2005 IEEE International Conference on Systems, Man and Cybernetics
Waikoloa, Hawaii October 10-12, 2005

# Fast and Robust Traffic Sign Detection

**Aryuanto Soetedjo**
Information Science and Control Engineering
Nagaoka University of Technology
Nagaoka, Japan
a_aryu@alice.nagaokaut.ac.jp

**Koichi Yamada**
Management and Information Systems Science
Nagaoka University of Technology
Nagaoka, Japan
yamada@kjs.nagaokaut.ac.jp

**Abstract –** *This paper deals with the fast and robust detection of the traffic sign images. A new technique called geometric fragmentation is proposed to detect the red circular traffic signs. It detects the outer ellipses of the signs by combining the left and right fragments of the ellipse objects. A search based on the geometric fragmentation is used to find the ellipse fragments. This search is somewhat similar to genetic algorithm (GA) in the sense that it employs the terms of individual, population, crossover, and objective function usually used in GA. To increase the accuracy and reduce the computational time, a new objective function is introduced for evaluating the individuals. The algorithm was tested for detecting the red circular traffic signs from the real scene image. The experimental results show that the proposed algorithm has a higher detection rate with a lower computational cost compared with the referential genetic algorithm-based ellipse detection.*

**Keywords:** Traffic sign detection, ellipse detection, geometric fragmentation, genetic algorithm, objective function.

## 1 Introduction

The traffic sign recognition system, an intelligent vision system for recognizing the traffic signs, becomes important in an autonomous vehicle. It will provide information of the road signs on the way and guide the vehicle when running in the street. In driver assistance systems, it helps the driver to recognize the road signs early and accurately. A false recognition of the traffic signs caused by the human factors could be avoided. Thus, the traffic sign recognition system makes the driving safer and easier.

In the outdoor environment, due to the illumination changes, rotation, shadows, and partial occlusion of the images, the traffic sign recognition becomes a challenging task. Furthermore, a fast algorithm is needed to operate in the real time environment. Generally, the traffic sign recognition system is divided into two stages: the detection stage, which finds the region of interest containing the traffic signs from an image, and the classification stage where the detected signs are classified into one of the road signs. In this paper, only the detection stage is described.

There are several preceding studies to detect the road signs [1-7]. In [1], the color (RGB color) segmentation and the Laplacian of Gaussian (LoG) edge detector were used to group the red pixels. After that the objects were screened based on the four features (areas, height to width ratios, positions, and detected corners of the objects) to determine which objects might contain the traffic signs. In [2], RGB color was first converted to HSV color and quantized into specific colors. Then, the projection of specific colors onto the horizontal and vertical axes was used to detect the position of the road signs. The template matching technique was used in [3,4,5].

Another approach with genetic algorithm (GA) was used in [6,7]. In [6], a binary image was obtained by the smoothing filter and the Laplacian filter of a grayscale image. GA was then applied to detect the traffic signs. Both the position and the size of a traffic sign were treated as gene information. In [7], first a color analysis using hue and saturation components were done by constructing two look-up tables; one for the hue component and the other for the saturation one. The candidate signs were extracted from the blob image obtained in the first stage using GA. Instead of position and size, horizontal displacement, vertical displacement, horizontal scale, vertical scale, and horizontal rotation were used as gene information.

The shape analysis [1] and projection technique [2] are ineffective when the traffic signs are occluded or distorted. The template matching technique [3,4,5] needs many templates to match the traffic signs, which vary to scale and rotation. The technique using GA might be effective to detect the scaled, rotated, distorted and occluded traffic sign images [7]. Unfortunately, it consumes much computational time.

Here, we propose a fast and robust technique for detecting the red circular sign images. Though only the red circular signs are treated in the paper, the technique could be expanded to detect the other types of traffic signs. The technique is called as *geometric fragmentation*. It finds the circular signs by combining the left and right fragments of ellipses. The left and right fragments of the ellipses are extracted from the edge image obtained by a specific edge extraction method. The proposed technique is fast and robust for detecting the red circular traffic signs from the real scene image.

The paper is organized as follows. Section 2 explains the details of our proposed technique for detecting the red circular traffic sign image. Section 3 presents the results of the experiments. Conclusions are covered in section 4.

