Proceedings of the 1991 IEEE
International Conference on Robotics and Automation
Sacramento, California - April 1991

An Automated System for Design-Rule-Based Visual Inspection of Printed Circuit Boards

Seyfullah Halit OĞUZ and Levent ONURAL
Electrical and Electronics Engineering Department
Bilkent University
Bilkent 06533, Ankara, Turkey

Abstract

In this paper, the design and the implementation of an automated, design-rule-based, visual printed circuit board (PCB) inspection system are presented. The developed system employs mathematical morphology based image processing algorithms. This system detects PCB defects related to the conducting structures on PCBs by checking a set of geometric design rules. For this purpose, a new image segmentation algorithm and a new defect detection algorithm are designed. The defect detection algorithm is capable of verifying the minimum conductor spacing, minimum conductor trace width, and the minimum land width requirements on digital binary PCB images. Also, an already existing defect detection algorithm is modified for its implementation in our system. A prototype system is implemented in our image processing laboratory.

1 Introduction

In this work, the design and the implementation of a system which is capable of automatically performing the visual pattern inspection of printed circuit boards are accomplished. The automation of the visual inspection process of PCBs is indispensable in order to maintain and improve the productivity of PCB manufacturing. Due to its various advantages and flexibility in use, a design rule based system is developed. This system makes use of defect detection algorithms which are derived using image transformations based on mathematical morphology. In this study, only the detection of the defects related to the conducting structures on the PCBs is considered.

For the implementation, we began with the initial work for determining and classifying the defects to be detected. The following classification is made for the defects planned to be detected:

1. Defects related to the violation of the minimum land width (MLW) requirement.
2. Defects related to the violation of the minimum conductor spacing (MCS) requirement.
3. Defects related to the violation of the minimum conductor trace width (MCTW) requirement.

In the next step, the design rules which govern the appearance of different conducting structures on PCBs are determined. The design rules mentioned above are described in the form of (geometric) criteria that set such physical constraints as the width and clearance (separation) of conductor traces. In this phase of the work, we assumed the following rules for printed circuit pattern geometry on PCBs:

- Patterns are constructed mainly by conductor traces of arbitrary inclinations and widths.
- A minimum and a maximum limit exist for conductor trace widths and these two limits are common to all conductor traces on the PCB.
- A minimum limit exists for conductor trace separations and this limit is common to all substrate gaps in between conductor traces.
- Undrilled pads of arbitrary shapes are allowed but if holes (either through holes or vias) exist on the PCB, then the lands surrounding these holes should be annular shaped, and, a minimum limit exists for the widths of these annular lands where this limit is common to all lands on the PCB.

Then, we designed appropriate mathematical morphology based algorithms those are capable of verifying the design rules and detecting, locating, and classifying any violations. For this purpose, two new algorithms (an image segmentation algorithm specially designed for PCB images, and a defect detection algorithm capable of detecting the previously mentioned 3 major classes of PCB defects after some proper parameter changes) are developed, and another defect detection algorithm due to [7] is modified for implementation on our image processor. Later, we determined the most time-efficient realizations of the proposed morphological image processing algorithms where the tools (hardware) to be used for implementation and thus

their capabilities are fixed. After the determination of the most efficient realization methods, two system programs are generated for defect detection on PCBs. As for the interpretation step, the verifications of both the defect detection algorithms and the developed system software are made by many test runs of the software on some sample PCBs and a master artwork containing many typical defects. In this way, a prototype realization of the rule based visual inspection system is completed in our image processing laboratory.

2 Mathematical Morphology

There is an extensive and yet increasing literature on mathematical morphology and for this reason, instead of supplying precise definitions of mathematical morphology here, we will only describe qualitatively the fundamental operations, and direct the reader to three of the related excellent references, [1], [2], [3].

Hit or Miss transformation is a template matching process where the Hits are marked by foreground pixels (with value 1) and the Misses are marked by background pixels (with value 0). Erosion operation as the name suggests, tappers the foreground structures and may alter the connectivity properties of the image. Dilation operation enlarges the foreground structures and may also alter the connectivity properties of the image. ([1], [2])

A *skeletal element* is a foreground pixel that is necessary to maintain the 8-connectivity of its 8-neighborhood; i.e. changing the pixel to a background pixel breaks the 8-connectivity between at least two other foreground pixels in the 8-neighborhood. An *n-join* is a foreground pixel with n foreground 8-neighbors; hence, a join can be of order 0 to 8. A *T-join* is a 3-join whose 8-neighborhood contains only skeletal elements. ([3], [4])

Symmetrical thinning algorithm tappers the foreground structures **without altering** the connectivity properties of the image. Pruning algorithm trims "hair" like protrusions which may result as an artifact of the symmetrical thinning operation. In this work, a symmetrical thinning and a pruning algorithm based on the cascade implementation of [5] are used.

