

# Comparative Analysis of Detection Algorithms for Corner and Blob Features in Image Processing

Xing Xiong and Byung-Jae Choi

Department of Electronic Engineering, Daegu University, Gyeongsan, Korea

---



---

## Abstract

Feature detection is very important to image processing area. In this paper we compare and analyze some characteristics of image processing algorithms for corner and blob feature detection. We also analyze the simulation results through image matching process. We show that how these algorithms work and how fast they execute. The simulation results are shown for helping us to select an algorithm or several algorithms extracting corner and blob feature.

**Keywords:** Features from accelerated segment test, Laplacian of the Gaussian, Difference of Gaussian, Corner, Blob, Interest point, Image feature

---

## 1. Introduction

In computer vision, a feature can be defined as an interesting part or point of an image, as in [1]. A feature contains some information which apart from its neighborhood. It means that the feature is considered distinctiveness within a certain range. These features could be shape, texture, color, edge, corner and blob etc. Features have high informative contents and appear over and over again in different images. They may be described by the shape, edge or color of objects, or the correlations between two or more images or objects. Until now, many image processing algorithms are based on image feature such as photo stitching, self-localization of robot, objects recognition, object tracking and 3D reconstruction.

In [1], image feature detection refers to methods to compute abstraction of image. Tuytelaars and Mikolajczyk [2] addressed an overview of invariant interest point detectors, how they evolved over time, how they work, and what their respective strengths and weaknesses are. The three robust feature detection methods scale invariant feature transform (SIFT), principal component analysis SIFT and speeded up robust features (SURF) were summarized in [3]. In [4], blob feature detection is defined as one of feature detection. The blob detection is a mathematical method which detects regions or points in a digital image. The regions or points have noticeable difference with their neighborhood. Additionally, a blob is a region or point in which some properties are invariant within a prescribed range of values. Kim and Seo [5] introduced genetic programming based corner detectors for an image processing. In [6-8], some applications for image processing were presented.

In this paper, we present that how some existed corner and blob feature detection algorithms work and how long they execute for an analysis of some different algorithms. And we also discover the application context for each algorithm. The related works are presented in

---

Received: Dec. 5, 2013  
Revised : Dec. 23, 2013  
Accepted: Dec. 24, 2013

Correspondence to: Byung-Jae Choi  
(bjchoi@daegu.ac.kr)  
©The Korean Institute of Intelligent Systems

---

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Section 2. In Section 3, the overview of the corner and blob feature detection algorithms are discussed. Some experiments and simulation results are shown in Section 4. We describe some discussions and future works in Section 5. We here see that Harris corner detection can detect a large number of interest points, but they contain less information for interest point region. Difference of Gaussian (DoG) and determinant of the Hessian (DoH) have similar results in interest point detection, but DoH has faster than that of DoG.

## 2. Some Algorithms for Feature Detection

In 1988, Harris and Stephens [9] proposed a method to detect a corner of an image, which is later called Harris Corner Detector. The method is based on the eigenvalues of the second-moment matrix. The Harris Corner can detect robust corner of an image. However, the algorithm only detects the location of corner, and avoids the interest points which not belong to the corner.

In 1997, a new algorithm called features from accelerated segment test (FAST) was proposed for corner detection. The most contribution of FAST is its computational efficiency. Comparing with other well-known feature detection algorithms such as Harris Detector, smallest univalue segment assimilating nucleus and DoG, it is so faster. So, FAST has been applied in many real time applications such as cleaning robot.

Laplacian of the Gaussian (LoG) is one of the first and also most common blob detectors. The blob feature can be detected from different scale. In comparison with corner algorithms, the scale-space theory is leaded into for detecting interest point of image and a blob contains more information than a corner feature. It means that the point can be described by its neighborhood around it. The feature detection is performed to blob detection from corner detection. In 1999, Lowe used DoG to replace LoG for detecting the interest point with rapid calculation.

Because Hessian matrix has a good performance in accuracy, DoH was also used for detection of the blob feature. In 2006, Bay and Tuytelaars used the scale-normalized DoH from Haar wavelets to detect the interest point. From their experiment results, DoH has similar effect in the interest point detection with DoG, but it is three times faster than that of DoG.

