A Comparison of SIFT and Harris Conner Features for Correspondence Points Matching

My-Ha Le
Graduated School of Electrical Engineering
University of Ulsan, Ulsan, Korea
lemyha@islab.ulsan.ac.kr

Byung-Seok Woo
Graduated School of Electrical Engineering
University of Ulsan, Ulsan, Korea
bswoo@islab.ulsan.ac.kr

Kang-Hyun Jo
Graduated School of Electrical Engineering
University of Ulsan, Ulsan, Korea
acejo@ulsan.ac.kr

Abstract—**This paper presents a comparative study of two competing features for the task of finding correspondence points in consecutive image frames. One is the Harris corner feature and the other is the SIFT feature. First, consecutive frames are extracted from video data then two kinds of features are computed. Second, correspondence points matching will be found. The results of SIFT key points matching and Harris key points will be compare and discussed. The comparison based on the result of implementation video data of outdoor with difference condition of intensity and angle of view.**

Keywords- ***SIFT; Harris conner; correspondence points matching***

I. INTRODUCTION

3D objects reconstruction is one of important process in application of virtual environment, navigation of robots and object recognition. Important progress has been made in the reconstruction of 3D from multiple images and image sequences obtained by cameras during the last few years. In this technology, correspondence points matching are the most important process to find fundamental matrix, essential matrix and camera matrix.

Many kind of features are considered in recent research include Harris, SIFT, PCA-SIFT, SUFT, etc [1], [2]. In this paper, we considered those kinds of features and check the result of comparison. Harris corner features and SIFT are computed then the correspondence points matching will be found. The comparisons of these kinds of features are checked for correct points matching. Once descriptor of key points localization of SIFT were found, we measure the distance of two set points by Euclidian distance. Whereas, Harris corner points were found, matching of correspondence points can be obtained by correlation method. The results of these methods are useful for 3D object reconstruction [3] especially for our research field: building reconstruction in outdoor scene of autonomous mobile robot.

This paper is organized into 6 sections. The next section describes SIFT feature computation, section 3 is Harris corner feature computation. We explain matching problem and discussion in section 4, conclusion is showed in section 5. Paper is finished with acknowledgement in section 6.

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the Human Resources Development Program for Convergence Robot Specialists support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2010-C7000-1001-0007)

II. SIFT

SIFT [4] is first presented by David G Lowe in 1999 and it is completely presented in [5]. As we know on experiments of his proposed algorithm is very invariant and robust for feature matching with scaling, rotation, or affine transformation. According to those conclusions, we utilize SIFT feature points to find correspondent points of two sequence images. The SIFT algorithm are described through these main steps: scale-space extrema detection, **a**ccurate keypoint localization, orientation assignment and keypoint descriptor.

A. Scale space extrema detection

First, we build the pyramid of image by continuous smooth with Gaussian mask. DoG (Difference of Gaussian) pyramid of the image will be obtained by subtraction adjacent smoothed images. By comparing each pixel of current scale with upper and lower scales in the region 3x3, i.e. 26 pixels, we can find the maximum or minimum value among them. These points are also considered as candidate of keypoint. The equations below will be used to describe Gaussian function, scale space and DoG.

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

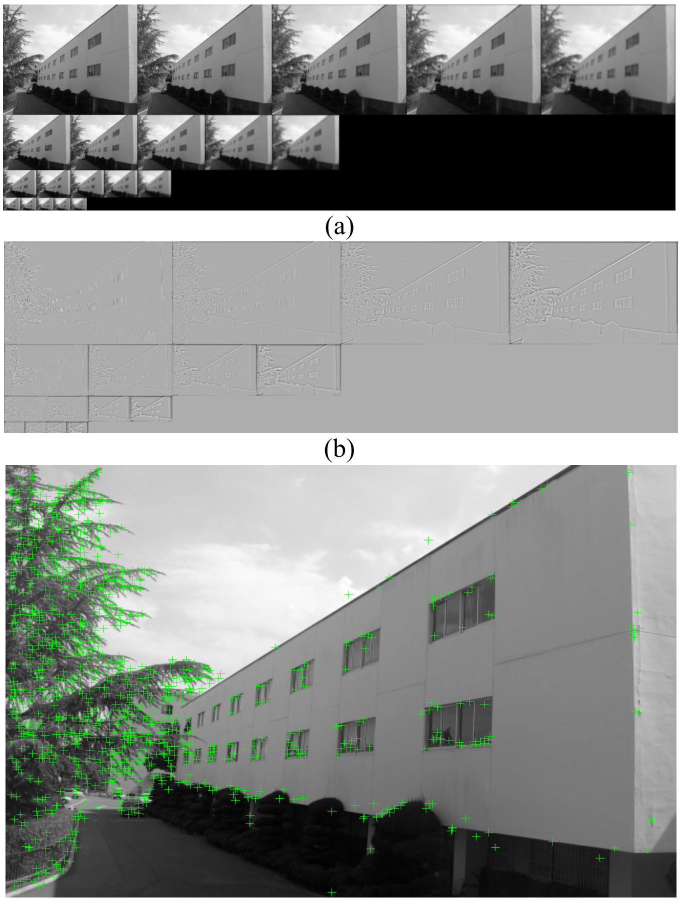
$$L(x,y,\sigma)=G(x,y,\sigma)*I(x,y) \quad (2)$$

