An Adaptive Texture and Shape Based Defect Classification

Jukka Iivarinen
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O. Box 2200
FIN-02015 HUT, Finland
Jukka.Iivarinen@hut.fi

Ari Visa
Lappeenranta University of Technology
Department of Information Technology
P.O. Box 20
FIN-53851 Lappeenranta, Finland
Ari.Visa@lut.fi

Abstract

In this paper classification of surface defects is considered. The classification system consists of several classifiers whose outputs are combined in order to produce the final classification. The self-organizing maps (SOMs) are used as classifiers. Each SOM is taught unsupervised with examples of defects. Classification is based on the internal structure and the shape characteristics of defects. Texture features from the co-occurrence matrix and the gray level histogram are used to describe the internal structure. The set of simple shape descriptors is used for shape characterization The results of experiments with base paper defects are encouraging.

1. Introduction

Recognition of objects is one of the basic tasks in computer vision applications. The goal of object recognition is to find a description which contains sufficient information to distinguish between different target objects. Recognition is usually based on gray levels or colors and shape characteristics [14, 12] of target objects.

Recently, neural network methods have become popular due to their capability to adapt to new circumstances in an unsupervised way. Some of the methods have been based on a simple one-layer neural network, for example, the self-organizing map (SOM) [11]. Yläkoski and Visa developed a method for timber classification that was based on a SOM and texture measures [22]. The method had a grammar that described the quality of a surface by the means of defects. Visa et al. proposed the SOM and co-occurrence matrices to segmentation and classification of satellite images [21, 20], and extended the method to surface segmentation [9].

The most demanding surface inspection problems contain the subproblems: defect detection, defect classification, and process control. These problems can be solved only under certain circumstances. For instance, for running web processes there exist commercial solutions [13, 16, 4]. A review on texture defect detection can be found in [17]. The complete defect classification system is described in a recent paper by Brzakovic and Vujovic [2]. Other related work can be found in [1, 19, 5]. The defect classification part is discussed in this paper. Surface defects, which are extracted from base paper samples by an unsupervised segmentation procedure [9], are used as a case study.

2. The proposed classification procedure

In many image processing tasks it is common to have different measurements taken from an input data (usually an image). The combination of these measurements (i.e., features) is not always straightforward. The usual way is to collect them into one feature vector and use it as an input for a classifier. If features measure different properties from data, it is not always sensible to use them together in one feature vector. Instead, it may be better to split the classification problem so that there are as many sub-classifiers as there are different feature sets. Each sub-classifier gives its own classification. These classifications are then combined. For example, in case of surface defects we are interested in the shape and the internal structure of them, so at least two classifiers are needed (one for the shape and one for the internal structure).

The proposed classification system is depicted in Figure 1. It has N distinct classifier blocks and a combiner unit. Each classifier block consists of a feature extraction unit and a classification unit. The feature extraction unit calculates some predefined set of features from input data and inputs the obtained feature vector into the classification unit. Here we are using the self-organizing map (SOM) as a classifier but any other classifier may also be used. The best-matching unit (BMU) is searched for the given feature vector. The BMU is the SOM unit with the smallest distance between unit's weight vector and the given feature vector. The output

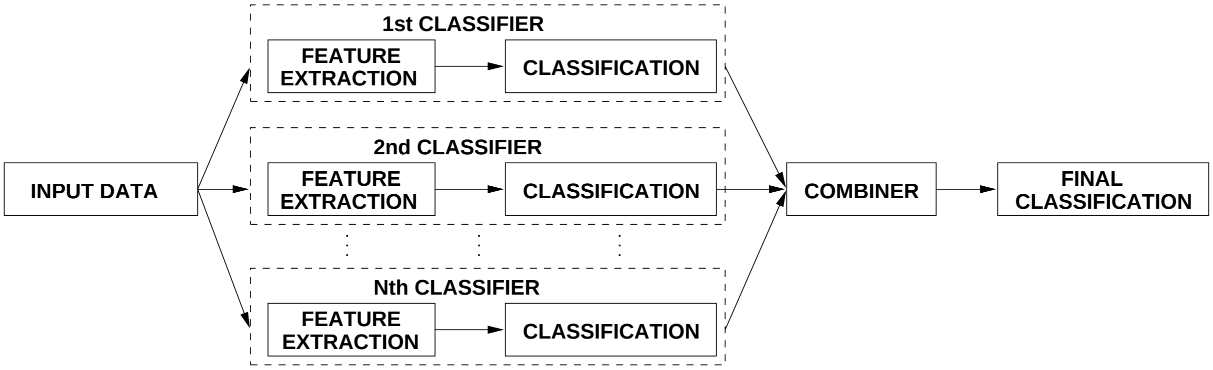

Figure 1. The proposed classification procedure.

of the classification unit consists of two items: the error between the weight vector of the BMU and the given feature vector, and the label of the BMU.

