Comparing Effectiveness of Feature Detectors in Obstacles Detection from a Video

Shaohua Qian
Department of Mechanical &Control Engineering, Kyushu Institute of Technology
Sensuicho 1-1, Tobata, Kitakyushu, 804-8550, Janan
E-mail: qian@ss10.cntl.kyutech.ac.jp

Joo Kooi Tan[1], Hyoungseop Kim[1], Seiji Ishikawa[1], Takashi Morie[2] and Takashi Shinomiya[3]
[1]*Department of Mechanical &Control Engineering, Kyushu Institute of Technology*
Sensuicho 1-1, Tobata, Kitakyushu, 804-8550, Janan
E-mail: {etheltan, ishikawa}@ss10.cntl.kyutech.ac.jp; kim@cntl.kyutech.ac.jp

[2]*School of Brain Science & Engineering, Kyushu Institute of Technology*
Hibikino 2-4, Wakamatsu, Kitakyushu, 804-8550, Japan
E-mail: morie@brain.kyutech.ac.jp

[3]*Department of Business Administration, Japan University of Economics*
Sakuragaokacho24-5, Shibuya, Tokyo, Japan
E-mail: t.shinomiya@aa.cyberhome.ne.jp

Abstract

We have already proposed an obstacles detection method using a video taken by a vehicle-mounted monocular camera. In this method, correct obstacles detection depends on whether we can accurately detect and match feature points. In order to improve the accuracy of obstacles detection, in this paper, we make comparison among four most commonly used feature detectors; Harris, SIFT, SURF and FAST detectors. The experiments are done using our obstacles detection method. The experimental results are compared and discussed, and then we find the most suitable feature point detector for our obstacles detection method.

Keywords: Feature detectors, Harris, SIFT, SURF, FAST, car vision.

1. Introduction

Detection of obstacles in a video sequence is a basic task in autonomous collision avoidance systems of intelligent vehicle. Accurate obstacles detection will improve the performance of obstacles tracking, recognition, classification and motion analysis. We have already proposed an obstacles detection method using a video taken by a vehicle-mounted monocular camera [1]. But this method detects 2D and 3D objects simultaneously. Since these 2D objects are not dangerous to driving, they will reduce the accuracy of detection, if they are detected as obstacles. In order not to detect these 2D objects, we also have proposed a method for classifying 2D objects and 3D objects [2]. In this 2D and 3D objects classification method, correct classification depends on whether we can accurately estimate the camera motion parameters: Using the camera parameters, the method calculates the coordinates of 3D points in the world coordinate system using two corresponding feature points in two consecutive images. The first step of the camera motion estimation is corresponding feature points detection in two consecutive images: It involves the feature points detection and matching.

Feature points detection and matching is the task of establishing correspondences between two images of the same scene. This is an important problem in computer

Published by Atlantis Press

vision with applications in object detection, object recognition and structure from motion. While feature points detection and matching has been studied extensively for various applications, our interest is to match two images reliably in real time for camera motion estimation. Accurate feature points detection and matching can improve the accuracy of 2D and 3D objects classification, and improve the accuracy of obstacles detection ultimately.

Currently the most commonly used feature detectors are Harris, SIFT, SURF and FAST detectors. In this paper, we will make comparison among these four different feature detectors. The experiment will be done using the 2D and 3D objects classification method [2]. The experimental results will be compared and discussed, and then we find the most suitable feature point detector for our obstacles detection method.

The structure of this paper is as follows. Four feature detector methods are overviewed in section 2. Experimental results are shown in section 3. Finally the paper is concluded in section 4.

2. Feature Detectors

2.1. Harris

The Harris detector [3] is based on the local auto-correlation function of a signal: The local auto-correlation function measures the local change of the signal with small windows shifted by a small displacement in different directions.

Let $E(u,v)$ be the change of intensity caused by small shift $[u,v]$:

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u,y+v) - I(x,y)]^2 \quad (1)$$

where $w(x,y)$ is a smooth circular window function, for example a Gaussian.

This change can be concisely written as

$$E(u,v) = [u \quad v] M \begin{bmatrix} u \\ v \end{bmatrix} \quad (2)$$

where M is a 2×2 matrix computed from image derivatives:

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (3)$$

Let λ_1, λ_2 be the eigenvalues of matrix M. Then, if λ_1 and λ_2 have large positive values, a corner is found at the position (x, y).