3 Defect Detection Algorithms

3.1 Isotropic Structuring Elements

In order to check a **common** minimum width requirement on a binary image in the Euclidean 2-space, E^2, containing some foreground stripes of arbitrary widths and inclinations, one can consider the enclosure of a single circular structuring element (SE) into the stripes. This **isotropic** SE having the common minimum width as its diameter enables the use of a single template on **all** foreground stripes with **arbitrary** inclinations.

A similar problem in the discritized Euclidean 2-space, Z^2, can be handled by considering discrete approximations to circular structures. These approximate SEs must have the following two necessary features. They must be as similar as possible to the original circular structure and decomposable into a dilation of smaller sized SEs. This last condition is related to implementation considerations. In general octagonal SEs have these two features.

In case of rectangular sampling with unequal horizontal and vertical sampling periods, the isotropic SEs turn out to be elliptic structures with a fixed axis length ratio. The similarity and decomposability requirements are also to be satisfied. In this work, due to our available hardware, we extensively used discrete approximations to elliptic structures.

3.2 Algorithm for Removing Lands

Given a PCB image with drilled through holes and vias, this new image segmentation algorithm removes the conductor lands surrounding these holes from the image and thus generates two binary images, one containing only the conductor traces and the other only the lands surrounding the holes. This enables the application of only the related design rule checking algorithms to these structures and hence avoids false alarms. This algorithm works as follows:

1. A 3 level digital image is captured from the PCB (Figure 1.a). In this image, pixels with the medium gray level have value 0 and indicate the substrate, black pixels having value 1 indicate the conducting structures, and the white pixels having value 2 indicate the places where there are holes.
2. The following look-up table (LUT) transformation is applied to the image of Figure 1.a in order to extract the locations of the holes. $0 \rightarrow 0$, $1 \rightarrow 0$, $2 \rightarrow 1$. Figure 1.b shows the resultant binary image.
3. The image of Figure 1.b is dilated with an isotropic SE (an elliptic one). Hence, as seen in Figure 1.c, the hole locations are enlarged so as to cover their surrounding lands.
4. The following LUT transformation is applied to the image of Figure 1.a to extract only the conducting structures. $0 \rightarrow 0$, $1 \rightarrow 1$, $2 \rightarrow 0$. Figure 1.d shows the resultant binary image.
5. The images of Figure 1.c and Figure 1.d are logically ANDed. The resultant binary image, Figure 1.e, contains only the lands.
6. The images of Figure 1.d and Figure 1.e are logically EXORed. The resultant binary image given in

Figure 1.f contains only the conductor traces.

3.3 Algorithm Verifying MCS Requirement

This new algorithm checks the MCS requirement on binary PCB test images and directly detects the violations and also their places without the need for a time consuming comparison step which is typically needed in most of the similar applications [4], [5], [6]. The algorithm works as follows:

1. The original PCB test image (Figure 2.a) to be inspected against defects is dilated once by an isotropic SE (an elliptic one). This SE is specifically designed to induce new symptomatic foreground connections between neighboring conducting structures only at the defective places. Figure 2.b shows the resultant image.
2. The image of Figure 2.b is symmetrically thinned so that the symptomatic features are preserved and the pixels in the resultant image, Figure 2.c, reduce to either symptomatic pixels or pixels present in the original image.
3. The image of Figure 2.c is pruned so that possible false alarms due to "hair" like protrusions are avoided. Figure 2.d shows the resultant image.
4. The images of Figure 2.a and Figure 2.d are logically ORed. As seen in the image of Figure 2.e, this results in the inclusion of the symptomatic pixels into the original image.
5. The images of Figure 2.a and Figure 2.e are logically EXORed. This step extracts only the pixels which are not common in the above mentioned images. As Figure 2.f shows, these are the symptomatic pixels showing the defective locations.

