社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

Multiple Shapes Extraction with Guidance to Random Sampling

Yingdi XIE† Jun OHYA‡

† Student of GITS, Waseda University A201, Nishitomida Okuboyama 1011, Honjo, Saitama, 367-0032 Japan

‡ Faculty of GITS, Waseda University A201, Nishitomida Okuboyama 1011, Honjo, Saitama, 367-0032 Japan

E-mail: † xieyingdi@ruri.waseda.jp, ‡ ohya@waseda.jp

Abstract In this paper, we present a novel method that is capable of detecting multiple shapes, including: straight line, circle, ellipse, triangle, rectangle and other polygons. Multiple shapes' detection is realized via analysis of edge orientation curve, which are obtained by tracing the contours of pre-segmented image region, and recording contour pixels' orientations. A modified histogram is generated from the edge orientation curve, for initial shape guess, by two groups: line/polygon, and ellipse/ellipse arc. Based on the primary shape recognition, target objects are precisely estimated through detecting the corresponding equation's parameters.

Keyword shape detection, a modified ransac, edge orientation curve

1. Introduction

Shape of objects is one of the most important and primitives features for our human being to remember and describe objects. Hence it holds the same fundamentality for machine intelligence. In diverse areas of computer vision, such as medical imaging, scene understanding, intelligent transportation system etc, we may find benefit from detecting shape features

From decades ago, researchers began to make effort on basic shape recognition. Up to now, fruitful methods have been proposed, among which, modules for detecting lines and circles are equipped with many schemes such as [1] and [2], against multifarious undesirable factors, including dense noise, perplexing scene structure etc. As long as the improvement of processor technique, recognition of more complicated shape, such as ellipse is receiving much attention in recent years. Hough Transform family owns the most popularity on shape detection issues. However, methods of Hough Transform will lose flexibility against high dimensional parametric shapes, for the sake of demanding computation load and memory consumption. Its corresponding solutions are proposed, including the following three folds. (1) Detect ellipse by evolved Hough Transform methods [3] [4]. (2) Allay computation cost by decreasing the parameter magnitude [5], or setting geometrical constraint to reduce complexity of parameter space [6]. (3) Improvement of detection accuracy with reasonable speed via analysis of line or arc segments [7].

Single shape estimation is becoming more and more robust and efficient. Instead of dealing with further complicated shapes, our focus is given to multiple shape recognition in one image, because such cases are frequently visible in natural world. As for arbitrary shape estimation, generalized Hough Transform is effective method. However, to detect multiple shapes simultaneously, generalized Hough Transform has to traverse through all the difference parameter spaces for each shape, which will results in unfeasibly difficulty. In this paper, we proposed a method to extract various regular shapes simultaneously based on edge contour analysis. The shapes involved in this paper include: line ellipse, and polygon, which is treated as the combination of line segments.

In the following section, the entire processing framework is described. In section 3, we introduce the definition of edge orientation curve, and primary shape recognition based on the analysis result. Then the line detection from a modified ransac with importance sampling is presented in section 4. Ellipse and elliptical arc's detection is presented in section 5. Section 6 is the experimental result; and the last section gives out the conclusion of this work.

2. System Overview

The shape primitives that are frequently adopted by human being when describing object's appearance, includes: point, line, circle (including half circle), ellipse, polygon (triangle, rectangle, pentagon, and so on), star shape; saw tooth shape, etc. All these shapes can be decomposed or approximated by line and curve segments. In this paper, we call the line and curve segments "fundamental shape elements".

Instead to memorize the individual shapes, a more efficient way is to realize recognition from the fundamental elements. Consider a contour E that a single shaped region inside, through the analysis of "edge orientation curve", which is proposed in this paper, we first classify the edge pixels of E into segments. For each segment, its shape is basically estimated by observing the orientation's distribution. At this step, however, noise or other undesirable factors might impose negative effect toward result, thus that the computer undertakes the risk of missing detection. To solve this problem, we further implement line or certain curve model's estimation via a modified ransac with importance sampling. Afterward of the segmentation and detection of basic shape elements in

E, shape recognition of this contour is carried out by geometrical analysis of elements.