# 2 Detection of traffic signs

## 2.1 Outline of detection

The proposed technique uses both color and shape features to extract traffic signs from a real scene image. In this paper, only circular signs with the red color border (include no entry sign) are treated. Since the circular sign image appears as an ellipse when taken from the oblique direction, the detection becomes a task to find red ellipses. Detecting an ellipse from an image can be understood as the extraction of predefined geometric primitives (ellipses) from geometric data [8]. GA has been used to extract certain types of geometric primitives [6-9], where GA is used to search the minimal subset, the smallest number of points necessary to define a unique instance of primitive from the geometric data.

In general, an ellipse is represented by five parameters. A popular way to represent it is by the shape parameters: the two center coordinates, the lengths of semi major and semi minor axes and the orientation of the ellipse. This representation provides a high accuracy of the detected ellipses. However, it suffers from complexity of the search space, when the parameters are used directly in the detection process [6,7,9]. The other way is representation by the minimal subset, a set of five points on the edge of the ellipse [8]. It has a large merit to reduce the complexity of the search space, while it may have significant displacement due to the quantization errors, which is unavoidable in the digital images. Here, we propose a new technique called geometric fragmentation. This technique adopts the representation by the edge points in order to speed up the execution time. The problem of the accuracy would be covered by the approach to combine the left and right fragments of an ellipse, and by the proposed objective function.

The proposed algorithm starts with a red color thresholding. A simple color thresholding with RGB color space is used. The points $(x,y)$ are classified into red if $R(x,y)/(R(x,y)+G(x,y)+B(x,y))>TR$, where $R(x,y)$, $G(x,y)$, and $B(x,y)$ are the values of red, green and blue of point $(x,y)$ in the RGB color space, respectively. $TR$ is the red threshold. After the red color thresholding, a blob image is obtained. Then, a specific edge point extraction [10] is applied to the blob image. The resulting edge points are separated into left group and right group. From the left (right) group, the left (right) fragments of ellipses are extracted by the proposed search technique similar to GA. Then the original ellipses are obtained by combining the left and right fragments.

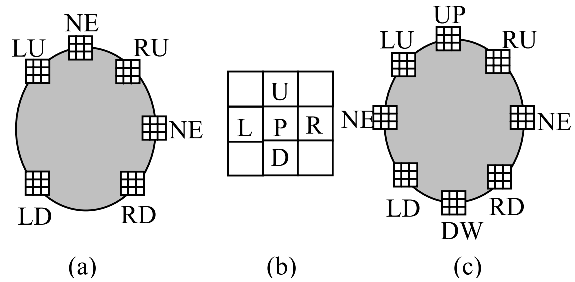


Figure 1. Classification of boundary points on an object: (a) Five classes of boundary points; (b) Four connected neighbors of point *P*; (c) Seven classes of boundary points

## 2.2 Edge points extraction

The common methods to extract edge points of an object are those with Canny, Sobel, Laplacian of Gaussian edge detectors. Those methods extract all edge points of objects. However, we do not have to extract all of them. Only edge points of the ellipse objects must be extracted. Thus the classification of boundary point method [10] is used to extract the edge points from the blob image.

In [10], the boundary points of an ellipse are classified into five classes: *LU, LD, NE, RU* and *RD* as shown in Figure 1.a. In Figure 1.b, let *P* be a boundary point of an object, and *U, D, L*, and *R* be four-connected neighbors of *P*. The five classes of boundary points are defined by the following rules: if both *L* and *U* are not object points, *P* belongs to class *LU*; if both *R* and *U* are not object points, *P* belongs to class *RU*; if both *L* and *D* are not object points, *P* belongs to class *LD*; if both *R* and *D* are not object points, *P* belongs to class *RD;* otherwise, *P* belongs to class *NE*.

In our approach, two classes *UP* and *DW* are introduced as shown in Figure 1.c, using the following rules: if *U* is not an object point, *P* belongs to class *UP*; if *D* is not an object point, *P* belongs to class *DW.* Using this method, only the points that meet the classification criteria are extracted. The noisy points or the points that do not lie on the edges of the ellipse are discarded or reduced.