3.4 Algorithm Verifying MCTW Requirement

For detecting MCTW violations, we note that the MCTW requirement in the original image can be interpreted as the minimum spacing requirement between the substrate traces in the (bitwise) complemented image. Hence, in order to detect this class of defects we apply the algorithm given in Section 3.3 to the background of the test images with appropriate parameter changes in the algorithm. These parameter changes cover the SE, and the number of symmetrical thinning and pruning operations applied to the background pixels.

3.5 Algorithm Verifying MLW Requirement

Since the lands are in the form of ring shaped conductor traces, the extension of the algorithm of Section 3.4 to the case of lands is rather straightforward. Hence, we apply the algorithm of Section 3.4 to a subimage containing the lands only, after appropriate parameter changes in the algorithm. As before, these parameter changes cover the SE, and the number of symmetrical thinning and pruning operations applied to the background pixels.

3.6 An Algorithm to Supersede the Algorithms of Sections 3.3 and 3.4

Although the two defect detection algorithms of Sections 3.3 and 3.4 work very well, for an industrial application of the system a faster algorithm was needed to replace them. Such an algorithm proposed in [7] was based on a special purpose hardware architecture and had some uncommon features which made it impossible to be implemented in a useful way on a general class of image processors. So we had to adopt the implementation technique by keeping the idea of the algorithm essentially the same. The algorithm as implemented by our modified technique and used in our system works as follows:

1. The original PCB test image, Figure 3.a, to be inspected against defects is dilated once by a properly chosen isotropic SE. This step has exactly the same significance as the first step of the algorithm given in Section 3.3. Figure 3.b shows the resultant image.
2. Images of Figure 3.a and Figure 3.b are logically EXORed. This operation as can be seen in the resultant image of Figure 3.c, induces some "T" shaped foreground structures only in the defective places. These structures which are certain to reduce to "T joins" after symmetrically thinning the image of Figure 3.c down to its skeleton, constitute the symptomatic features we want to induce. The places where there are no defects, will reduce to connected chains of "2 joins" after appropriate symmetrical thinning.
3. In order to induce similar symptomatic features at the defective places where the MCTW requirement is violated, we proceed as follows. The original PCB test image given in Figure 3.a is eroded once by a properly chosen isotropic SE. This step induces new symptomatic background connections at the places where the MCTW requirement is violated. Figure 3.d shows the resulting image.
4. Images of Figure 3.a and Figure 3.d are logically EXORed. This operation as can be seen in the resultant image of Figure 3.e, induces some "T" shaped foreground structures only in the defective places. As can be easily judged, this step has exactly the same significance as the second step of this algorithm.
5. At this point, in order to speed up the algorithm we note that exactly similar kinds of further processing is required on the images of Figure 3.c and Figure 3.e. We should symmetrically thin both of the images and detect the places of "T joins". In fact, even

exact skeletonization (i.e. application of the symmetrical thinning operation until idempotance), which is very time consuming, is not required. Instead, we can skeletonize only the defect-free places (i.e. generate connected curves of "2 joins" in defect-free places), and generalize the class of symptomatic features from "T joins" to "all joins except 2 joins". This will speed up the algorithm further and also enable us to detect some defects which otherwise we wouldn't be able to detect. To perform the common further processing on both of the images simultaneously, we can combine these two nonoverlapping images. But, in order not to miss some symptomatic features, before combining the two images we make them spatially disjoint (separated) by an appropriate sequence of symmetrical thinning and pruning operations. The resultant images obtained after the application of this processing to the images of Figure 3.c and Figure 3.e are shown in Figure 3.f and Figure 3.g, respectively.

6. The images of Figure 3.f and Figure 3.g are logically ORed. As Figure 3.h shows, by this operation we achieve our goal of combining these two images into one which contains all of the symptomatic features of both images.

7. The image of Figure 3.h is symmetrically thinned and pruned for a sufficient number of times so that the defect-free places are reduced to connected curves of "2-joins". Figure 3.i shows the resultant image.

8. In the image of Figure 3.i, the positions of the joins of order different than 2 are extracted. The result of this operation is shown in Figure 3.j. The foreground pixels in this image show the defective places where the design rules related to the MCS or to the MCTW requirements are violated.

9. As a final step, our defect detection software encircles the defective locations by red colored rectangular frames on the gray level test image captured from the PCB. This operation is shown in Figure 3.k.