In order to detect image features which are invariant through image affine transformation, Mikolajczyk and Schmid [10] proposed Harris affine detector in. Harris affine detector identifies initial blob using Harris-Laplace detector and then normalizes the blob using affine shape adaption. And the blob is estimated

iteratively. The algorithm can identify similar regions between images. However, the algorithm expends a great of time.

## 3. Blob Feature Detection

For the comparison of several algorithms, we describe the implementation of the detection methodology by step

### 3.1 Harris Corner Detector

Let the image be given by  $I$ ,

$$H(x, y) = \sum_W G(x, y, \sigma) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_{xy} \rangle \\ \langle I_{xy} \rangle & \langle I_y^2 \rangle \end{bmatrix}, \quad (1)$$

where  $H$  stands for a Harris matrix,  $W$  is weighted window,  $I_x$  and  $I_y$  is the partial derivatives of  $I$ , and  $G(x, y, \sigma)$  is Gaussian matrix and expressed in Eq. (5).

For each pixel, Harris and Stephens used Eq. (2) to compute corner response for the computational efficiency,

$$M_c = \det(H) - k * \text{trace}^2(H), \quad (2)$$

where  $k$  is a tunable sensitivity parameter. If the  $M_c$  of the point is greater than the given threshold, the point is considered to be a corner.

Kanade and Tomasi [11] computed directly the  $M_c$  by Eq. (3):

$$M_c = \min(\lambda_1, \lambda_2), \quad (3)$$

where  $\lambda_1$  and  $\lambda_2$  are the eigenvalues of  $H$ .

A large number of experiments have approved that the method is better results because the corner is more stable for tracking.

### 3.2 FAST

The FAST uses a circle of 16 pixels to determine whether the center of circle is a corner, like Figure 1.

- 1) Assume that the intensity of the center of the circle is  $I_p$ . Set a threshold intensity value  $T$ .
- 2) For fast computing,  $I_p$  is first compared to the intensity of pixels  $I_1, I_5, I_9$ , and  $I_{13}$ . If at least three pixels of these four pixels are all brighter than  $I_p + T$  or darker than  $I_p - T$ , the corner will exist. If not, it will be quit.

		16	1	2		
	15				3	
14						4
13			<i>p</i>			5
12						6
	11				7	
		10	9	8		

Figure 1. Set of N contiguous pixels to be determined as a corner (using FAST). *p* is a center of circle. FAST, features from accelerated segment test.

0	1	0
1	-4	1
0	1	0

Figure 2. Discrete approximations to the Laplacian filter.

- 3) If a set of N contiguous pixels out of the 16 needs to be either above or below  $I_p$  by the value  $T$ , then  $I_p$  is a corner.

### 3.3 LoG

The Laplacian is a 2D isotropic measure of the second order derivative of an image. The Laplacian  $L(x, y)$  of the original image  $I$  is given as follows.

$$L(x, y) = L_{xx} + L_{yy} = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (4)$$

Because the original image is represented as a set of discrete pixels, the Laplacian can be approximated by using a discrete convolution kernel as shown in Figure 2.

Because the Laplacian filter is very sensitive to a noise, it is common to smooth image using a Gaussian kernel of Eq. (5) before applying the Laplacian filter.

$$g(x, y, \sigma) = \frac{1}{2\pi\sigma} \exp\left(-\frac{x^2 + y^2}{2\sigma}\right) \quad (5)$$

So, the two-step process is called LoG operation. The overall process for LoG is as follows:

$$L(x, y, \sigma) = g(x, y, \sigma) * I(x, y) \quad (6)$$

In order to detect the blob features which are invariant in different environment, a multi-scale  $\sigma$  is used to filter the original image. A scale-normalized approach is necessary to obtain a multi-scale blob detector with automatic scale selection as the following Eq. (7).

$$\Delta_{norm}^2 LoG = \sigma(L_{xx} + L_{yy}) \quad (7)$$

The points are considered as the interest points (blob feature) which are local maxima or minima of  $\Delta_{norm}^2 LoG$ .

$$(\hat{x}, \hat{y}, \hat{\sigma}) = \arg \max \min local_{(x,y,\sigma)}(\Delta_{norm}^2 LoG) \quad (8)$$

### 3.4 DoG

Similar to LoG, the image  $I$  is first smoothed by convolution with Gaussian kernel of a certain width  $\sigma$ . Then a different Gaussian kernel  $\sigma + \Delta\sigma$  is used to smooth the original image again. The difference of these two Gaussian smoothed images is called DoG, as the following Eq. (9).

$$DoG_{norm}^2 = \frac{\sigma}{\Delta\sigma}(L(x, y, \sigma + \Delta\sigma) - L(x, y, \sigma)) \quad (9)$$

In fact, DoG is a band-pass filter, which can remove high frequency components representing noise and some low frequency components representing the homogeneous areas.