The error is used to determine if the given feature vector belongs to the BMU or not. If the latter is the case then we are having a case that has not been encountered previously. A new unit can then be added to the SOM to represent a new class. The weight vector of the new unit is set equal to the feature vector, and the new unit is given a label by hand.

In the combiner block the labels from each classifier are combined to produce the final classification. The labels should be verbal descriptions so that the final classification would also be a verbal description.

2.1. Codebook generation

The codebooks are formed with the self-organizing map (SOM) algorithm [11]. The advantage of using the SOM in codebook generation lies in the fact that the codebooks can be formed in an unsupervised way, simply by clustering unclassified training samples together, and then labeling the SOM units (or clusters) to represent different classes. It is easier to give a label to a cluster of training samples than to each training sample alone, especially if the classes are hard to determine and they are overlapping. In addition to manually given labels each SOM unit can be assigned a few representative defects. They act as examples of typical defects that belong to a SOM unit.

To train the SOM a set of feature vectors is extracted from training samples. The weight vectors of the SOM units are initialized randomly in input space. Other initialization methods can be also used, for example, the weight vectors can be placed in an orderly fashion along a two-dimensional subspace spanned by the two principal directions of input data. Then the weight vectors are moved according to the SOM learning rule [11] to positions that correspond cluster centers as accurately as possible. The training data is fed into the SOM several times.

3. Case study: Defect classification

The proposed classification scheme is demonstrated by defects which are extracted from base paper samples by an unsupervised segmentation procedure [9]. Classification of defects may be hard even to an expert. There exists several defect classes that overlap in some extent. A covering collection of different defects is also needed to teach the classifier. However, it is hard and time-consuming to collect defects. The proposed approach allows new, previously encountered defects to be added to the codebooks gradually during a long period.

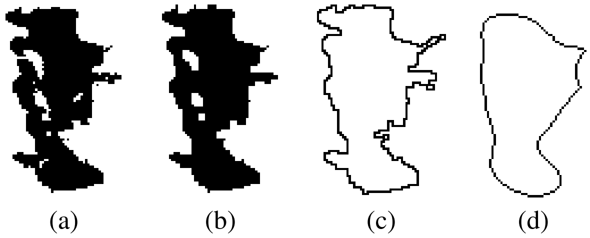

Figure 2. (a) The segmented defect, (b) the defect after closing, (c) the contour of the defect, and (d) the Gaussian smoothed contour of the defect.

Pre-processing of defects In Figure 2(a) is a defect after the segmentation procedure. It has a ragged contour with small holes and gulfs. Before any classification is made, some pre-processing steps must be carried out. The basic morphological operations [7], erosion and dilation, are used in processing a defect. To fill up small holes and gulfs in a defect a morphological closing operation (dilation followed by erosion) is applied to the defect (Figure 2(b)). The contour is obtained by subtracting the closed defect from the dilated defect (Figure 2(c)). Thus the obtained contour is larger than the true contour. Finally the contour is smoothed with the Gaussian filter (Figure 2(d)).

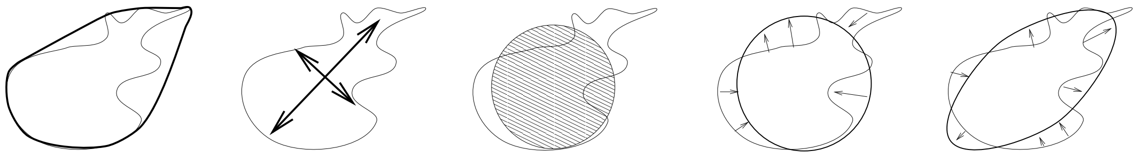

Figure 3. Five simple shape descriptors.

Characterization of shape Shape is an important characteristic of an object. The goal of shape recognition is to find a description which contains sufficient information to distinguish between differently shaped target objects. We have evaluated in detail three different methods, namely the chain code histograms [10], the pairwise geometric histograms [3], and combinations of some simple shape descriptors. The simple shape descriptors seem to provide the best compromise between recognition power and computational complexity for our purposes [8].