However, in most cases of natural images, not only single object exist, but also other regions. On the other hand, traditional edge detector, such as canny operator, does not provide any grouping guidance for edge pixels. Simply applying these methods might embarrass the basic shape elements by geometrical complexity. In order to keep the accordance with pixel's color intensity, we adopt a mean shift based image segmenter for edge detection. The basic processing steps are as follows:

(1) Initialize centroid as the starting location
(2) Compute new centroid with points in predefined kernel bandwidth both of spatial and range domain.
(3) Shift to the new centroid.
(4) Repeat the above steps until centroid convergence.

After convergence, displacement vector is then utilized to determine whether it is an edge point. The displacement vector is defined as the difference between initial position and final centroid in spatial domain. For each point, the one with greater displacement vector are treated as edge point. The contours for each region are separated from others, thus that we can avoid the complicated geometrical analysis for all the edge pixels in image.

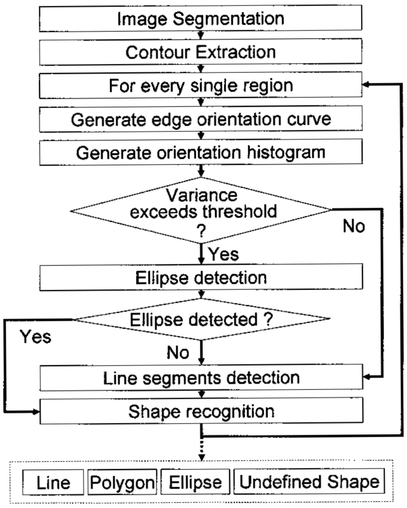
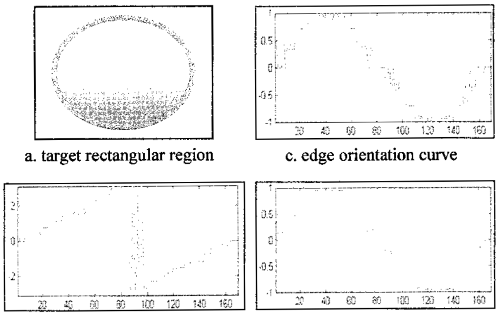


Figure 1, processing block diagram of this proposal

As demonstrated in Fig.1, for an image, it is firstly implemented with mean shift algorithm based image segmentation and edge detection. The edge orientation curve is generated by tracing the edge pixels for each segmented region. Basic shape element is separated through the detection of inflexion in edge orientation curve. In order to strengthen the robustness toward undesirable factors such failure of basic shape segmentation, we apply a modified ransac with importance sampling to retrieve the basic shape elements. Shape recognition for one region is then achieved by analyzing the geometrical relationship of all the detected basic shape elements.

3. Local Feature Analysis

In this section, we introduce an edge orientation curve to shape detection.



a. target rectangular region c. edge orientation curve

b. edge pixel's gradient direction d. smoothed edge orientation curve

Figure 2. Contour tracing and edge orientation curve
In b, the edge orientation curve contains discontinuity around

3.1. Definition

Mean shift based image segmentation provides important sampling constraint when running a modified ransac to compute shape models. However, taking the case of natural images into consideration, multiple basic shape elements may exist in one region, for example, triangle region is composed by three line segments. The line detection with a modified ransac might be lack of efficiency and accuracy, especially under a noisy environment. Therefore, we adopt the local structure: edge orientation curve to segment the contour of single region. The purpose is to cut-off the conjunction between different shapes' borders, which belonging to one region.