The DoG is one of applicants for edge detection and blob detection in image processing algorithm. The best known application is a keypoint detection proposed by Lowe [12] in 2004.

For fast computing in [12], DoG is defined as an operator or a convolution kernel like the following Eq. (10).

$$\begin{aligned} DoG(x, y) &= L_{\sigma+\Delta\sigma} - L_{\sigma} \\ &= I(x, y) * (g(x, y, \sigma + \Delta\sigma) - g(x, y, \sigma)) \end{aligned} \quad (10)$$

### 3.5 DoH

The Hessian matrix is a square matrix of the second partial derivatives of the function and it could be described by the local curvature of an image as mentioned in [13].

Similar to LoG, DoH considers scale-space to detect the

interest points.

$$\Delta_{norm}^2 DoH(x, y, \sigma) = \sigma^2(L_{xx}L_{yy} - L_{xy}^2) \quad (11)$$

The interest points are selected using the following function to find maxima in scale-space.

$$(\hat{x}, \hat{y}, \hat{\sigma}) = \arg \max_{local(x,y,\sigma)} (\Delta_{norm}^2 DoH) \quad (12)$$

DoH was used in SURF [10], the detection effect of the interest points is similar with SIFT. However, the detection time is three times faster than that of the SIFT through integral image used.

### 3.6 Harris Affine Region Detector

According to affine-invariant theory and multi-scale Harris corner points detection theory, Harris affine detector uses following iterative methods to detect interest points.

- 1) Initialize the search space through Harris-Laplace detector.  $U^{(0)} = Identity Matrix E$
- 2) Normalize ellipsoid region to circle region using the shape adaptation matrix which generated in pervious iteration  $U^{(k-1)}$ , center in  $X_w^{k-1} = U^{(k-1)^{-1}} X^{(k-1)}$
- 3) Select the integration scale,  $\sigma_I^{(k)}$ . The  $\sigma_I^{(k)}$  is determined as the scale that maximizes the LoG.

$$\begin{aligned} \sigma_I^{(k)} &= \arg \max_{\sigma_I = t\sigma_I^{(k-1)}} |LoG(X, \sigma_I)| \\ &= \arg \max_{\sigma_I = t\sigma_I^{(k-1)}} \sigma^2 \det(L_{xx}(X, \sigma_I) + L_{yy}(X, \sigma_I)) \end{aligned} \quad (13)$$

- 4) Select the derivation scale,  $\sigma_D^{(k)}$ . The derivation scale is taken to be related to the integration scale through a constant factor:  $\sigma_D^{(k)} = s\sigma_I^{(k)}$ , where  $s \in [0.5, \dots, 0.75]$ , and

$$\sigma_D^{(k)} = \arg \max_{\sigma_D^{(k)} = s\sigma_I^{(k)}} \frac{\lambda_{\min}(\mu(X_w^{(k)}, \sigma_I^{(k)}, \sigma_D^{(k)}))}{\lambda_{\max}(\mu(X_w^{(k)}, \sigma_I^{(k)}, \sigma_D^{(k)}))} \quad (14)$$

where  $\mu$  is the second moment matrix, and  $\lambda_{\min}$  and  $\lambda_{\max}$  are the Maximum and Minimum eigenvalues of the matrix  $\mu$ , respectively.  $\mu$  is also defined as

$$\begin{aligned} \mu(X_w^{(k)}, \sigma_I^{(k)}, \sigma_D^{(k)}) \\ = \sigma_D^2 g(\sigma_I^{(k)}) \otimes \begin{bmatrix} L_{xx}(X_w^{(k)}, \sigma_D) & L_{xy}(X_w^{(k)}, \sigma_D) \\ L_{xy}(X_w^{(k)}, \sigma_D) & L_{yy}(X_w^{(k)}, \sigma_D) \end{bmatrix} \end{aligned} \quad (15)$$