Five simple shape descriptors are used to characterize the shape of a defect (Figure 3). The descriptors are: convexity (CONV), principal axis ratio (PCR), compactness (COMP), circular variance (CVAR), and elliptic variance (EVAR). They are translation, rotation, and scale invariant shape descriptors.

Convexity is defined as the ratio of perimeters of the convex hull and the original contour,

$$\mathrm{CONV} = \frac{\mathcal{P}_{convexhull}}{\mathcal{P}}. \tag{1}$$

Principal axes of a given object can be uniquely defined as the segments of lines that cross each other orthogonally in the centroid of the object and represent the directions with zero cross-correlation. Let us denote the elements of the covariance matrix $\mathbf{C}$ of a contour as

$$\mathbf{C} = \begin{pmatrix} c_{xx} & c_{xy} \\ c_{yx} & c_{yy} \end{pmatrix}. \tag{2}$$

Then the principal axes ratio can be calculated directly,

$$\mathrm{PCR} = \frac{c_{yy} + c_{xx} - \sqrt{(c_{yy} + c_{xx})^2 - 4(c_{xx}c_{yy} - c_{xy}^2)}}{c_{yy} + c_{xx} + \sqrt{(c_{yy} + c_{xx})^2 - 4(c_{xx}c_{yy} - c_{xy}^2)}}. \tag{3}$$

Compactness is the ratio of the perimeter of a circle with equal area as the original object and the perimeter of the original contour,

$$\mathrm{COMP} = \frac{\mathcal{P}_{circle}}{\mathcal{P}} = \frac{2\sqrt{\mathcal{A}\pi}}{\mathcal{P}},\ \mathcal{A}_{circle} = \mathcal{A}. \tag{4}$$

Circular variance is the proportional mean-squared error with respect to a solid circle,

$$\mathrm{CVAR} = \frac{1}{N\mu_r^2}\sum_i (||\mathbf{p}_i - \boldsymbol{\mu}|| - \mu_r)^2, \tag{5}$$

where N is the number of contour points, $\mathbf{p}_i = (x_i\ y_i)^T$ is the contour point, $\boldsymbol{\mu}$ is the centroid and μ_r is the mean radius of the contour. Thus, CVAR gives zero for a perfect circle and increases along shape complexity and elongation. Elliptic variance is the proportional mean-squared error with respect to an ellipse having an equal covariance matrix, $\mathbf{C}_{ellipse} = \mathbf{C}$,

$$\mathrm{EVAR} = \frac{1}{N\mu_{r_C}^2}\sum_i (\sqrt{(\mathbf{p}_i - \boldsymbol{\mu})^\top \mathbf{C}^{-1}(\mathbf{p}_i - \boldsymbol{\mu})} - \mu_{r_C})^2,$$
$$\mu_{r_C} = \frac{1}{N}\sum_i \sqrt{(\mathbf{p}_i - \boldsymbol{\mu})^\top \mathbf{C}^{-1}(\mathbf{p}_i - \boldsymbol{\mu})} \tag{6}$$

Variations of most of these shape descriptors have been widely used in object recognition [12, 18]. Each descriptor alone is insufficient for a complex recognition task, but the combination of them is shown to have good recognition capabilities and low computation costs [15, 8].

Characterization of internal structure In addition to shape a defect has some kind of internal structure that consists of dark and light areas, holes, etc. The internal structure can be regarded as a distribution of gray levels in a defect. This distribution can be characterized as a gray-level histogram,

$$p(i) = \frac{n_i}{n},\ \ i = 0, 1, ..., N-1, \tag{7}$$

where n_i is the number of pixels having gray level value i in a defect, n is the total number of pixels in a defect, and N is the total number of gray levels.

