

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

**ANSI***net*

Asian Network for Scientific Information  
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

## Efficient Multi-resolution Detection of Binary Object

Xianyong Fang and Hao Wu

MOE Key Laboratory of Intelligent Computing and Signal Processing,  
School of Computer Science and Technology, Anhui University, Anhui 230039, China

---

**Abstract:** This study presents an efficient multi-resolution method to detect binary object directly. Both intensity and geometry differences are used to measure the similarity between the source object template and target objects inside the test image. Especially for the geometrical difference, the histogram of oriented gradients is proposed to measure the similarity of the edge images between the source object template and the target object. Based on these two measurements, a multi-resolution strategy is used to accurately locate the object position in a coarse-to-fine way, where intensity difference and geometry difference applied consequently in each layer. Experimental results demonstrate the efficiency of the proposed method.

**Key words:** Binary object detection, multi-resolution strategy, intensity similarity, geometry similarity, histogram of oriented gradients

---

### INTRODUCTION

Binary object has at least two merits for object recognition: (1) It is not affected by illumination which may generate different textures, (2) its simple 2-value structure is computationally faster with less workload than other types of popular object imaging method, such as gray image (Bouzenada *et al.*, 2007), or color image (Hsien-Chou and Pao-Tang, 2009). Binary object detection has been widely applied to manufacturing, inspection, target recognition, etc. In this study, we propose a new multi-resolution method to accurately locate the binary object by matching the object template inside the test image in both intensity and geometry.

While there are a lot of studies in object detection and recognition aiming at various types of gray and color images, a few studies are taken specifically for binary object detection. Among them, Murase and Nayar (1995), Krumm (1997) and Hong and Zhang (2001) proposed learning-based methods which compute the object features first by training. But the learning based methods require laborious training process and are not flexible for various types of objects. Ye (2005) proposed another interesting method which attempts to compare the geometry similarity between the template and the test image directly. This method creates a tree structure for the template and the image respectively based on branch points in the contour. It can detect different objects conveniently because it does not need the special training set used in the learning based methods. But the tree structure heavily relies on the branch property of the

object contour. However, in comparison with learning-based methods, we prefer this direct style of method to realize binary object detection. In this paper, we do not use this tree structure, but instead propose a novel direct detection method. It borrows the idea of histogram of oriented gradients from gray and color object detection (Lowe, 2004; Dalal and Triggs, 2005; Watanabe *et al.*, 2009) and applies it to the edge of the object for geometry similarity measure. In addition, the pixel difference inside each object is also adopted as the intensity similarity measure to improve the detection performance. Furthermore, we use a multi-resolution strategy. By creating pyramids for both the template and the test image, this coarse-to-fine strategy eliminates the tedious search process in the fine image through previously locating the potentially similar objects in the coarser test image and thus improves the speed.

### THE BINARY OBJECT DETECTION METHOD

**The pipeline of the method:** Figure 1 shows the detection process. A template image containing the source object  $T_o$  and a test image  $I_o$  are input first. Then the edges of objects in the template and the test image are labeled by the seed fill algorithm as edge images  $T_E$  and  $I_E$ , respectively. After these initialization steps, a multi-resolution detection strategy is used to locate the target object in a coarse-to-fine way.

In this strategy, the test image under different rotation angle is detected with the template so that direction of the target object can be accurately located by