Where * is the convolution operation in x and y.

$$D(x,y,\sigma)=(G(x,y,k\sigma)-G(x,y,\sigma)*I(x,y)=L(x,y,k\sigma)-L(x,y,\sigma) \quad (3)$$

B. Accurate keypoint localization

The initial result of this algorithm, he just considers keypoint location is at the central of sample point. However this is not the correct maximum location of keypoint then we



(a)

(b)

(c)

Fig. 1. Keypoint detection. (a) Pyramid images, (b) DoG (Difference of Gaussian), (c) keypoint candidate detection.

need a 3D quadratic function to fit the local sample points to determine the true location, i.e. sub-pixel accuracy level of maximum value. He used the Taylor expansion of the scale space function shifted so the original is at the sample point.

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \quad (4)$$

Where D and its derivatives are evaluated at the sample point and $x = (x, y, \sigma)^T$ is the offset from this point. The location of the extremum, $\hat{x}$, is determined by taking the derivative of this function with respect to **x** and setting it to zero, giving

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} \quad (5)$$

The next stage attempts to eliminate some unstable points from the candidate list of key points by finding those that have low contrast or are poorly localized on an edge. For low contrast point finding, we evaluate $D(\hat{x})$ value with threshold. By substituting two equations above, we have:

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^{-1}}{\partial x} \hat{x} \quad (6)$$

If the value of $D(\hat{x})$ is below a threshold, this point will be excluded.

To eliminate poorly localized extrema we use the fact that in these cases there is a large principle curvature across the edge but a small curvature in the perpendicular direction in the difference of Gaussian function. A 2x2 Hessian matrix, H, computed at the location and scale of the key point is used to find the curvature. With these formulas, the ratio of principle curvature can be checked efficiently.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (7)$$

$$\frac{(D_{xx} + D_{yy})^2}{D_{xx} D_{yy} - (D_{xy})^2} < \frac{(r+1)^2}{r} \quad (8)$$

So if inequality (8) fails, the keypoint is removed from the candidate list.

C. Key points orientation assignment

Each key point is assign with consistent orientation based on local image properties so that the descriptor has a character of rotation invariance. This step can be described by two equations below:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (9)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{(L(x,y+1) - L(x,y-1))}{L(x+1,y) - L(x-1,y))}\right) \quad (10)$$

Two above equation are the gradient magnitude and the orientation of pixel (x, y) at its scale L(x, y). In actual calculation, a gradient histogram is formed from the gradient orientations of sample points within a region around the key point. The orientation histogram has 36 bins covering the 360 degree range of orientations, so each 10^0 represents a direction, so there are 36 directions in all. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with σ that is 1.5 times that of the scale of the keypoint. The highest peak in the histogram is detected and then any other local peaks have 80% of the highest peak value is used to create a keypoint with that orientation as the dominant direction of the key point. One keypoint can have more than one direction.

D. Key point descriptor

The repeatable local 2D coordinate system in the previous assigned image with location, scale and orientation are used to describe image region which is invariant with these

Fig. 2. Keypoint orientation assignment

parameters. In this step, descriptor of keypoint will be compute. Each keypoints will be described by a 16x16 region around. Then in each of the 4x4 subregion, calculate the histograms with 8 orientation bins. After accumulation, the gradient magnitudes of the 4x4 region to the orientation histograms, we can create a seed point; each seed point is an 8-demensional vector, so a descriptor contains 16x16x8 elements in total.

III. HARRIS CONNER FEATURE

The Harris corner detector algorithm relies on a central principle: at a corner, the image intensity will change largely in multiple directions. This can alternatively be formulated by examining the changes of intensity due to shifts in a local window. Around a corner point, the image intensity will change greatly when the window is shifted in an arbitrary direction. Following this intuition and through a clever decomposition, the Harris detector uses the second moment matrix as the basis of its corner decisions. To extract corner can give prominence to the important information. Those can be described by equation below.