A set of texture features calculated from the co-occurrence matrix [6] is also used to characterize defect's internal structure. The co-occurrence matrix $\mathbf{C}_\mathbf{d}$ is a second order statistical measure of gray level variation. It indicates the joint probability of gray level occurrences at a displacement $\mathbf{d} = (d_x\ d_y)$ in an image. The co-occurrence matrix $\mathbf{C}_\mathbf{d}$ is formed to a defect and four texture features, energy, contrast, entropy, and mean, are calculated from it.

$$\mathrm{ENERGY} = \sum_i \sum_j C_\mathbf{d}(i,j)^2 \tag{8}$$

$$\mathrm{CONTRAST} = \sum_i \sum_j (i-j)^2 C_\mathbf{d}(i,j) \tag{9}$$

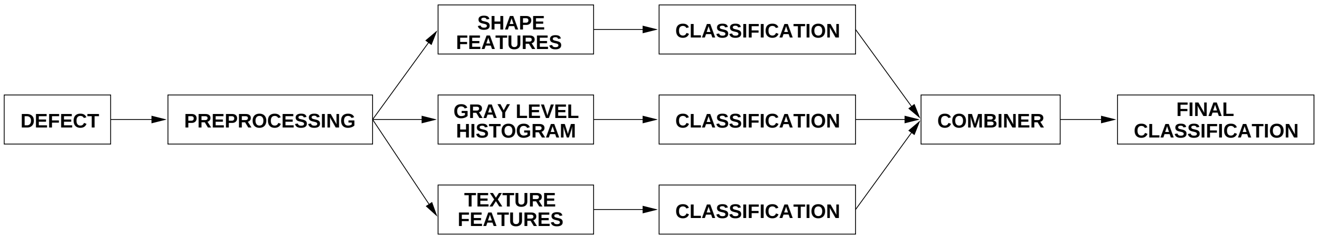

Figure 4. The defect classifier.

$$\text{ENTROPY} = -\sum_i \sum_j C_{\mathbf{d}}(i,j) \log C_{\mathbf{d}}(i,j) \quad (10)$$

$$\text{MEAN} = \sum_i \sum_j i C_{\mathbf{d}}(i,j) \quad (11)$$

Each feature is normalized to $[0, 1]$.

3.1. Experimental setup and results

The defect classifier is depicted in Figure 4. There are three distinct blocks, one block for each feature set. Each block has a feature extraction unit and a classification unit. Thus three codebooks are also needed, a shape codebook, a gray-level histogram codebook, and a texture codebook. The outputs for each block are combined to produce the final classification. The final classification is a verbal description for a given defect.

The actual system consisted of two separate classifiers as in Figure 4, one classifier for defects with holes and another classifier for defects without holes. The presence of hole is so important that to guarantee that holes are detected it is better to make a simple classification into holes and non-holes before any further classification. This hole-detection is made simple by calculating the number of hole pixels in a defect. The classification system thus consisted of six SOMs each one having 9 units. There were 111 defects that were divided into the training set (79 defects) and the test set (32 defects). Only holes (clean hole, weak hole etc.) and spots (light spot, slime spot etc.) were included in both sets. The training set consisted of 40 hole defects and 39 non-hole defects. After training each unit was assigned few representative defects to act as examples of typical defects that belong to a unit. Based on these example defects, each unit was given a label by hand. These labels are verbal descriptions that describe the defect class of a unit. In case of shape codebooks, labels are fuzzy ones, for example, elongated, much elongated, circle-like, and quite circle-like. In texture codebooks, on the other hand, labels are more precise, for example, dark hole, light hole, and weak hole.

An example defect and its features are depicted in Figure 5(a). The shape features (SSD) are calculated from the Gaussian smoothed contour of a defect and normalized. Notice also that the SSD features result in rotation invariant classification. The gray-level histogram (GLH) and the texture features (TEX) are calculated using only 8 gray levels. The displacement used in co-occurrence matrix calculation was $\mathbf{d} = (2, 0)$. The SSD and TEX features are normalized to have zero mean and unit variance. These normalizing constants were calculated from the training samples. In Figure 5(b)-(d) are the classification results. Each column shows the best-matching codebook defect when using different features. In the bottom row are the numbers and the weight vectors of the best-matching SOM units. The first and the second row show one of the typical defects (with its feature vector) that belongs to the best-matching unit. The labels of the best-matching units are *elongated, smooth shape* (SSD), *light spot* (GLH), and *light spot* (TEX). The final classification is then *elongated, smooth, light spot*. Another example defect is in Figure 6(a). In Figure 6(b)-(d) are the classification results. The labels of the best-matching units are *quite elongated, smooth shape* (SSD), *dark hole* (GLH), and *mixed hole* (TEX). The final classification is *quite elongated, smooth, dark mixed hole*.