3.7 Implementation of Morphological Operations

A straightforward implementation of morphological operators employing SEs of size at most 3x3, by using 3x3 2-D convolution followed by a table lookup operation, is mentioned in [8]. The fundamental algorithm proposed in this reference, requires the use of a 512 entry lookup table. In the same reference, an adopted version of this algorithm so as to take into account the implementations based on lookup tables restricted to 256 entries is also given. However, our image processor containing 256 entry lookup tables does not enable an efficient implementation based on this modified technique due to its hardware architecture. For this reason, we had to find another implementation technique. The technique we propose is similar to the technique of [8], however we relax the condition of exactly determining the pixel distribution under the convolution kernel. Instead, we choose the convolution kernel elements such that the result of the convolution on a 3x3 neighborhood specifies the image pixel distribution in that 3x3 neighborhood (of the image) just with sufficient information so that the correct decision can be made about the morphological operation's result in that neighborhood. For example, in the case of implementing the Hit or Miss transformations, the detection of the presence of a foreground pixel in a location where a background pixel should exist, or vice versa, specifies the resultant pixel uniquely. For this purpose, in choosing the convolution kernel elements, we make use of the significance of each pixel in the SE in terms of its effect on the result. An example will clarify the principles of the technique we propose. In Figure 4, the implementation of a single element foreground thinning operation is considered. This operation requires the implementation of Hit or Miss transformation followed by an EXOR operation. Both of these operations will be realized in a single step. In part (a) of this figure the thinning SE to be implemented is given. Part (b) of the same figure shows the 2-D 3x3 convolution kernel elements chosen. As can be seen, equal convolution kernel values are assigned to pixels in the SE which should belong to the background (10), or to the foreground (2). "Don't Care" pixel locations are weighted by zero. The only departure from this convention is in the center pixel which is weighted by 5 although it should belong to the foreground. In fact such a choice is indispensable, since in case of a Miss we should carry the pixel value in the original image to the thinned image without any change and for this reason we should be able to uniquely identify the value of the center pixel in the original image. In Figure 4.c, based on this argument, the entries of the LUT to be used to transform the convolution results to binary pixel values are given. This table covers all possible convolution results performed on a binary image and all other entries of the LUT not listed in this table are "Don't Care" conditions. Note that the convolution result corresponding to a Hit is 11, and in this case the center pixel in the original image which is a foreground pixel is subtracted from the original image and replaced by a background pixel in the thinned image. In case of all other entries corresponding to Miss conditions, the original value of the center pixel is not changed in the thinned image.

References

[1] J. Serra, *Mathematical Morphology and Image Analysis,* Academic Press, New York, 1982.

[2] R.M. Haralick, S.R. Sternberg, X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, July 1987, pp. 532-550.

[3] T.Y. Kong, A. Rosenfeld, "Digital Topology: Introduction and Survey," *Comput. Vision Graphics Image Process.* **48**, 1989, pp. 357-393.

[4] J.R. Mandeville, "Novel Method for Analysis of Printed Circuit Images," *IBM J. Res. Develop.*, vol. 29, no. 1, January 1985, pp. 73-86.

[5] A.M. Darwish, A.K. Jain, "A Rule Based Approach for Visual Pattern Inspection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, January 1988, pp. 56-68.

[6] A.E. Kayaalp, R. Jain, "Inspection of IC's for Random Pattern Defects Using SEM Images," College of Engineering, The University of Michigan, Ann Arbor, Michigan; Program in Automated Semiconductor Manufacturing, October 1985.

[7] Q.-Z. Ye, P.E. Danielsson, "Inspection of Printed Circuit Boards by Connectivity Preserving Shrinking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, September 1988, pp. 737-742.

[8] F.A. Gerritsen, P.W. Verbeek, "Implementation of Cellular-logic Operators Using 3*3 Convolution and Table Lookup Hardware," *Comput. Vision Graphics Image Process.* **27**, 1984, pp. 115-123.