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u, y+v) - I(x,y)]^2 \quad (11)$$

Where

- E is the difference between the original and the moved window.
- u is the window's displacement in the x direction
- v is the window's displacement in the y direction
- w(x, y) is the window at position (x, y). This acts like a mask. Ensuring that only the desired window is used.
- I is the intensity of the image at a position (x, y)
- I(x+ u, y+ v) is the intensity of the moved window
- I(x, y) is the intensity of the original

We've looking for windows that produce a large E value. To do that, we need to high values of the terms inside the square brackets. We expand this term using the Taylor series.

$$E(u,v) \approx \sum_{x,y} [I(x,y) + uI_x + vI_y - I(x,y)]^2 \quad (12)$$

We tucked up this equation into matrix form

$$E(u,v) \approx [u \quad v] \left(\sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix} \quad (13)$$

After that rename the summed-matrix, and put it to be M:

$$M = \sum w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (14)$$

Harris corner can be defined as the maximum in local area by the following formula:

$$R = Det(M) - kTrace^2(M) \quad (15)$$

Where

$$Det(M) = \lambda_1 \lambda_2 \quad (16)$$

$$Trace(M) = \lambda_1 + \lambda_2 \quad (17)$$

According to formulas above, all windows that have a score R greater than a certain value are corners.

Fig. 3. Harris corner detection results

IV. CORRESPONDENCE POINTS MATCHING FROM HARRIS AND SIFT FEATURES

A. Correspondence points matching

The corresponding matching point from Harris corner features between previously detected feature points in two images can be found by looking for points that are maximally correlated with each other within windows surrounding each point. Only points that correlate most strongly with each other in both directions are returned. The demonstration result is showed below.

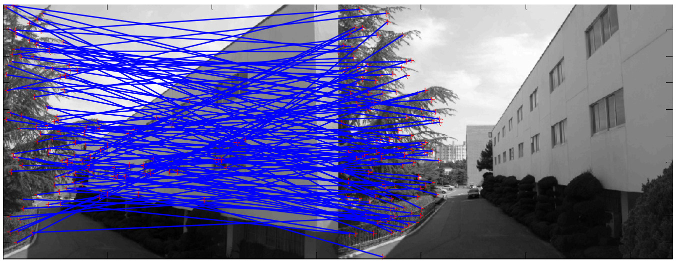


Fig. 4. Matching points result from Harris features

SIFT uses the Euclidean distance between two feature vectors as the similarity criteria of the two key points and uses the nearest neighbor algorithm to match each other. Suppose a feature point is selected in image 1, the nearest neighbor feature point is defined as the key point with minimum metric distance in image 2. By comparing the distance of the closest neighbor to that of the second-closest neighbor we can obtain a more effective method to achieve more correct matches. Given a threshold, if the ratio of the distance between the closest neighbor and the second-closest neighbor is less than the threshold, then we have obtained a correct match.

Fig. 5. Matching points result from SIFT features

B. Comparison

Based on experiment results, we can see that correspondence point from Harris features can be obtained with low time consuming but it is very difficult to get high correct point. Whereas from SIFT features we can get high correctness and robustness correspondence points. Summarizations of these features are showed in the table below.

TABLE I. COMPARED PROPERTIES OF HARRIS AND SIFT FEATURES

Properties	Harris corner features	SIFT features
Time consuming	low	high
Correctness	low	high
Robustness	low	high

V. CONCLUSIONS

This paper was implemented correspondence matching problem from Harris and SIFTS features. Based on the results we can see that SIFT features are robust and correct for that purpose. In future works, we improve matching of two methods by using RANSAC algorithm, calculate fundamental matrix and reconstruct in 3D of outdoor objects.

VI. ACKNOWLEDGMENT

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the Human Resources Development Program for Convergence Robot Specialists support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2010-C7000-1001-0007)

REFERENCES

[1] Luo Juan and Oubong Gwun, "A Comparison of SIFT, PCA-SIFT and SURF" International Journal of Image Processing, Volume 3, Issue 5, 2010

[2] Herbert Bay, Tinne Tuytelaars, Luc Van Gool, "SUFT: speeched up robust features", ECCV, Vol. 3951, pp. 404-417, 2006

[3] R. Hartley, A. Zisserman. "Multiple view geometry in computer vision", Cambridge University Press, 2000.

[4] D. Lowe, "Object recognition from local scale-invariant features". Proc. of the International Conference on Computer Vision, pp. 1150-1157 1999

[5] D. Lowe, "Distinctive Image Features from Scale-Invariant Interest Points", International Journal of Computer Vision, Vol. 60, pp. 91-110, 2004.

[6] C. Harris, M. Stephens, "A combined corner and edge detector", Proceedings of the 4th Alvey Vision Conference, pp. 147-151, Manchester, UK 1998.