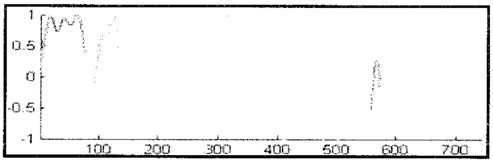
The edge orientation of edge pixel is computed as the gradient direction within a 3-by-3 block, in which the target pixel is centered. We mask out all the other regions. More specifically set all the pixel intensity value of the target region to "1", while the other pixels intensity to "0". Edge gradient computation is based on the binary image with single region. Gradient for a pixel *p* at location *(x,y)* is defined as Eq.3, and its direction is computed by Eq.4.

$$\nabla p = \begin{bmatrix} G_x & G_y \end{bmatrix} = \begin{bmatrix} \partial p / \partial x & \partial p / \partial y \end{bmatrix} \quad (1)$$

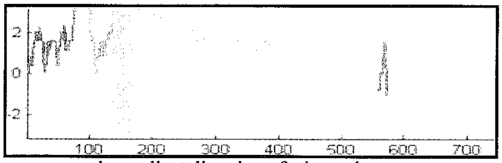
$$\theta_p = \arcsin\left(\frac{G_{p,y}}{\sqrt{G_{p,x}^{\ 2} + G_{p,y}^{\ 2}}} \right) \quad (2)$$

For every region, an edge orientation curve is computed as the gradient for each contour pixel (Fig.2). Tracing contour of the target region in clockwise

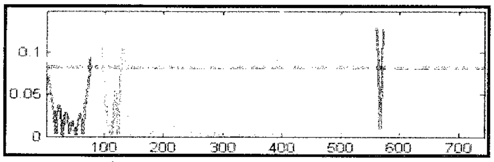
a. contour tracing of complicated region



c. smoothed sine value of gradient direction

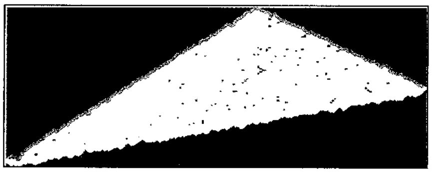


b. gradient direction of edge points

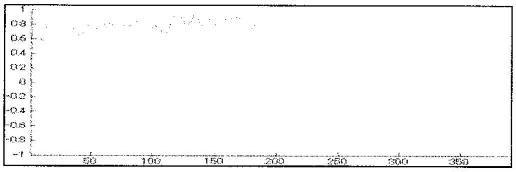


d. EOD Curve of target contour

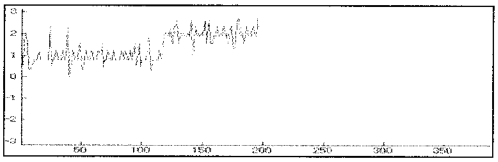
Figure 3. Contour segmentation based on the analysis of Edge Orientation Difference Curve;
Different color demonstrates contour segments. The dashed line in d is the threshold of local peaks of EOD curve.



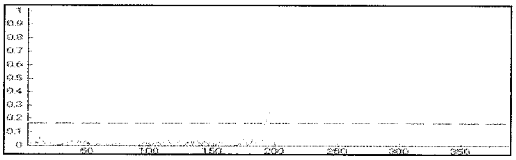
a. contour tracing of a noisy region



c. smoothed sine value of gradient direction



b. gradient direction of edge points



d. EOD Curve of target contour

Figure 4. Edge Orientation Difference Curve of a noisy region,
The threshold in EOD is automatically computed, however with an undesirable value that fails in proper contour segmentation.

direction, a gradient direction curve can be generated. For instance, demonstrated in Fig.2.a and Fig.2.b, a circular region's contour will result in an ascending line if tracing. However, the direction curve lost its consistency around π and $-\pi$, in Fig.2.b horizontally between 80 and 100. In order to keep its continuity, the sine value of gradient direction curve is computed (Fig.2.c),

$$\sin(\theta_p) = \frac{G_{p,y}}{\sqrt{G_{p,x}^2 + G_{p,y}^2}} \tag{3}$$

For the sake of digitization of pixels, the gradient's sine values might contain serrate areas. Since in following steps, great values of the edge orientation difference between adjacent contour pixels might work as a negative impact toward the final result, we smooth the sine value by applying moving average filter (Fig.2.d). The smoothed curve is called as edge orientation curve in this paper.