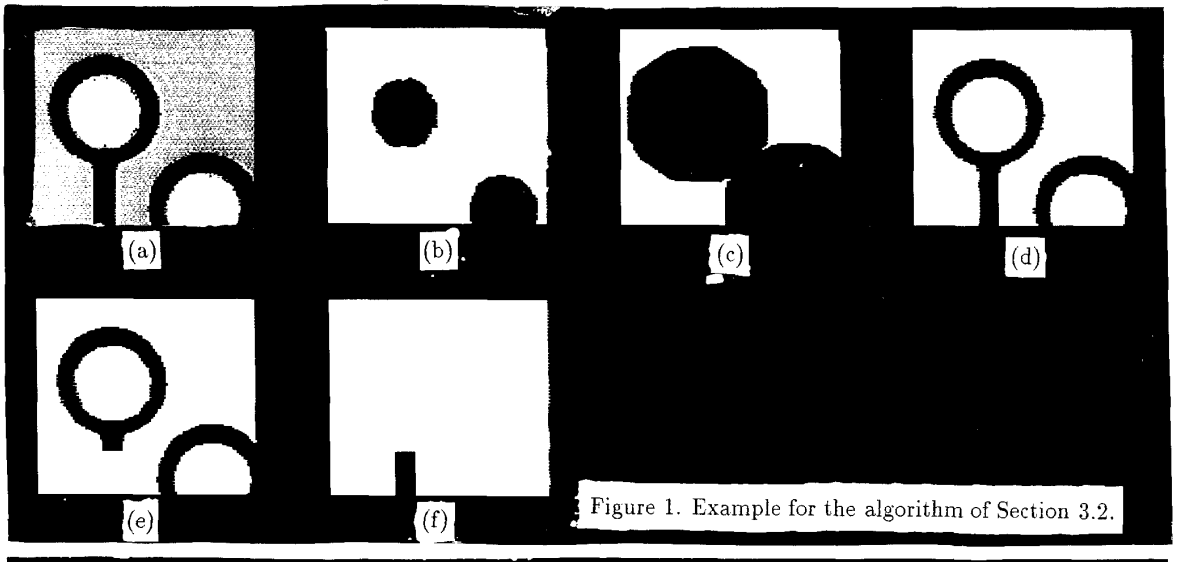

Figure 1. Example for the algorithm of Section 3.2.

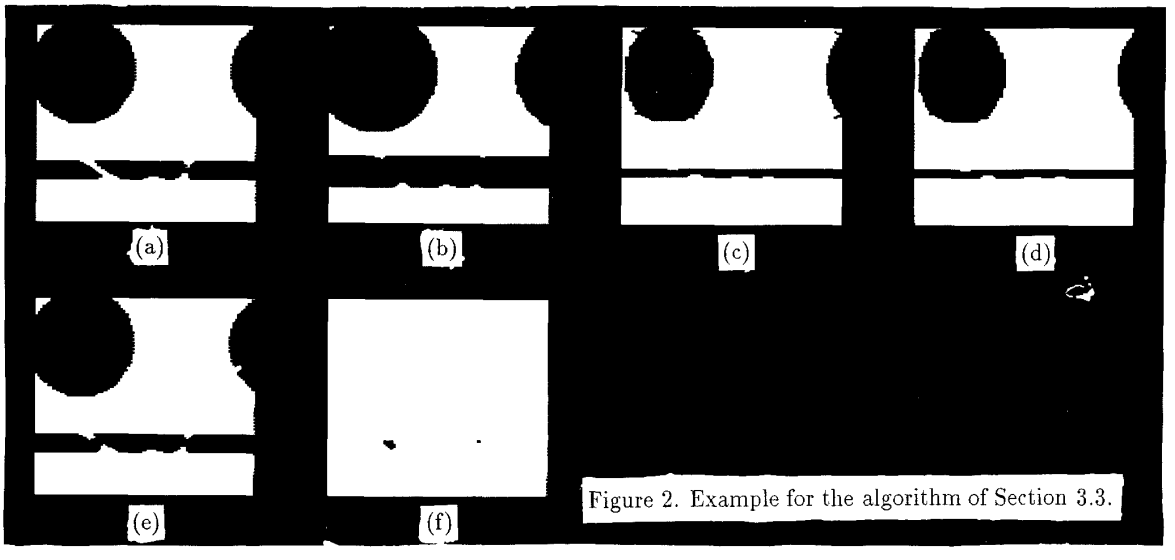

Figure 2. Example for the algorithm of Section 3.3.