- 5) Spatial localization: select the point  $X_w^{(k)}$  that computes the Harris corner measure (cornerness) within an 8-point neighborhood around  $X_w^{(k-1)}$ . Find the maximum in the neighborhood,  $X_w^{(k)}$  is as follows:

$$\begin{aligned} X_w^{(k)} &= \arg \max_{X_w \in W(X_w^{(k-1)})} \text{cornerness} \\ &= \arg \max_{X_w \in W(X_w^{(k-1)})} \det(\mu(X_w^{(k-1)}, \sigma_I^{(k)}, \sigma_D^{(k)})) \\ &\quad - \alpha \text{trace}^2(\mu(X_w^{(k-1)}, \sigma_I^{(k)}, \sigma_D^{(k)})), \end{aligned} \quad (16)$$

where the window  $W(X_w^{(k-1)})$  is the set of 8-nearest neighborhood of the pervious point.

Compute the location of interest point  $X^{(k)}$

$$X^{(k)} = X^{(k-1)} + U^{(k-1)} \cdot (X_w^{(k)} - X_w^{(k-1)}) \quad (17)$$

- 6) Update the shape adaptation transformation matrix  $U$ .

$$U^{(k)} = \mu_i^{(k)} \cdot U^{(k-1)} = \prod_k \mu_i^{(k)} \cdot U^{(0)}, \quad (18)$$

where  $\mu_i^{(k)} = \mu^{-\frac{1}{2}}(X_w^{(k)}, \sigma_I^{(k)}, \sigma_D^{(k)})$ .

- 7) Compute the stopping criterion. Sufficiently close implies the following stopping condition.

$$\text{Convergence ratio} = 1 - \frac{\lambda_{\min}(\mu)}{\lambda_{\max}(\mu)} < \varepsilon_c$$

$$\text{Divergence ratio} = \frac{\lambda_{\max}(\mu)}{\lambda_{\min}(\mu)} > \varepsilon_l$$

Mikolajczyk and Schmid had good result with  $\varepsilon_c = 0.05$ .

Meanwhile, a detection algorithm called Harris affine region detector are proposed. The Harris affine detector relies on the combination of corner points detected thorough Harris corner detection, Gaussian scale space, and affine shape adaptation algorithm. And like the Harris affine algorithm, the Hessian affine replaces Harris matrix using Hessian matrix for detecting affine interest points.

## 4. Simulation and Analysis

In this Section, some simulation results of corner and blob detection algorithm will be presented. For comparison, two sets of images are selected for simulation. The images were captured in different view perspectives as shown in Figure 3.

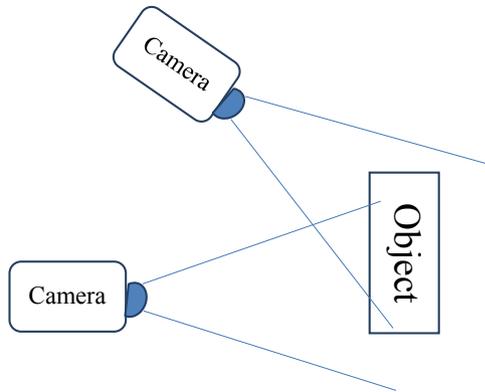


Figure 3. Capture images in different viewpoints.

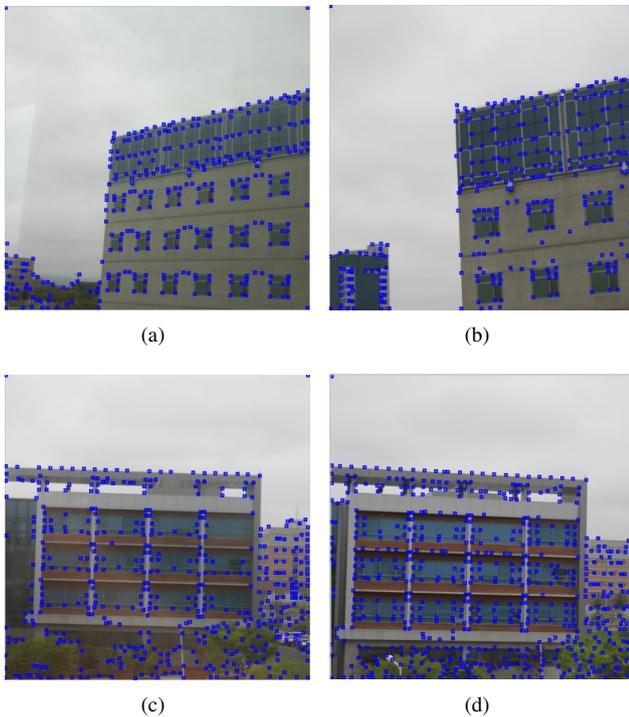


Figure 4. Simulation results of Harris corner detection algorithm.