2.2. SIFT

SIFT or Scale Invariant Feature Transform [4] is a feature detector which is invariant to image rotation and scale. SIFT consists of four main steps to find feature.

2.2.1. *Scale-space extrema detection*

This step is to find salient points in an input image. First, the Difference of Gaussian images are created by building an image pyramid of Gaussian-blurred image of the input image with different scales. Then, with these DoG images, the extrema are found by looking for each pixel in eight neighborhood pixels of the current scale image and in nine neighborhood pixels in the neighboring scale images. If the pixel is the minima or maxima, the pixel is a candidate key point.

2.2.2. *Key point localization*

Once a key point candidate has been found in the previous step, the next step is to eliminate the low contrast key points and the key points which have a strong edge response. For low contrast key points, we can eliminate them using the Taylor expansion of the Difference of Gaussian scale space function. For the key points which have a strong edge response, we can eliminate them using the principal curvature. The principal curvature can be computed from the eigenvalues of the Hessian matrix.

2.2.3. *Orientation assignment*

By assigning a dominant orientation to each key point, the key point descriptor can be represented relative to this orientation and therefore it can achieve invariance to image rotation. An orientation histogram is formed from the gradient orientations of sample points within a region around the key point. The peaks in this orientation histogram are the dominant orientations. If there are multiple dominant orientations, new key points are added at the same location and scale as the original key points.

2.2.4. *Descriptor*

This step is to compute a descriptor for each key point. The descriptor is formed from a vector containing the values of all the orientation histogram entries. Usually, for one key point, we use a 4×4 array of orientation histograms with 8 orientation bins in each to describe it. Therefore, the descriptor of each key point is a $4\times4\times8$ = 128 element feature vector.

2.3. SURF

SURF or Speeded Up Robust Features [5] has been developed to speed up the feature detecting process using SIFT as a basic algorithm.

2.3.1. *Detector*

The detector is based on the Hessian matrix. It uses a very basic approximation of the Hessian matrix which relies on integral images to reduce the computation time. Therefore it is called the 'Fast-Hessian' detector.

Given a point $X=(x,y)$ in an image *I*, the Hessian matrix $H=(X,\sigma)$ at *X* with a scale σ is defined as follows;

$$H(X,\sigma)=\begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix} \tag{4}$$

where $L_{xx}(X,\sigma)$ is the convolution of the Gaussian second order derivative with the original image *I* at point *X*, and similarly for $L_{xy}(X,\sigma)$ and $L_{yy}(X,\sigma)$. The Gaussian second order derivative approximation is further approximated using box filters. The approximations are denoted by D_{xx}, D_{xy} and D_{yy}. The approximation determinant of the Hessian matrix is given by

$$\det(H_{approx})=D_{xx}D_{yy}-(0.9D_{xy})^2 \tag{5}$$

The approximated determinant of the Hessian matrix represents the blob response in the image at location *X*. These responses are stored in a blob response map.

A pyramid is created by applying the box filters which have different sizes on the original image.

The localization of the key point is done by finding the local maxima over a $3\times3\times3$ neighborhood. The maxima of the approximated determinant of the Hessian matrix is then interpolated in the scale and image space.

2.3.2. *Descriptor*

The descriptor is based on the neighborhood of the location of the key point. A square region is centered at the key point, and oriented along the dominant orientation. The size of the window is 20. This region is split up in 4×4 sub-regions. For each sub-region, Haar wavelet responses d_x and d_y are computed in horizontal and vertical directions. The wavelet responses are summed up over each sub-region to form a first set of the feature vector. Also added to this vector are the sums of the absolute values of the responses. Therefore, each sub-region has a four-dimensional description vector of the form

$$v=(\sum d_x,\sum d_y,\sum|d_x|,\sum|d_y|) \tag{6}$$

This results in total vector length of $4\times4\times4$ = 64. Finally, the descriptor is converted into a unit vector.

2.4. FAST

FAST or Features from Accelerated Segment Test [6] is built on the SUSAN detector. The most promising advantage of FAST corner detector is its computational efficiency.