3.2. Inflexion Detection

In natural images, object region might be immingled with others with similar color intensity, the border pixels of the undesirable region are noise while implementing shape detection. For example, in one single region, multiple shapes are overlapped, and only the border of their mixture can be retrieved from mean shift process. In this case, we may find difficulty in ellipse detection, since the sampling box contains numerous shapes. The major purpose of edge orientation curve is to separate the undesirable shapes within one region by detecting inflexions. It is well known that a line's direction is should be flat in edge orientation curve; and ellipse or circle is a sine wave approximation. The neighbor pixels, which are probably belonging to one shape, are always continuous in edge orientation curve. Consequently, once the difference value is greater than a threshold, *TH*, the pixels on each side of the peak is segmented.

$$\frac{d\sin(\theta_p)}{dp} > th \qquad (4)$$

The threshold to determine the inflexion's location can be achieved through segmenting the peaks in edge orientation difference curve, into two clusters. Then the threshold *th* is the average between the two clusters, as demonstrated as blue dashed line in Fig.3.d. We compared clustering of peaks by the following two ways: (1) Hierarchical clustering; and (2) Clustering by the largest distance among peaks' distance to each other.

As a result, Hierarchical clustering is lack of robustness against scene structures. Clustering by the second method is straight forward, and owns the better performance.

Nevertheless, in the case that target region is only composed by single shape, for example a circle or ellipse, the above contour method compulsively and wrongly segments it into several curves. We set constraints for thresholding as follows. In case of small circle or ellipses, edge orientation curve lost its continuity. Determination of threshold by the above method will result in over-trivial segmentation. Thus when the target region's contour pixel number is small, and it is over-trivial segmented, we treat it as a single segment.

4. Line Segment Detection

Line segments are fundamental elements of polygons. Therefore line segments' detection is performed by the following two folds: (1) geometrical analysis of edge pixels; (2) a modified histogram of edge orientation curve.

4.1. Analysis of Edge Orientation Curve

Pixels of a straight line should have constant orientations in an ideal case; thereby the line's orientation curve forms a flattened area. We first carry out line detection by generating a histogram for the sine values of orientations. However, as described above, the digitized line's orientation curve might contain serrate areas, which severely interfere the histogram generation. Therefore, instead of a regular histogram, we compute its modified version via accumulating the weight of a pixel, namely "modified histogram" (Eq.(5)), demonstrated as Fig.5.b. In which the weight of orientation is calculated through the sine value distance between current pixel and the target sine value over Epanechnikov function (Eq.(6)).

$$hist(u) = \sum_{i=1}^{n} k\left(\left\| \frac{\frac{1}{m}\sum_{j=-m}^{m}\sin(\theta_{i+j}) - \frac{u}{q}}{h} \right\|\right) \qquad (5)$$

$$k(x) = \begin{cases} 1 - x^2 & \|x\| \le 1 \\ 0 & otherwise \end{cases} \qquad (6)$$

where, q in Eq.(5) is a quantization coefficient; h is the bandwidth. (128 and 3 respectively, in our experiment) Peaks of the built modified histogram are treated as the candidate locations of a line. For each peak, if the corresponding continuous region in the orientation curve has the length over a predefined threshold (one thirtieth of the total length of the orientation curve, in our experiment), a line is basically detected. Nevertheless, in a noisy environment straight line might be degraded, accordingly the edge orientation curve diverse drastically. The modified histogram based method tends to fail in line detection. In this case, we adopt a modified ransac process within one region.