## 2.3 Ellipses extraction

### 2.3.1 Geometric fragmentation

Before describing our approach, we will first introduce some definitions. Figure 2 gives the illustration. The *left* (*right*) *group* denotes the two classes of edge points *LU* and *LD* (*RU* and *RD*). The ellipses calculated from the minimal subsets in the left (right) group are called *left* (*right*) *ellipses*. The left (right) ellipse usually does not

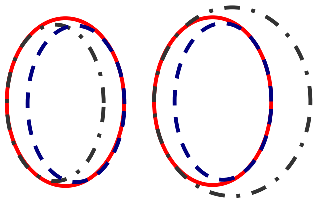


Figure 2. Illustration of the fragments

match the original one completely due to the quantization error in the digital image, especially the arc of the opposite side of the minimal subset has a large error as shown in Figure 2. The *left (right) fragments* are the left (right) part of the original ellipses matched with the left (right) ellipses.

To extract ellipses using GA, a uniform crossover operator, which switches each gene (minimal subset point) according to a uniform probability, is used in [8]. The crossover has the potential to combine the individuals to produce a better ellipse. This should be done evolutionary, thus it needs much computational time.

Our approach improves the speed time in the following three ways: a) instead of searching the minimal subsets of a whole ellipse, we search them of the left and right fragments; b) the original ellipse is extracted by a simple combination of the left and right fragments (unless the left half or the right half of the ellipse is not fully occluded), and c) the left (right) fragment is extracted using the points in the left (right) group.

The geometric fragmentation is also effective in dealing with the occlusion. A typical occlusion of traffic sign images is the one by poles or trees. Thus, in the paper we assume that the occlusion area is less than a half of the circular sign, and is a part of the left half, the right half or the middle half. Then, unless the occlusion is in the whole part of the left half or the right half, the ellipse can be extracted using both the left and right fragments in the same way as the case of no occlusion. In the worst case where the left half or the right half is fully occluded, the fragment of the occluded side is not available. In this case, to increase the accuracy of the extracted ellipse, the edge points in the classes *UP* and *DW* are used in addition to the available fragment. It is easy to find the edge points in the classes *UP* and *DW,* which belong to the same ellipse as the available fragment, because the spatial relationship between those points and the points composing the fragment can be utilized.

### 2.3.2 Search to find fragments

In order to find left (right) fragments from the points in the left (right) group, the search mentioned in this section is used. It is somewhat similar to GA, and employs the tools used in GA such as individuals, population, crossover, etc. The left (right) fragments are found by extracting the left (right) ellipses. Here, just for convenience, an individual of the left (right) fragment is represented by a set of six points instead of five points. In the left (right) group, the six points are taken from the class *LU* (*RU*) and *LD* (*RD*), each three points. The individuals are expressed as follows:

$$\begin{aligned} SV\_left &= \{p_1^{LU}, p_2^{LU}, p_3^{LU}, p_1^{LD}, p_2^{LD}, p_3^{LD}\}, \\ SV\_right &= \{p_1^{RU}, p_2^{RU}, p_3^{RU}, p_1^{RD}, p_2^{RD}, p_3^{RD}\}, \end{aligned} \tag{1}$$

where $p_n^C$ is the $n^{th}$-point taken from the class-*C*. Those three points (hereafter, it is called a sub-individual) are chosen so that they are close together. This is because the chance is high that those three points are the edge points of an ellipse.

The search process described in the following is used to find right fragments. The procedure to find left fragments is done in the same way. It starts with creating the initial population. This task is very important. If the initial population does not contain the points on the edge of an ellipse, the ellipse will not be detected.

To create the individuals in the initial population, first the sub-individuals of all possible ellipses in the classes *RU* and *RD* are extracted from the edge image. Then, pairs of sub-individuals are chosen from *RU* and *RD* to compose individuals so that their horizontal distance is lower than a threshold, and the vertical position of the sub-individual in the class *RD* is beneath the one in the class *RU*. Using this method, the good individuals would be created in the initial population, and all the sub-individuals that compose the original ellipses would be contained in the initial population.

Since the chance is high that the sub-individual contains the edge points of an ellipse, the process to find the minimal subset could be considered as the finding a pair of the sub-individuals lying on the ellipse. The search can be done using a single point crossover with a fixed position. In this case, the individuals do not need be evolved along iteration differently from the normal GA. Therefore the new individuals obtained by the crossover operation are not inserted into the next population. Thus the population members are unchanged throughout iteration. In every iteration, new individuals with objective values greater than a threshold are added to the candidate list. To provide an efficient search, the individuals that are added to the candidate list are not evaluated in the next iteration. Concerning with the multiple ellipses detection, our approach adopts the candidate list as used in [9]. It could detect the multiple ellipses by a single run.