#### 4.1 Harris Corner Detector

According to the method in Section 3, we use four images for the simulation. The simulation results are shown as followings.

From two set of images in Figure 4, a large number of corners are detected. In the same position of two images, the same corners were detected.

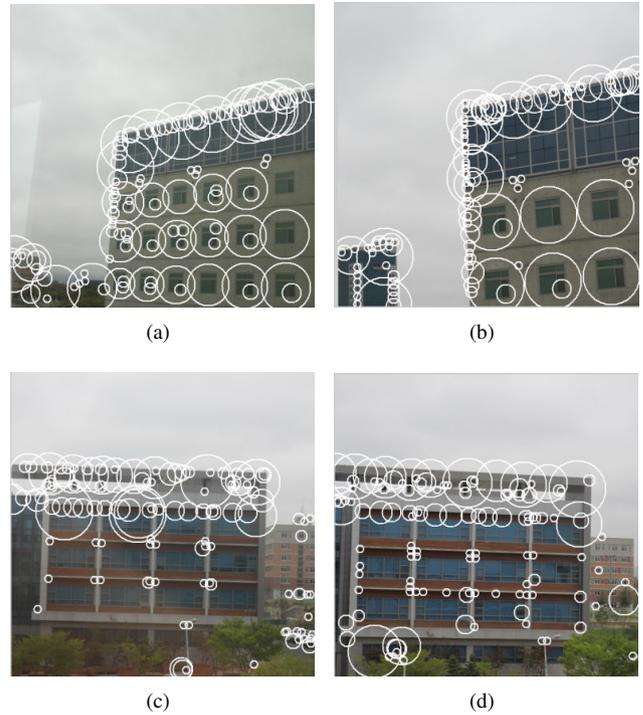


Figure 5. Interest points by Laplacian of the Gaussian detection algorithm.

#### 4.2 LoG

Scale-space has been led to detect interest points. The interest point not only contains the coordinate, but also the scale which describes an effect of interest point to its neighbor.

From Figure 5, the detected interest points are less than that of Harris corner detection algorithm and its computation speed is also slower.

#### 4.3 Harris-Laplace

Because of similar results between LoG and DoG, the scale-space has been used for Harr corner detection, which is called Harris Laplace detection.

From Figure 6, the results have a smaller number of interest points. The computing time is similar to that of LoG.

#### 4.4 DoH

In DoH, integral image has been led to fast detection of interest points. The computing time is also three times faster than that of LoG.

The figure 7 shows interest points detected by DoH algorithm. The great advantage of DoH is computing time. If scale-space

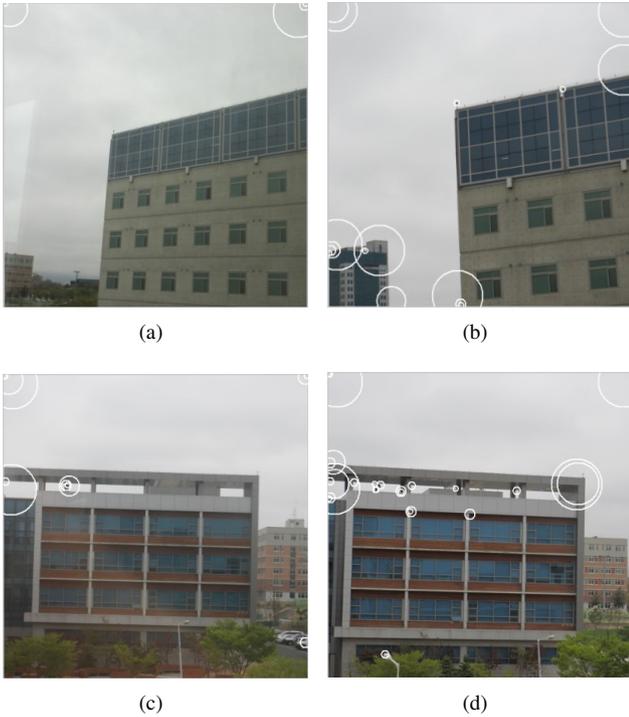


Figure 6. Interest points by Harris-Laplace detection algorithm.