4.2. Modified ransac with importance sampling

We modify the original ransac aiming at multiple models' detection. This profile is requested in our application, because, in some cases, the inflexion of line segments with similar edge orientation might suffer from missing of detection, an example is demonstrated in Fig.3. Multiple line segments are wrongly grouped into one cluster. The logical flow of the modified ransac is indicated by the following steps:

(1) Randomly select a minimum data set $\boldsymbol{S_1}$, required to compute a required model from the universal set, denoted as $\boldsymbol{I}$.

(2) Compute a model $\boldsymbol{M_1}$ with set $\boldsymbol{S_1}$. Suppose that the, "proper inlier" and "Quasi-inlier" are the data whose distance to the computed model $\boldsymbol{M_1}$ is less than a predefined threshold $\boldsymbol{TH_1}$ and $\boldsymbol{TH_2}$ respectively, where $\boldsymbol{TH_1 < TH_2}$. Compute the proper inlier and quasi-inlier for all data within $\boldsymbol{I}$. Count their numbers. Let the number of proper inlier as $\boldsymbol{N_{th1,1}}$, quasi-inlier be $\boldsymbol{N_{th2,1}}$ respectively. Notice, both proper inlier and quasi-inlier are treated as the inliers to $\boldsymbol{M_1}$.

(3) If the inlier number is larger than the predefined threshold $\boldsymbol{TH_3}$ for validating the effectiveness of the computed model $\boldsymbol{M_1}$, $\boldsymbol{N_{th1,1}} + \boldsymbol{N_{th2,1}} > \boldsymbol{TH_3}$ label the proper inliers with $\boldsymbol{M_1}$ and "proper" mark. The quasi-inliers are labeled with $\boldsymbol{M_1}$ and "quasi" mark. Update the universal data set $\boldsymbol{I}$ by deleting the proper inliers.

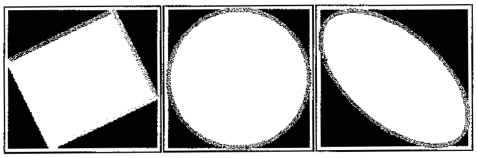
(4) Repeat step 2 and 3 until no valid model that satisfies the inlier number constraint, can be found in a certain times of trial for the universal set $\boldsymbol{I}$, which keeps being updated. For the quasi-inliers, when competence occurs for their assignment to different labeling, treat them as the inliers of the model with smaller distance.

From above description, the modified ransac reserve all the desirable models, while eliminates the outliers. However, random sampling process might result in computation waste, because not any two pixels in one segmentation can construct a suitable model. In order to improve the speed as well as accuracy, we assign weights of importance to ever pixel according to its corresponding value in the modified histogram. Hence, instead of random sampling, the pixels' weight provides importance to guide a modified ransac.

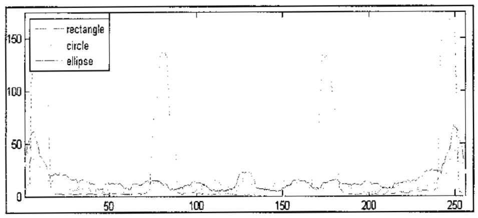
After line segments' detection, the models are optimized via combining similar ones, and input to morphologic analysis. The shape of line and polygon, including triangle, rectangle, pentagon, hexagon etc, is determined when line segments occupy a certain percentage of a region contour, 80% in our experiment. Otherwise, it might contain ellipse arcs or other undefined shapes.

5. Ellipse Model Estimation

Different from the orientation of a line, ideal ellipse's contour is a anamorphic sine wave shaped curve, specifically, sine wave when ellipse's major and minor axes are of the same length (circle). The corresponding modified histogram demonstrates that sine values are widely distributed (Fig.3). Based on this trait, when the variance of edge orientation curve is large, ellipse detection is carried out.



a. rectangle b. circle c. ellipse



d. modified histogram of edge orientation

Figure. 5, modified histogram of rectangle, circle and ellipse shows that circle and ellipse's orientation have greater variance than that of polygon, especially with fewer border number.