The search process is described in the following:
Step 1. Generate the initial population.

Step 2. Clear the candidate list.
Step 3. Select randomly, pairs of the two individuals, which are not listed in the candidate list.
Step 4. Mate the two individuals using a single point crossover with a fixed position.
Step 5. Evaluate the objective values of the new individuals.
Step 6. If the objective values of the new individuals are greater than a threshold, then add them into the candidate list. If two members in the candidate list are too close, retain the better one.
Step 7. Go to step 3 until N-iteration.

Compared with the GA based-ellipse detection [6-9], our approach is more effective to find the ellipses. First, the initial population could be created effectively using the spatial relationship of the classes. Second, since an individual is composed of two sub-individuals and the good individual is found by changeover of the sub-individuals, it could be considered that an individual is represented by two genes rather than five (six) genes. It is obvious that a good combination of the two genes is easier to be found than the one of the five genes. Third is the problem of trapping in the local maximum, which sometimes occurs in the GA could be avoided. Since the population members are maintained the same throughout iteration, such problem of the local maxima never occurs.

## 2.4 Objective function

In [8,9], the objective function is computed by counting the number of points lying on the candidate ellipse. In [7], the partial Hausdorff distance between the candidate ellipse and the geometric data is used to compute the objective function. The distance indicates how two shapes differ between them. Since the objective function is computed for all individuals in each generation, it contributes to the major computational costs [9]. Another problem of the evaluation of the objective function is the accuracy. Using those objective functions, if a small error tolerance (the distance between points in the edge image and points in the template or candidate ellipse) is used, sometimes the ellipse cannot be detected. If the error tolerance is large, the accuracy will be lower, especially when there are many scattered edge points.

In order to cope with the drawbacks mentioned above, a new objective function is proposed. This objective function is used to evaluate the individuals of the fragments. Figure 3 illustrates the calculation of the objective function for the right fragment. Both the edge image and the blob image are used. To evaluate an individual, first an ellipse is extracted from the six points using a least-square ellipse-fitting algorithm. The extracted ellipse has five parameters: *h, k* (center coordinate), *a* (semi major axis), *b* (semi minor axis), and $\theta$ (ellipse orientation). The points *P* and *Q* are found by introducing a given parameter $\alpha$, which is chosen so that the line *PQ* is in the black part of the blob image. The coordinate of the points *P*(*xp,yp*) and *Q*(*xq,yq*) are given by:

$$\begin{aligned} xp &= h - a\cos(\alpha)\cos(\theta) + b\sin(\alpha)\sin(\theta); \\ yp &= k + a\cos(\alpha)\sin(\theta) + b\sin(\alpha)\cos(\theta); \\ xq &= h + a\cos(\alpha)\cos(\theta) + b\sin(\alpha)\sin(\theta); \\ yq &= k - a\cos(\alpha)\sin(\theta) + b\sin(\alpha)\cos(\theta). \end{aligned} \tag{2}$$

The points *P*' and *Q*' are found by extending the line *PQ* so that the length of the line *PP'* (*QQ'*) is a half of the length of the line *PQ*. Then the objective function is given by:

$$f_{obj} = \frac{\sum_{(x,y)\in PQ} Im(x,y)}{\|PQ\|} - \frac{\sum_{(x,y)\in PP'} Im(x,y)}{\|PP'\|} - \frac{\sum_{(x,y)\in QQ'} Im(x,y)}{\|QQ'\|}, \tag{3}$$

where, *Im*(*x,y*)=1 if the color of the point (*x,y*) in the blob image is black, else *Im*(*x,y*)=0; $\|PQ\|$, $\|PP'\|$, and $\|QQ'\|$ are the lengths of the lines *PQ, PP'*, and *QQ'*, respectively.

The first term of eq. (3) implies that the fragment will have a high value when the fragment has the black points inside it. Thus for a typical blob image of the circular sign as shown in Figure 4.a, the value of the outer fragment is high, while the value of the inner one is almost zero. Therefore only the outer ellipse will be detected.