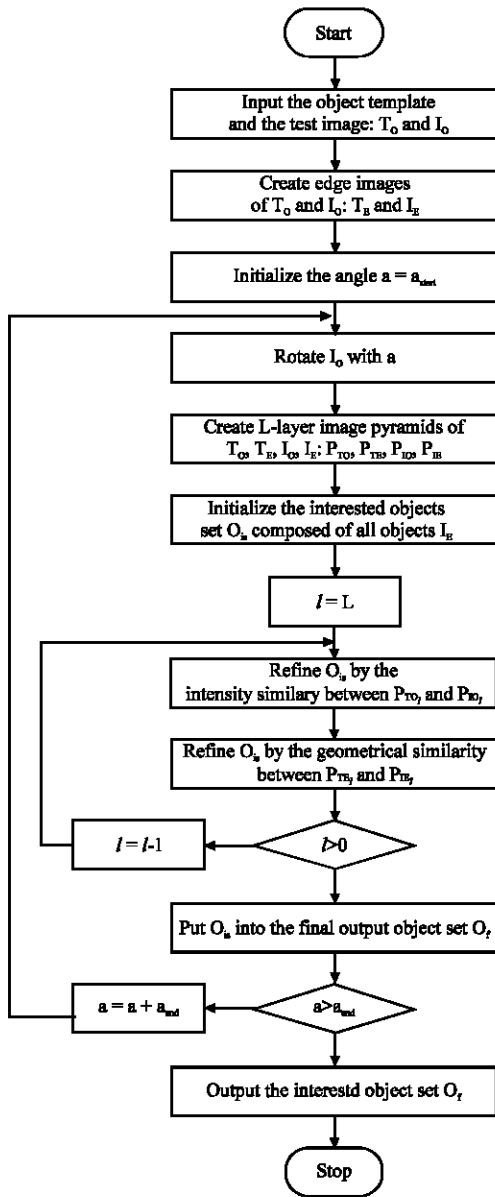


Fig. 1: The binary object detection pipeline of our proposed method

the rotation angle. Therefore, first the rotation angle  $a$  is initialized to the start test angle  $a_{start}$ . Then the progressive multi-resolution detection of the test image under this angle, which will be discussed in the following paragraphs, starts.

First,  $I_o$  is rotated  $a$  degrees. Then the  $L$ -layer image pyramids for  $T_o$  and  $T_e$  of  $T$ ,  $P_{T_o}$  and  $P_{T_e}$ , are created respectively with layer  $L$  as the coarsest layer. The pyramids,  $P_{I_o}$  and  $P_{I_e}$ , for  $I_o$  are also created in the same way. The interested object set  $O_s$  is initialized to contain all objects in  $I_e$  so that dissimilar ones are discarded in the

following steps. Then layer by layer refinement is adopted to find the matched object positions in  $I_o$ . In the current layer  $l$ , first, the intensity similarity of current layer between  $P_{T_o_l}$  and  $P_{I_o_l}$  is checked first to refine the interested objects set  $O_s$ . Some objects that have big intensity difference between  $P_{T_o_l}$  and  $P_{I_o_l}$  are removed after this step. Then, the geometry similarity between  $P_{T_e_l}$  and  $P_{I_e_l}$  for each object in  $O_s$  is checked for the refinement of the interested objects  $O_s$ . After this step, some objects in  $O_s$  will be removed because of their geometrical dissimilarity to the template  $P_{T_e_l}$ .

The above multi-layer refinement repeated until all layers are processed and then  $O_s$  containing all the objects that match with the object under angle  $a$  is obtained. Then,  $O_s$  is put into the final output set  $O_f$  and another round of progressive search with increased rotation angle  $a = a + a_{step}$  is taken again. After all angles has been tested ( $a > a_{end}$ ), the algorithm stops with  $O_f$  containing the final matched objects under different angles.

We now turn to discuss the details of the two core steps in the multi-resolution scheme, the refinement of  $O_s$  by the intensity similarity and the geometry similarity.

### REFINEMENT OF INTERESTED OBJECTS IN EACH LAYER

In this refinement process, the intensity similarity is used first to remove those in  $O_s$  that are dissimilar to the template. Assuming the current layer  $l$  of the template and the test image are  $P_{T_o_l}$  and  $P_{I_o_l}$ , the acceptance of each interested object in  $O_s$ ,  $AcceptI$ , can be formulated as:

$$AcceptI = \begin{cases} \text{accept} & \text{if } Dist_i < a_i \\ \text{reject} & \text{else} \end{cases} \quad (1)$$

$$Dist_i = \frac{\sum_{x,y} (P_{T_o_l}(x,y) - P_{I_o_l}(x,y))}{\sum_{x,y} P_{T_o_l}(x,y)}$$

where,  $a_i$  is the intensity similarity threshold and the intensity similarity  $Dist_i$  is measured as the ratio of the sum of all pixel positions  $(x, y)$  having different values in both images to the sum of pixel values in the template  $P_{T_o_l}$ .

Those objects that rejected by the intensity similarity refinement are removed in  $O_s$  and those left in  $O_s$  are refined again by the geometry similarity which will be discussed in the following.