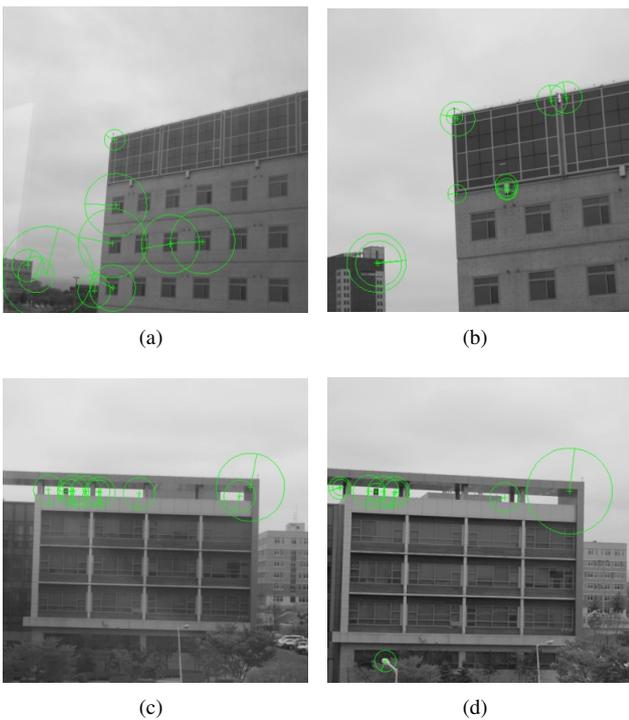


Figure 7. Interest points by DoH detection algorithm.

is not particularly large, DoH can be used for real-time interest points detection.

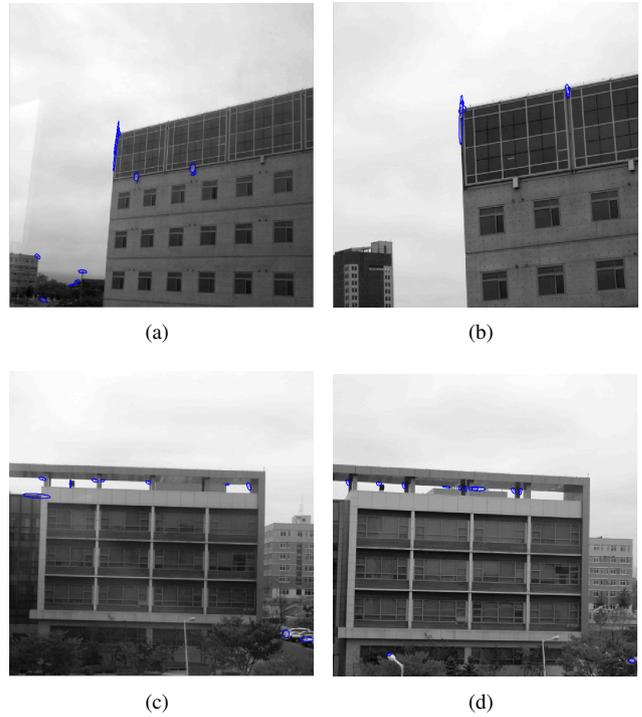


Figure 8. Simulation results of Harris affine detection algorithm.

#### 4.5 Harris Affine Region Detection

For the affine transformation and denser describe object, affine shape adaptation is used. The feature of images is not described by a circle such as those of LoG and DoH, but an ellipse.

From Figure 8, each interest point contains an ellipse region. We could see that the ellipses have similar pixels and similar orientations by the comparison of two sets of images. This provides great convenience for more accurate matching in two images. However, the iterative time for learning is equally vast.