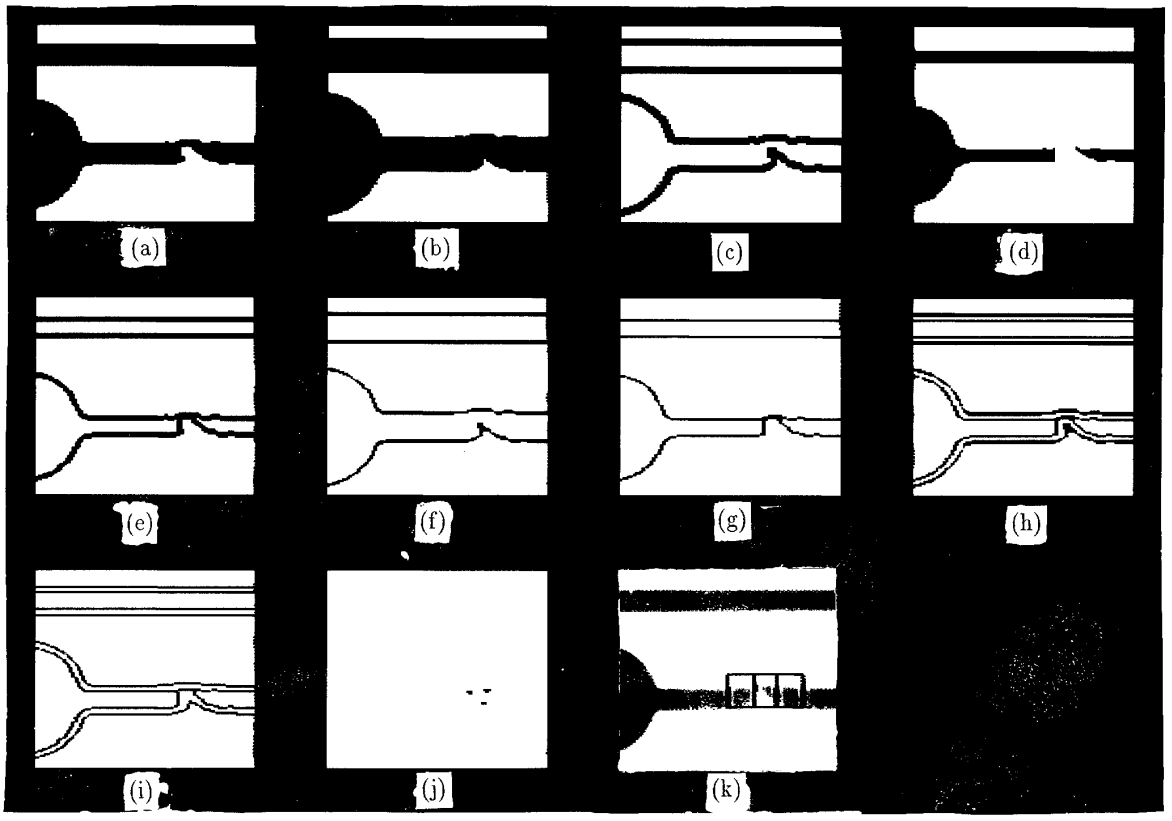

Figure 3. Example for the algorithm of Section 3.6.

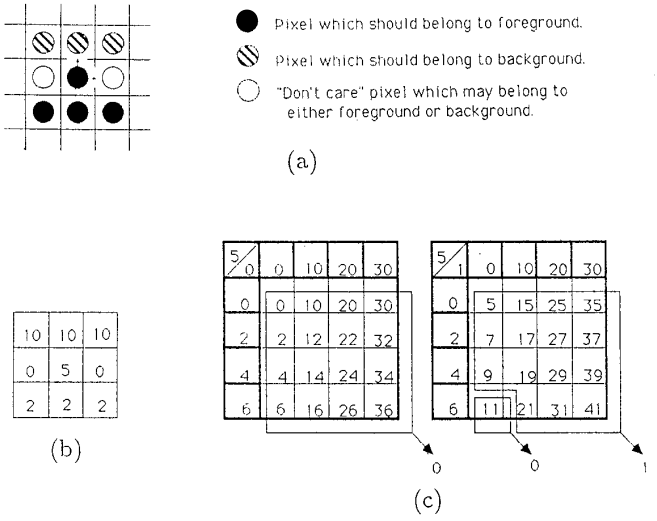

Figure 4. Example for the algorithm of Section 3.7, (a) the structuring element to be used, (b) convolution kernel coefficients, (c) lookup table contents.