Assuming the histograms of oriented gradients for the edge images of the template and test image are  $H_{T_e_l}$  and  $H_{I_e_l}$ , respectively, the acceptance of each interested object in  $O_s$ ,  $AcceptG$ , can be formulated as:

$$\text{AcceptG} = \begin{cases} \text{accept} & \text{if } \text{Dist}_g < a_g \\ \text{reject} & \text{else} \end{cases} \quad (2)$$

$$\text{Dist}_g = \sum_{x,y} H_{TE_1}(x,y) - H_{IE_1}(x,y)$$

In Eq. 2,  $a_g$  is the geometry similarity threshold and the geometry similarity  $\text{Dist}_g$  is measured by accumulating the bin differences of histograms of oriented gradients along the edges between  $H_{TE_1}$  and  $H_{IE_1}$

**Outlier removal:** There will be many similar objects detected which are partially overlapped with the best match because both the intensity and geometry similarity are verbosely checked block by block and angle by angle. Calling those that are not the best match to the template as the outliers, we adopt a center based method to remove them, i. e., we first calculate every putative object center and then remove those clustering together except the one with the highest match score to the template. The match score  $S$  is defined as:

$$S = \frac{1}{\text{Dist}_i + \lambda \text{Dist}_g} \quad (3)$$

where,  $\lambda$  is a coefficient that weights the final contribution of the intensity similarity and the geometry similarity.

### EXPERIMENTAL RESULTS

Experiments are undertaken in MATLAB to demonstrate the performance of the proposed method. In our experiments, pyramid is created as the Gaussian pyramid and the search step in layer  $l$  is set to be  $\max(T_{\text{height}}, T_{\text{width}})/100$  where,  $T_{\text{height}}$  and  $T_{\text{width}}$  are the height and width of the template image in level  $l$ , respectively. But in the top layer (coarsest layer), the step is set to be 1 for better detection accuracy. The angle step  $a_{\text{step}}$  is set to be 1. The thresholds of equation 1 and 2,  $a_i$  and  $a_g$ , are set to be 0.2 and 1.2, respectively. We also set the coefficient in equation 3,  $\lambda$ , to be 0.5 to emphasize the importance of geometrical difference since, geometrical difference is about 2 ~ 4 times higher than the intensity difference.

Figure 2 shows the dent detection test in a gear image. Figure 2a shows the gear image and the dent pattern. The gear image is 320×480 but the surrounding

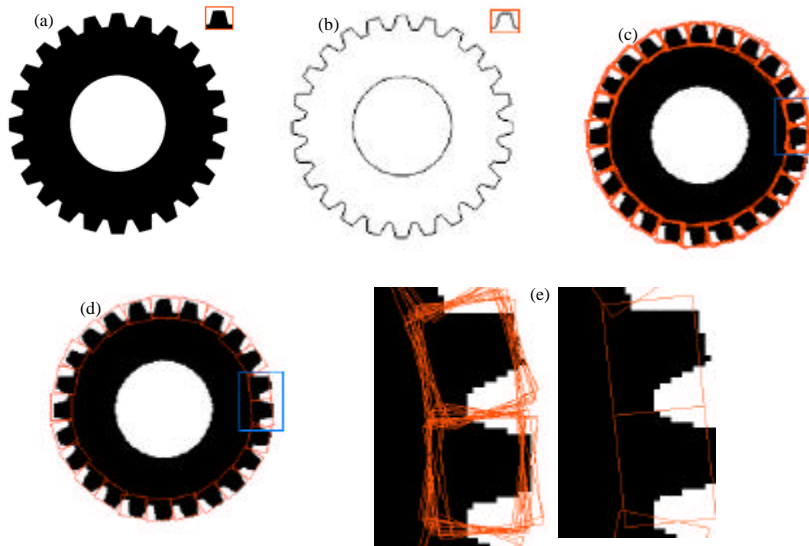


Fig. 2: Detection of a dent pattern in a gear image. (a) The gear image and the dent pattern showed in the red rectangle. The gear image is 320×240 while the pattern is 20×16. The gear image has been clipped out the surrounding blank (white) area for the tight show in this study, (b) the edges of the gear image and the dent pattern showed in the red rectangle. They are detected with seed fill algorithm, (c) the detection result before outlier removal. Each putative object is shown in a red rectangle. About 8 putative objects similar in intensity and geometry around every dent are detected. The details inside the blur rectangle are shown in Fig. 2e for performance comparison, (d) the detection result after outlier removal. Each object is shown in a red rectangle and only the most similar one around each dent is detected in comparison with Fig. 2c. The details in the blur rectangle are shown in Fig. 2e for performance comparison and (e) the performance of the outlier removal. The enlarged sub-images inside the blur rectangles of Fig. 2c and d are shown in the left side and the right side, respectively