### 5. Conclusion

In this paper, a comparative analysis for some corner and blob detection algorithms has been done. Harris corner detection can detect a large number of interest points and has good performance in interest point's robust and computing time. These points can reflect texture of objects. However, these points contain less information for interest point region. Through scale-space has led to detect interest points of image, the interest points have been replaced by blobs. The blob is easier to reflect the feature of interest point region in different environment change such as illumination, rotation and scale of images. For simulation results in DoG and DoH, both have similar re-

sult in interest point detection, but DoH has faster than that of DoG. Harris affine region detection algorithm added affine shape adaptation for region normalization. The blob detection satisfies image affine transformation. However, the computing time and complexity are much increased by the iteration.

### Conflict of Interest

No potential conflict of interest relevant to this article was reported.

### References

[1] Wikipedia, "Feature detection (computer vision)," Available [http://en.wikipedia.org/wiki/Feature\\_detection\\_\(computer\\_vision\)](http://en.wikipedia.org/wiki/Feature_detection_(computer_vision))

[2] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: a survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177-280, Jun. 2008. <http://dx.doi.org/10.1561/06000000017>

[3] L. Juan and O. Gwon, "A comparison of SIFT, PCA-SIFT and SURF," in *ternational Journal of Image Processing*, vol. 3, no. 4, pp. 143-152, 2009.

[4] Wikipedia, "Blob detection," Available [http://en.wikipedia.org/wiki/Blob\\_detection](http://en.wikipedia.org/wiki/Blob_detection)

[5] Y. K. Kim and K. S. Seo, "Automated generation of corner detectors using genetic programming," *Journal of Korean Institute of Intelligent Systems*, vol. 19, no. 4, Aug. 2009. <http://dx.doi.org/10.5391/JKIS.2009.19.4.580>

[6] C. J. Boo, H. C. Kim, and M. J. Kang, "Image reconstruction of subspace object using electrical resistance tomography," in *ternational Journal of Fuzzy Logic and Intelligent Systems*, vol. 6, no. 1, pp. 47-51, Mar. 2006. <http://dx.doi.org/10.5391/IJFIS.2006.6.1.047>

[7] O. S. Byun and S. R. Moon, "A study on fuzzy wavelet basis function for image interpolation," in *ternational Journal of Fuzzy Logic and Intelligent Systems*, vol. 4, no. 3, pp. 266-270, Dec. 2004. <http://dx.doi.org/10.5391/IJFIS.2004.4.3.266>

[8] X. Xing, B. J. Choi, S. Chae, and M. H. Lee, "Design of a recognizing system for vehicle's license plates with english characters," in *ternational Journal of Fuzzy Logic*

*and Intelligent Systems*, vol. 9, no. 3, pp. 166-171, Sep. 2009. <http://dx.doi.org/10.5391/IJFIS.2009.9.3.166>

[9] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, Manchester, UK, August 31-September 2, 1988, pp. 147-151.

[10] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Computer Vision ECCV 2002, Lecture Notes in Computer Science vol. 2350*, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Heidelberg: Springer Berlin, 2002, pp. 128-142. [http://dx.doi.org/10.1007/3-540-47969-4\\_9](http://dx.doi.org/10.1007/3-540-47969-4_9)

[11] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seattle, WA, June 21-23, 1994, pp. 593-600. <http://dx.doi.org/10.1109/CVPR.1994.323794>

[12] D. Lowe, "Distinctive image features from scale-invariant keypoints," in *ternational Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, Nov. 2004. <http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94>

[13] Wikipedia, "Hessian matrix," Available [http://en.wikipedia.org/wiki/Hessian\\_matrix](http://en.wikipedia.org/wiki/Hessian_matrix)



xiongxing@nate.com

**Xing Xiong** received his B.S. degree in automation engineering at NanJing Institute of Technology, and M.S. and Ph.D. degrees in Electronic Engineering, 2009 and 2013, Daegu university. His research interests include intelligent control and embedded systems. His e-mail address is



**Byung-Jae Choi** received her B.S. in Electronic Engineering, in 1987 from Kyungpook National University, Daegu. And he received his M.S. and Ph.D. degree in Electrical and Electronic Engineering, 1989 and 1998, KAIST in Daejeon. He is a Professor of School of Electronic and Electrical Engineering, Daegu University, Daegu, Korea, since 1999. His current research interests include intelligent control and its applications. His e-mail address is bjchoi@daegu.ac.kr