5.1. Compute Ellipse' parameter

Standard ellipse equation, Eq.(7), is composed of five unknown parameters: the length of major and minor axes (a and b), ellipse center coordinate (h,k), and orientation angle θ.

$$\frac{((x\cos\theta + y\sin\theta) - h)^2}{a^2} + \frac{((-x\sin\theta + y\cos\theta) - k)^2}{b^2} = 1 \quad (7)$$

Five points without degeneration can build one ellipse model. Similar to line detection, we implement modified ransac for ellipse detection as follows:

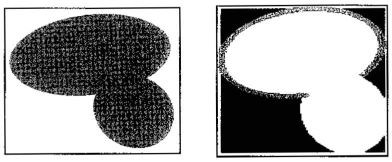
(1) Randomly sample five points from current region contour to build an ellipse.

(2) For edge points within the whole image, weight for current estimation is computed by Eq.(6), against the distance from edge point to ellipse.

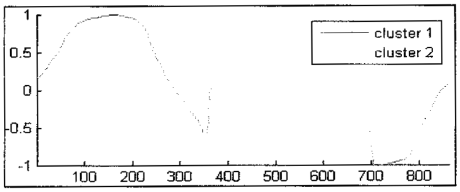
(3) If the summation of each weight exceeds a certain threshold, the estimation is treated as valid, thus that current ellipse model is recorded; and its inlier pixels are masked out. Here, the threshold is set to a certain ratio of estimated ellipse' circumference, 0.6 in our experiment. In order to prevent false estimation by inlier from different region, we assign greater weight to inliers from current region and smaller weight to the ones from other region.

(4) Repeat the above steps until all the pixels in current region are labeled as inliers, or attempts reach a certain number.

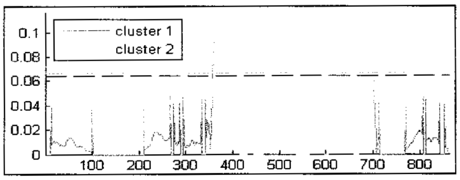
Concerning multiple overlapped ellipses might be detected for the same target region, the estimated ellipse models are optimized through clearing the duplicated labeling of the edge pixel. For one edge pixel, we allow only one label be assigned. Duplicated labeling is cleared in case that the corresponding ellipse' integrity is weak. As long as clearing of duplicated inlier label, inlier number for the corresponding ellipse models decreases. Once the inlier number becomes a value that does not satisfy the predefined inlier threshold, step (3) in modified ransac, the ellipse model is deleted. Otherwise, we will reserve all the detected ellipse models.



a. target region b. clustered contour



c. edge orientation curve



d. edge orientation difference curve

Figure 6. contour clustering for multiple shapes existing region. For the peak in edge orientation difference curve, whose value exceeds the threshold (dashed line in d) is a starter of new cluster.

The above strategy is capable of extracting incomplete ellipse arcs within the same region; however, for the sake of discreteness of digitized pixel, even compute from five pixels belonging to the target ellipse, its parameter is difficult to compute. It is obvious that noise or pixels of different shape objects (such as irregular shape or partially overlapped ellipse which might also have the modified histogram with wide distribution) will impose great negative impact to the ellipse detection, specifically in computation cost and precision.

The straightforward solution is to provide constraint for the random sampling step in modified ransac. We split the contour of single region into segments, thus that the random sampling pool is constrained in one segment. As described above, inflexions are detected when the peaks in edge orientation difference curve exceeds the automatically determined threshold. The area between two adjacent peaks corresponds to a segment in the contour of the region. By finding two adjacent peaks the contour is partitioned into segments. For example, as shown in Fig.6d, the contour in Fig.6b is partitioned into the green and red segments. However, the estimated ellipses might contain undesirable models, such as extreme small or large ellipse. These ellipses are probably reasoned by false detection. We filtered out these bad results by observing the major and minor axes' length.