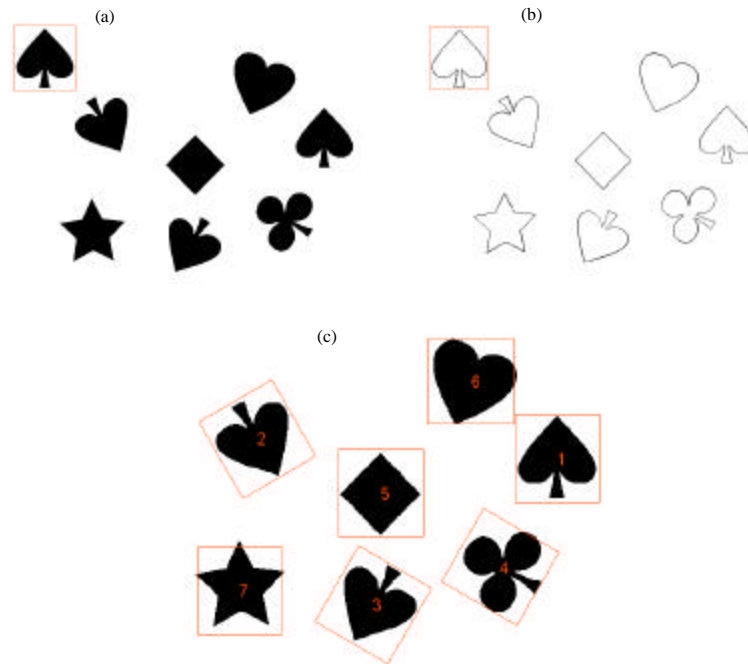


Fig. 3: Detection of a spade pattern in a poker icon image, (a) the poker icon image and the spade pattern showed in the red rectangle. The icon image is  $640 \times 480$  while the pattern is  $107 \times 111$ , (b) the edges of the poker icon image and the spade pattern showed in the red rectangle. They are detected with seed fill algorithm and (c) the detection result after outlier removal. It is shown in a larger size than Fig. 3a and b. Each detected object is shown in a red rectangle. The number in the center of each icon shows the detected order based on the similarity score  $S$  to the spade pattern, where 1 means the most similar one among them

white area is clipped out for the tight show of the image in this study. Figure 2b shows the edges of the image as well as the dent pattern obtained after the seed fill algorithm. The binary images as well as their edge images are taken into the multi-resolution detection process for 360 degree of search. Two-layer pyramids are created for each rotated gear image and the dent pattern respectively because 3 or more layers will make the pattern (only sized  $20 \times 16$ ) too ambiguous. Totally there are 233 putative objects obtained in the layer 1 and then 186 objects after the geometry match. Figure 2c shows these 186 objects obtained before we remove the outliers. Clearly they are heavily clustered. But there is only 24 objects left after applying the outlier removal (Fig. 2d) where all dents are correctly located. Figure 2e gives a close view of the performance of the outlier removal where the two sub-images inside represent the blue rectangles cut out from Fig. 2c and d, respectively. After the outlier removal, the heavy cluster in the left sub-image is removed with only one best match left as showed in the right sub-image.

Our method is also successfully tested with many other binary images. Figure 3 gives one interesting

example among them. Figure 3a shows the test image having poker icons and the spade pattern for detection and Fig. 3b shows the edges of the icon image as well as the pattern obtained after applying the seed fill algorithm. In this example, three-layer pyramid is created for the icon image and its edge image respectively since the pattern is rather large ( $107 \times 111$ ). The angle range is also between 0 and 360 degree. Figure 3c shows the final result obtained after the outlier removal. In this figure, their orders are displayed to show their similarities to the spade pattern. Denoting those icons as their orders, we can have two groups, group 1 containing 1, 2 and 3 and group 2 containing 4, 5, 6, 7. While there are relatively large differences between icons in group 1 and icons in group 2, there are relatively small differences between icons inside each group. The order of icons inside each group is finally decided by their scores  $S$  although visually it is hard for us to judge whether one of them is more similar to the pattern than the others.

Our method can also be applied to gray object detection. In this case, first the test image and the pattern are threshold into binary images. Then our method is