2.4.1. *Segment test detector*

FAST corner detector uses a circle of 16 pixels around the corner candidate *p* to classify whether or not a candidate point *p* is actually a corner. Each pixel in the circle is labeled from integer number 1 to 16 clockwise. If a set of *N* contiguous pixels in the circle are all brighter than the intensity of candidate pixel *p* plus a threshold value *t* or all darker than the intensity of candidate pixel *p* minus a threshold value *t*, then *p* is classified as a corner. *N* is chosen to be 12 because it admits a high-speed test which can be used to exclude a very large number of non-corners: The test examines only the four pixels at the four compass directions.

2.4.2. *Machine learning of a corner detector*

The ID3 algorithm is used to select the pixels which yield the information about whether or not the candidate pixel is a corner. This is measured by the entropy of the positive and negative corner classification responses based on this pixel. The process is applied recursively on all three subsets and terminates when the entropy of a subset is zero. The decision tree resulting from this partitioning is then converted into C-code, creating a long string of nested if-then-else statements which is compiled and used as a corner detector.

2.4.3. *Non-maximal suppression*

Finally non-maximal suppression is applied on the sum of the absolute difference between the pixels on the contour of the circle and the center pixel.

3. Experimental Results

In this section, we carry out 2D and 3D objects classification, which is described in [2], using these four feature detectors. For the experimentations we use two different image sets. The four feature detectors are all tested with those image sets. The results from these experiments contain the information about the detection time, number of detected feature points and the precision of 2D and 3D classification.

First, we analyze the detection time and the number of detected feature points. Figure 1 shows the detection time and Fig. 2 gives the number of detected feature points. From these two figures, we see the FAST detector can find a lot of feature points, and also it costs the least time. It seems the FAST detector is the best detector in these four detectors.

Next, we evaluate the precision of 2D and 3D objects classification. The result of the evaluation is shown in Fig. 3. From Fig. 3, we see SIFT and SURF are the best detectors for this 2D and 3D objects classification method. So, why can the FAST detector find the most feature points, but get the lowest precision? We can find this reason in Fig. 4. Figure 4 shows the detected feature points. Fig.4 (d) shows the detected feature points using FAST detector. In this result, the detected feature points are concentrated in the upper part of the image: Just a few are located in the road region. But in the 2D and 3D classification method, we need to use the feature points which are located in the road region to estimate the parameters of the road plane. So, if we use the FAST detector, we cannot estimate the equation of the road plane accurately, and this results in the low classification precision.

According to the precision of 2D and 3D classification (shown in Fig. 3), although SIFT and SURF have almost the same precision, according to the detection time (shown in Fig. 1), it is easy to get this conclusion: The SURF detector is the most suitable feature detector for our obstacles detection method.

4. Conclusion

In this paper, we performed comparison among four most commonly used feature detectors; Harris, SIFT, SURF and FAST detectors. The experiments are done using our 2D and 3D objects classification method [2]. The experimental results are compared in different ways, and we find that the SURF detector is the most suitable feature point detector for our 2D and 3D objects classification method from the points of the detection time and the precision of 2D and 3D classification. This will also improve the accuracy of obstacles detection (described in [1]) ultimately.

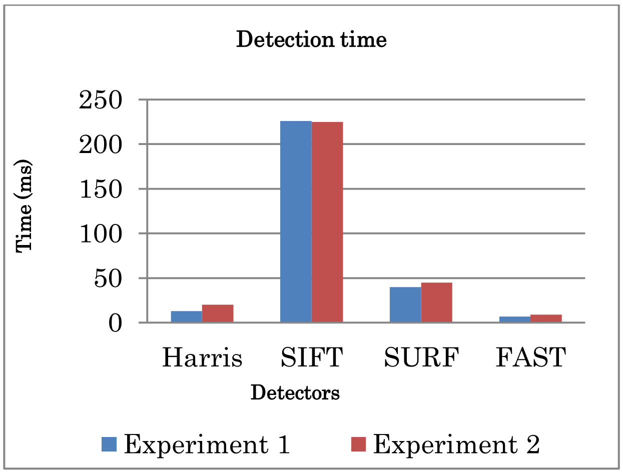


Fig.1. The detection time.

