [6]

1.1.4 Keypoint Descriptor

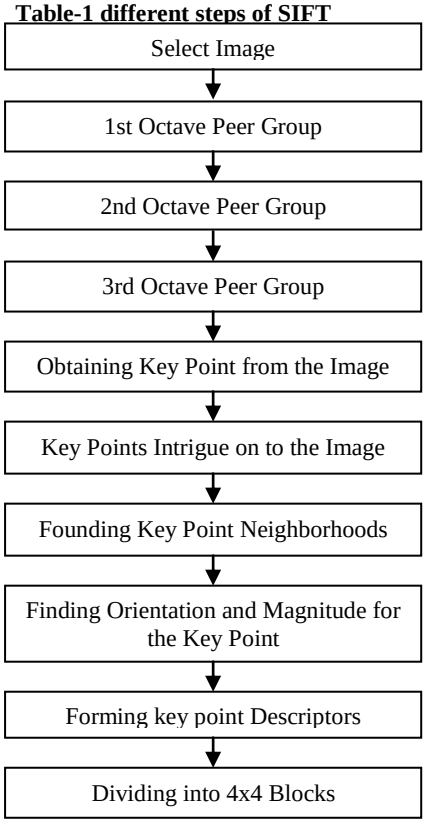
A 16x16 neighborhood nearby the Keypoint is engaged. It is isolated into 16 sub-pieces size of 4x4. In every sub-piece, 8 canister introduction histogram is made. Along these lines, aggregate of 128 receptacle qualities are accessible. It is spoken to as a vector to make Keypoint descriptor. [7]

Keypoints between two pictures are coordinated by looking at their closest neighbors yet now and again, second nearest match might be extremely close to the first. It might have because of clamor or by some other factors.[8] all things considered, the proportion of nearest separation to second nearest separation is taken in the event that it is more noteworthy than 0.8, than they are dismisses generally acknowledged.

2. DIFFERENT STEPS OF SIFT

SIFT is applied on various images with different sizes. Results are tested in PC (i3 processor with 4gb RAM, 2.20Ghz). Different steps of SIFT is shown in table-1;

Table-1 different steps of SIFT



These steps are followed in SIFT for extracting keypoints. Now in following table, fiving information about different images which I have taken for testing.

3. TIME TAKEN FOR RESULTS

Table 2: Different Images

Image name	Dimensions	resolution	size
1	1920*1200	96 dpi	517 kb
2	746*484	96 dpi	126 kb
3	512*512	72 dpi	64 kb
4	256*256	72 dpi	65.8 kb
5	256*256	96 dpi	12.3 kb
6	2560*1600	96 dpi	2.26 mb

In following table, time taken for pyramid level generation is shown which is different according to size and dimensions,

Table 3: Pyramid Level Generation

Image name	Time taken (sec)
1	81.207222
2	12.875532
3	9.908740
4	2.229857
5	2.947473
6	132.737513

In following table, time taken for finding keypoint is shown which is also different according to size and dimensions.

Table 4: Time Taken for Finding Keypoints

Image name	Time taken (sec)
1	55.819794
2	9.483398
3	7.193322
4	1.725977
5	1.986110
6	96.735580

Table 5: Magnitude and Orientaion

Image name	Time taken (sec)
1	67.025976
2	10.772037
3	8.202731
4	2.181993
5	2.410194
6	120.996700

Table 6: Finding Keypoints Description

Image name	Time taken (sec)
1	106.819499
2	12.887945
3	9.715812
4	2.491248
5	2.592090
6	499.752061

4. INPUTS AND OUTPUT IMAGES

These images are of different dimensions. In these output images are shown with keypoints extracted.

Figures (2-7) shows all images with extracted keypoints. There is not keypoint where contrast ratio is less. And no keypoints on edges.[10]