On the other hand, duplicate ellipse estimation for one region is also possible. Accordingly, some of the edge pixels are labeled with two or more ellipses. Our focus is given to the labeling of edge pixels, since there is no prior knowledge that indicates which one is redundant. For one edge pixel, we remove the labels by comparing the pixel's contribution to the corresponding ellipses. Removing labels will result in decrease of inlier number for some ellipses, so that they might lose the validity and be deleted from final detection result.

For the detected ellipse parameters, when its major and minor axes have similar lengths, it is recognized as a circle.

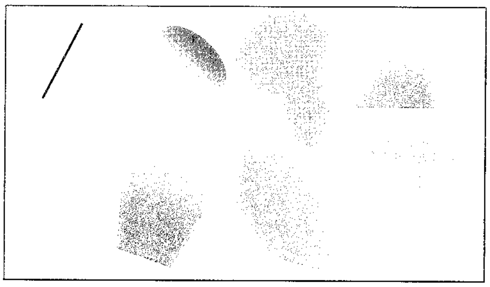
6. Experimental Results

We testify the performance of the proposed method by experiment with synthesized images with line, circle, ellipse, overlapped ellipses, triangle, rectangle, pentagon, hexagon and polygon. As shown in Fig.4 different shapes are successfully recognized by the method described above. Another advantage of this method is reasonable computation speed, which is realized by providence with importance according to pre-clustering of contour pixels before shape model estimation.

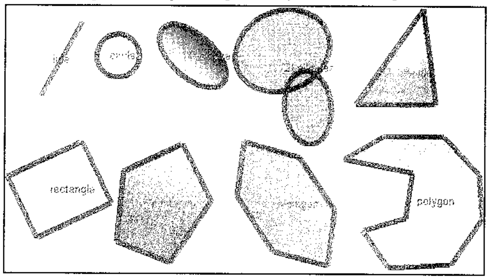
7. Conclusion

In this paper, we proposed a technique to detect multiple shapes within one image, including line segment, polygon, and ellipse. This method realizes shape recognition through a two-step strategy that firstly segmenting image into color region; then from the analysis of edge orientation curve and its corresponding histogram, line composed object or ellipse shaped curve is primarily estimated. To acquire a more precise description, the line and ellipse are estimated by compute parameters of corresponding standard equations from a modified ransac strategy. Demonstrated in the experimental results section, its effectiveness is proved. It has a good trade-off between the computation speed and functionality.

In our proposal, however, the curve shapes are constrained on ellipse, while natural images contain more complicated shapes. Thus, we are going to do further improvement on other shapes' detection.



a. query image composed by various shapes



b. shape detection and recognition result

Figure 7. Experimental results on virtual image which is composed by various shapes

References

[1] Ching Y.-T., "Detecting line segments in an image - a new implementation for Hough Transform," Pattern Recognition Letters, Volume 22, Number 3, March 2001 , pp. 421-429(9)

[2] B. Chazelle, H. Edelsbrunner, "An optimal algorithm for intersecting line segments in the plane," Journal of the ACM (JACM), Volume 39, Issue 1, pp. 1-54, Jan.1992

[3] T.C.Chen and K.L.Chung, "An efficient randomized algorithm for detecting circles," CVGIP : Computer Vision and Image Understanding, Academic Press, 83, 2001, pp. 172-191.

[4] Walsh, D., Raftery, A.E., "Accurate and Efficient Curve Detection in Images: The Importance Sampling Hough Transform," Pattern Recognition Letters, Volume 35, No. 7, July 2002, pp. 1421-1431.

[5] Xie Y. and Ji Q., "A new Efficient Ellipse Detection Method," International Conference on Pattern Recognition, 2002, pp. II: 957--960, 2002

[6] Saburo Tsuji, Fumio Matsumoto, "Detection of Ellipses by a Modified Hough," Transformation. IEEE Trans. Computers, 27(8): 777-781 (1978)

[7] E. Kim, M. Haseyama, H. Kitajima. "Fast and Robust Ellipse Extraction from Complicated Images," Proceedings of the IEEE International Conference on Information Technology and Applications, 2002.