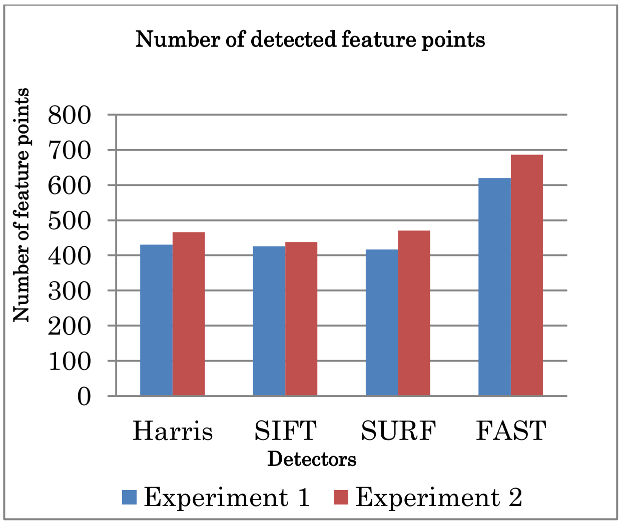


Fig.2. The number of detected feature points.

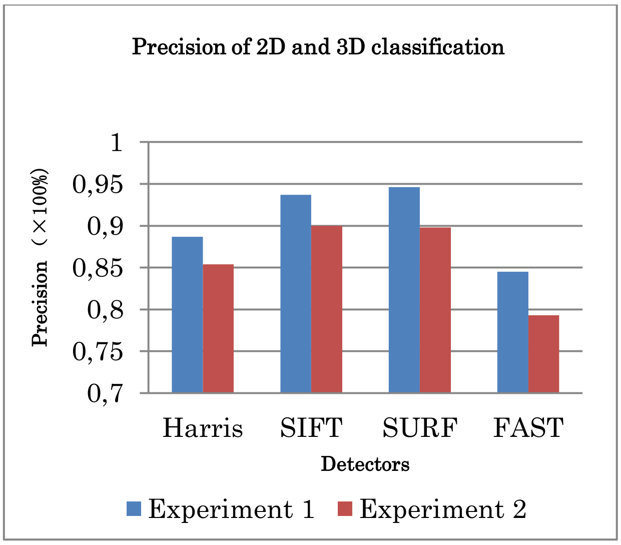
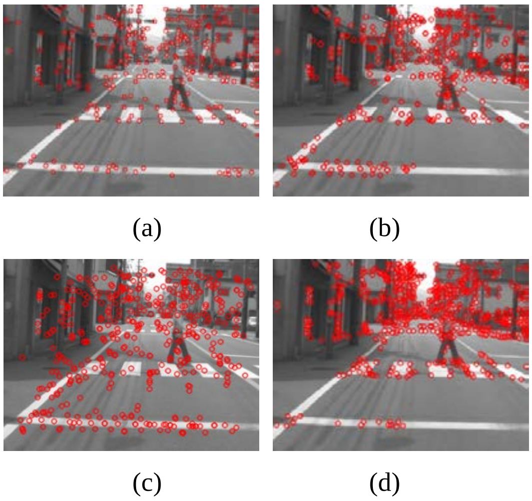


Fig.3. The precision of 2D and 3D classification.


(a) (b)

(c) (d)

Fig.4. The detected feature points. (a) Harris detector, (b) SIFT detector, (c) SURF detector, (d) FAST detector.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 25350477.

References

1. S. Qian, J. K. Tan, H. Kim, S. Ishikawa, T. Morie and T. Shinomiya, Road region estimation and obstacles extraction using a monocular camera. *International Journal of Innovative Computing, Information and Control*, 9(9) (2013), pp. 3561-3572.
2. S. Qian, J. K. Tan, H. Kim, S. Ishikawa, T. Morie and T. Shinomiya, Classifying 2D and 3D objects on a road employing the road plane. in *Proc. SICE Annual Conference* (2013), pp.1689-1692.
3. C. Harris and M. Stephens, A combined corner and edge detector. in *Proc. Alvey Vision Conference* (1988), pp.147-151.
4. D. G. Lowe, Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2) (2004), pp.91-110.
5. H. Bay, T. Tuytelaars and L. V. Gool, SURF: Speeded up robust features. in Proc. *the European Conference on Computer Vision* (2006), pp.404-417.
6. E. Rosten and T. Drummond, Machine learning for high-speed corner detection. in *Proc. the European Conference on Computer Vision* (2006), pp.430-443.