Input image output Image

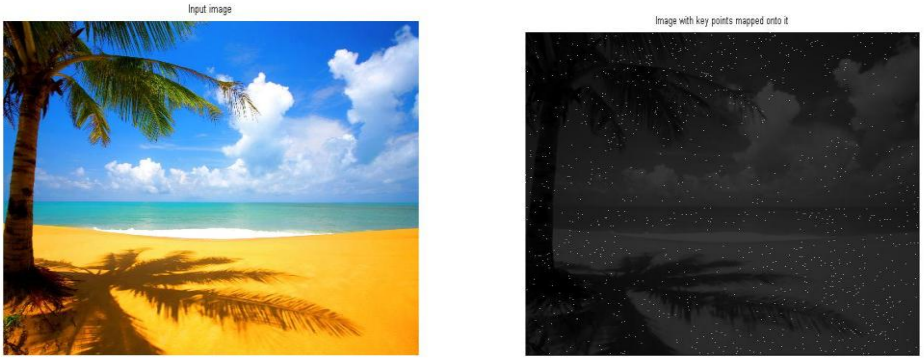


Figure 2: 1st Image with Output

Figure 3: 2nd Image

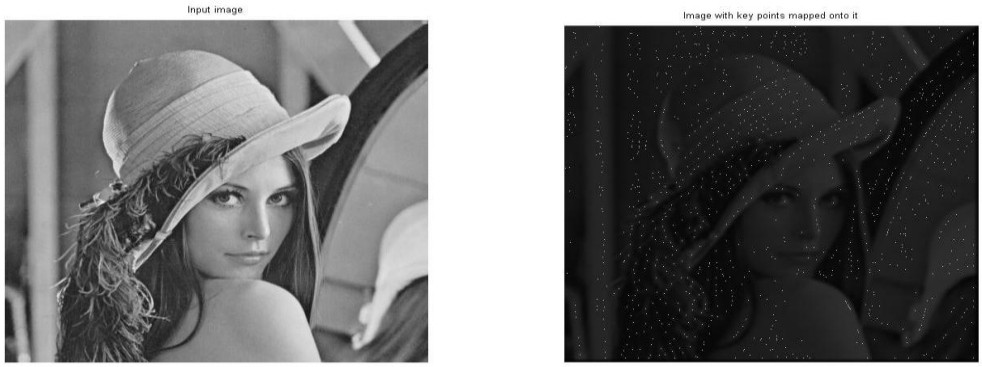


Figure 4: 3rd Image

Figure 5: 4th image

Figure 6: 5th image

Figure 7: 6th image

5. CONCLUSION

In this paper, SIFT connected on various pictures with various measurements. As in this, build measurements of a picture, our time prerequisite for keypoiny extreaction will likewise increment. Filter is better regarding keypoint extraction yet in the event that need results in lesser time than can utilize SURF which is speeded-up form of SIFT. In future scope SIFT can be applied with other method of keypoint extraction.

6. REFERENCES

[1] Luo Weiqi, Jiwu Huang and Guoping Qiu, "*Robust detection of region-duplication forgery in digital image*", 18th International conference on image recognition IEEE, Vol. 4, 2006.

[2] M. Ali Qureshi, M. Deriche, "*A review on copy-move image forgery detection techniques*".

[3] B.I. Shivakumar, Lt. Dr. S.Santhosh Baboo. "*detecting copy-move forgery in digital images: A survey and analysis of current methods",* Global journal of Computer science and technology, vol. 10, pp. 61-65, 2010.

[4] Davide Cozzolino, Giovanni Poggi and Luisa Verdoliva, "*Efficient Dense-field copy-move forgery detection*", IEEE Transaction on information forensics and security, vol. 10, No. 11, november 2015.

[5] P.M. Panchal, S .R. Panchal and S.K, Shah, "*A comparison of SIFT and SURF", International journal of Innovative Research in computer and communication engineering*" Vol. 1, april 2013.

[6] Mohammad Farukh Hashmi, Vijay Anand, Avinas G, Keskar, "*copy-move image forgery detection using efficient and robust method combining Un-decimated Wavelet Transform and Scale invarient Feature Transform*", AASRI conference on circuit and signal processing", Procedia 9, pp.

84-91, 2014.

[7] Weihai Li and Nenghai Yu, "*Rotation robust detection of copy-move forgery*", IEEE 17th International conference on image processing, sept, 2010.

[8] Prerna.C, percy Granaph.J, Angaline.S, Thanga Belsi.I, "*A keypoint based copy-move forgery detection*", international journal of advanced information science and technology, vol. 12, No. 12, April 2013.

[9] Chen-Ming Hsiu, Jen-Chun Lee and Wei-Kuei Chin, "*An efficient detection algorithm for copy-move forgery*", 10th Asia joint conference on Information security, 2015.

[10] Harpreet Kaur, Jyoti Saxena and Sukhjinder singh, "*Key-point based copy-move forgery detection and their Hybrid methods*", Journal of the International Association of Advanced Technology and science, Vol. 16, June 2015.