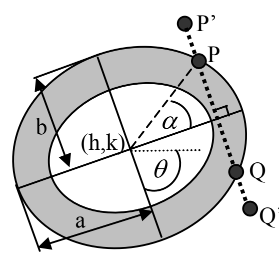


Figure 3. Illustration for calculation the objective function

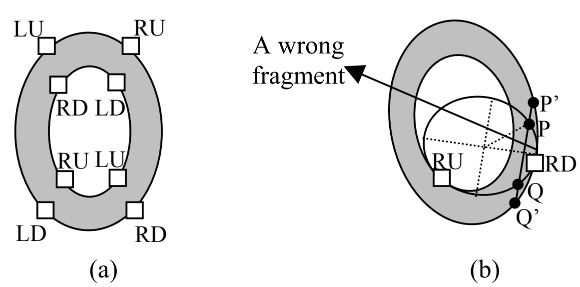


Figure 4. Typical circular sign image: (a) Four classes of the edge points; (b) A wrong fragment

The second and third terms of eq. (3) are used to compensate such problem in the following. In the typical circular sign image, the four classes *LU, LD, RU*, and *RD* of the outer and the inner ellipses are shown in Figure 4.a. Using the method to create the initial population as described before, it is a high chance that only the sub-individuals in the classes *RU* and *RD* of the outer ellipse are created in the initial population. However, in the complex images or when there are two signs, which are close together in the vertical position, the sub-individuals in the classes *RU* or *RD* of the inner ellipse might be created. Then by crossover operation, the individual composed from the sub-individual in the class *RU* of the inner ellipse and the one in the class *RD* of the outer ellipse as shown in Figure 4.b, could be produced. This individual represents a wrong fragment. The points *P, P', Q*, and *Q'* used to calculate the objective function of the individual are shown in the figure. If only the first term of eq. (3) is used, the value will be high. However, since the values of the second and third terms of eq. (3) are also high, the objective value of eq. (3) will be small or negative.

Compared with the previous works, the proposed objective function is more accurate. The problem of the scattered edge points could be overcame by using both the edge image and the blob image. In previous work, the sine function is used to find out all points along the circumference of the ellipse [7,8], or along the first quarter of the ellipse [9]. Here, since the sine function is only used to find out two points *P* and *Q*, and the other points lie in the straight line, the computational time is fast.

### 2.5 Combination

After searching the fragments, we have two candidate lists; one is the list of left fragments and the other is the list of right fragments. The original ellipse is found by combining the two fragments. To combine the two fragments, we use the following method.At the first, a left fragment is combined with a right fragment if they are close together and the position of the left one is on the left side of the right one, vice versa. When multiple ellipses exist in an image, there are several combinations obtained. To validate the proper combination, the combined individual is evaluated using another objective function, which counts the number of points along the circumference of the combined ellipse. Since the number of the combinations is usually small, and the objective function is used for validation only, the drawbacks of this objective function as discussed previously could be avoided.

If there are fragments that do not have their counterparts, a similar processing to the above is done using the points chosen from the classes *UP* and *DW*. These points are searched in the small area determined by the position of the individual of the left (right) fragment.

## 3 Experimental results

The algorithm was implemented using MATLAB and tested on a PC Pentium-4, 2 GHz. To test the execution time and the robustness of the proposed method, 100 real scene images were used. All of them are RGB images with size 240 x 180 pixels. 25 images of them contain the signs, which are partially occluded by poles or trees. The number of red circular signs in an image is varying from one to three. To make a comparison with the existing techniques, we developed a referential algorithm.

In the referential algorithm (RA), two approaches are used to get the edge points from the blob image; one is the Sobel detector and the other is the technique proposed in the paper. Then, the normal GA is used to detect the ellipses. The individual is represented by the five points as used in [8]. The objective function using the partial Hausdorff distance [7] is employed. To cope with the multiples ellipses, the technique using candidate list [9] is used.

To compare the performance of the proposed algorithm (PA) and RA, their execution time and detection rate are evaluated. The execution time is divided into two parts; one to get the edge image and the other to detect the circular signs from the edge image. The detection rate is defined as the ratio of the correctly detected ellipses with the total number of the ellipses in the images. Here, the ellipses refer to the outer ellipses of the signs. We use a population of 100 individuals for all test images in both algorithms. RA uses 100 number of generation, and PA uses 10 number of iteration. The crossover rate for both algorithms is 1. The mutation operator is not used. For each test image, we run each algorithm ten times. Table 1 shows the average detection rate and the average execution time for all test images.