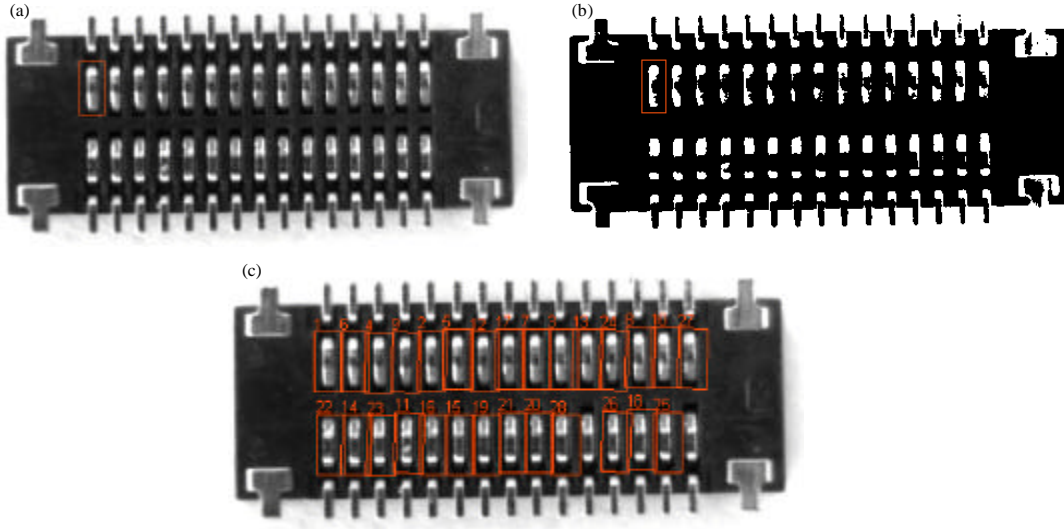


Fig. 4: Semiconductor example, (a) the gray semiconductor image and its clipped pattern which is shown in the red rectangle. This image is  $640 \times 480$  while the pattern is  $24 \times 50$ . The image has been clipped out the surrounding blank (white) area for the tight show in this study, (b) the thresholding result. The red rectangular area is the pattern and (c) the detection result showed in a larger size than Fig. 4a and b. Each object is shown in a red rectangle. The number denotes the order of the similarity as Fig. 3c. We manually display the first 28 similar objects to demonstrate the robust of our method for gray object detection

applied to detect the objects. Figure 4 shows such an example where a semiconductor image is searched for its sub-pattern (Fig. 4a). Figure 4b shows the binary image obtained for our method and Fig. 4c shows the final detection result with the number showing the similarity of each object to the pattern. The first 28 objects that are similar to the pattern are shown manually since, the 29th and after are not very accurately located as we wish. But these first 28 objects have demonstrated the applicability of our method to gray image, where the original pattern is correctly located as the No. 1 object with other similar ones following it closely.

### CONCLUSIONS

This study presents a new multi-resolution binary object detection method. It applies first the intensity similarity and then the geometry similarity into the coarse-to-fine process to accelerate the detection and accurately locate the target objects. Experiments on both binary image and gray image show the efficiency of our method.

Our method has been successfully applied to detect patterns in binary images with the proposed multi-layer pipeline. As demonstrated in Fig. 4, we are also trying to extend the current idea to pattern detection in gray image. But, as Fig. 4 shows, it is sometimes hard to robustly

detect all visually similar patterns in gray image. This is partly because our thresholding step discards much useful texture information in the gray image. In this case, we may introduce more robust cues, such as curvature or surface gradients, into our pipeline to improve performance.

### ACKNOWLEDGMENTS

The study is mainly supported by Hefei Beide Information Technology Co., Ltd. It also receives the co-support from the Key Science Fund for Higher Education of Anhui Province, China (KJ2010A010) and the Key Science Fund for Youth Researchers of Anhui University (2009QN009A).

### REFERENCES

- Bouzenada, M., M.C. Batouche and Z. Telli, 2007. Neural network for object tracking. *Inform. Technol. J.*, 6: 526-533.
- Dalal, N. and B. Triggs, 2005. Histograms of oriented gradients for human detection. *IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognition*, 1: 886-893.
- Hong, L. and Y. Zhang, 2001. Two value image pattern recognizing technology of ANN. *Infrared Laser Eng.*, 30: 432-437.

- Hsien-Chou, L. and C. Pao-Tang, 2009. A novel visual tracking approach incorporating global positioning system in a ubiquitous camera environment. *Inform. Technol. J.*, 8: 465-475.
- Krumm, J., 1997. Object detection with vector quantized binary features. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (CSCCVPR'97), Washington, DC, USA., pp: 179-185.
- Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60: 91-110.
- Murase, H. and S.K. Nayar, 1995. Visual learning and recognition of 3-d objects from appearance. *Int. J. Comput. Vision*, 14: 5-24.
- Watanabe, T., S. Ito and K. Yokoi, 2009. Co-occurrence histograms of oriented gradients for pedestrian detection. *Adv. Image Video Technol.*, 5414: 37-47.
- Ye, Q., 2005. A branch moment invariant extraction algorithm for binary image. *Comput. Eng. Appl.*, 41: 78-80.