It is shown from Table 1 that PA has a high detection rate for both the single sign and the multiple signs. The execution time of PA is very fast compared with RA. The edge extraction method used in PA could improve the detection rate of RA. The execution time of this edge extraction method is almost the same with the one of the standard edge extraction method.

The typical detection results using RA and PA are shown in Figure 5 and Figure 6, respectively. In the left side of Figure 5, RA fails to detect the outer ellipse of the upper sign. Since the outer and the inner ellipses are close together, it makes the detection difficult. The ellipse extracted from the points in the outer and the inner one will have a relative high objective value when computed using the standard objective function (RA). Contrary, using the proposed objective function, the difficulty could be solved. The left side of Figure 6 shows the detection result using PA, where two outer ellipses could be detected properly.

Table 1. Experimental results

| | Referential algorithm (RA) | | Proposed algorithm (PA) |
|---|---|---|---|
| | Sobel detector | Boundary points | |
| Average edge extraction time | 0.36 s. | 0.31 s. | 0.34 s. |
| Average detection time | 46.82 s. | 46.36 s. | 3.73 s. |
| Average detection rate (single sign) | 40.73 % | 44.79 % | 87.84 % |
| Average detection rate (multiple signs) | 12.31 % | 16.22 % | 86.08 % |


Figure 5. Detection results using RA


Figure 6. Detection results using PA

In the right side of Figure 5 and Figure 6, the signs are occluded by a pole with red color painted on some parts of it. There is a red car on the background. RA could not detect the signs. Meanwhile, PA could detect the two signs successfully.

## 4 Conclusions

The experimental results show that the proposed method works effectively for detecting the circular signs from the real scene images. Compared with the previous works, the proposed algorithm has a better performance, in both the detection rate and the execution time.

The proposed objective function is robust to the noise and very fast to be computed. In the case of the circular traffic signs, where the outer ellipse and the inner ellipse exist, the outer one would have a higher objective value than the inner one. Thus, only the outer ellipse is detected. As a part of the traffic sign recognition system, the detection step is used to provide the border of the traffic sign. Using this scheme where only the outer ellipse is detected, it has two advantages: it makes the ellipse detection more efficient, and the border of the circular sign could be determined directly without the additional step to separate the outer one and the inner one.

The geometric fragmentation technique proposed in this paper, offers the parallel processing scheme, where each group is processed in parallel. Furthermore, many groups could be made without a high computational cost. As many groups are used, the accuracy and robustness would increase.

## References


[1] H.M. Yang, C.L. Liu and S.M. Huang, "Traffic sign recognition in disturbing Environments", Proc. of ISMIS'03, Maebashi City, Japan, 28-31 October 2003.

[2] J.C. Hsien and S.Y. Chen, "Road sign detection and recognition using markov model", The 14th Workshop on OOTA, 2003.

[3] S.-H. Hsu and C.-L. Huang, "Road sign detection and recognition using matching pursuit method", *Image and Vision Computing*, Vol. 19, pp. 119-129, 2001.

[4] D.M. Gavrila, "Traffic sign recognition revisited", Proc. of the 21st DAGM Symposium fur Musterekennung, pp. 86-93, Springer Verlag, 1999.

[5] L. Sekanina and J. Torresen, "Detection of Norwegian speed limit signs", Proc. of the 16th European Simulation Multiconference, Delft, NL, SCS, pp. 337-340, 2002.

[6] Y. Aoyagi and T.Asakura, "A study on traffic sign recognition in scene image using genetic algorithms and neural networks", Proc. of 22nd International Conference on Industrial Electronics, Control and Instrumentation, IEEE, August 1996.

[7] A. de la Escalera, J.M. Armingol and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles", *Image and Vision Computing*, Vol. 21, 2003.

[8] G. Roth and M.D. Levine, "Geometric primitive extraction using genetic algorithm", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 16, 1994.

[9] P.Y. Yin, "A new circle/ellipse detector using genetic algorithms", *Pattern Recognition Letter*, Vol 20, 1999.

[10] H. Chun-Ta and C. Ling-Hwei, "A high speed algorithm for elliptical object detection", *IEEE Transactions on Image Processing*, Vol. 5, No. 3, March 1996.