4. Conclusion

The proposed multi-level classification procedure is a general one and can be used in different classification problems. Reselection of features may be necessary to adapt the proposed classifier to work with different types of objects. The classifier is taught with examples of defects. Shape and internal structure characteristics of defects are learned from examples. The self-organizing maps (SOMs) are used to cluster example defects. The obtained clusters (the SOM units) are then given an interpretation by hand. It is desirable to have a lot of teaching data available. However, it is usually hard to collect a covering collection of different defects. To alleviate this problem the proposed classification scheme allows new, previously unknown, defects to be added to the codebook gradually during a long period.

The results of experiments with base paper defects are encouraging. The evaluation of the proposed method with more data and the refinement of the defect classification scheme are the main objectives for future work.

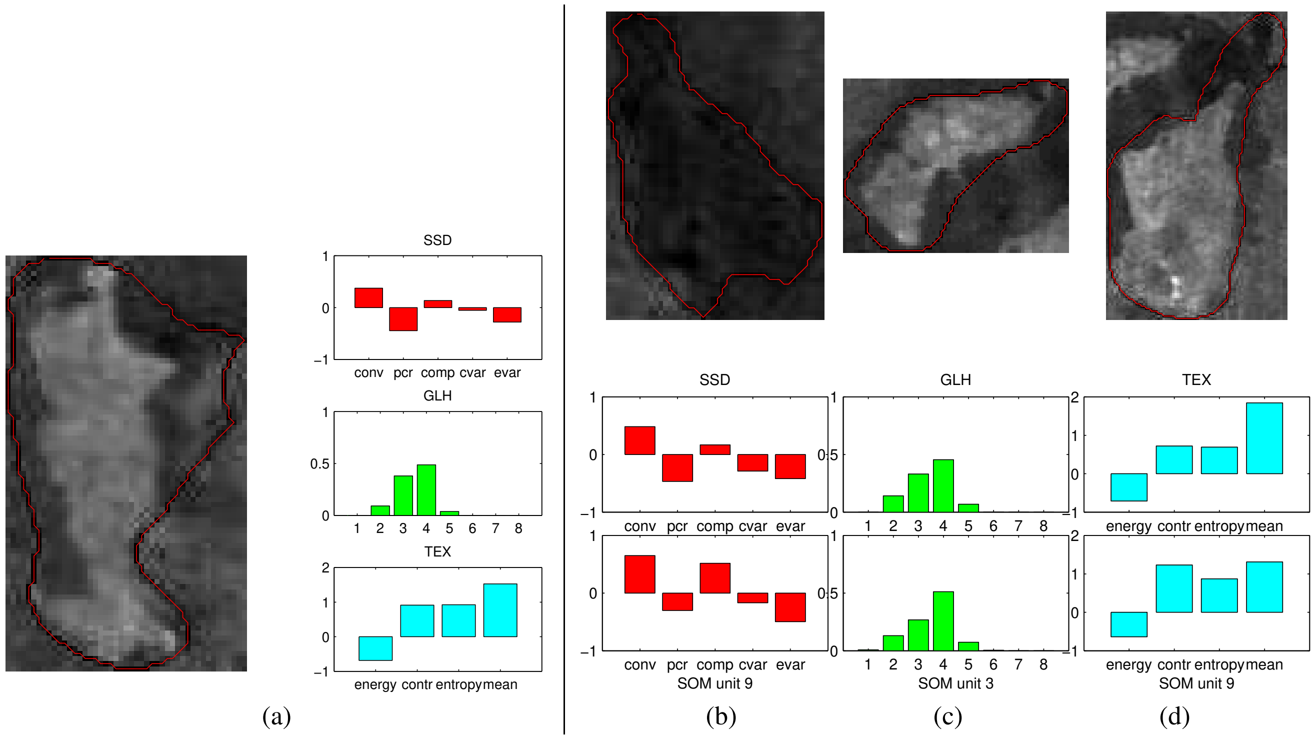

Figure 5. (a) A defect and its shape features (SSD), gray level histogram (GLH), and texture features (TEX). The best matching defects to the defect in (a) with respect to (b) the shape, (c) the gray level histogram, and (d) the texture features.

Acknowledgment

The authors wish to thank Mr Markus Peura from Helsinki University of Technology for work on shape descriptors and Mr Juhani Rauhamaa from ABB Industry Oy. The financial support of the ABB Industry Oy and the Technology Development Centre of Finland (TEKES's grant 4172/95) is gratefully acknowledged.

References

[1] M. Borghesi, V. Cantoni, and M. Diani. An industrial application of texture analysis. In *Proceedings of the 7th International Conference on Pattern Recognition*, volume 1, pages 420–423, Montreal, Canada, July 30–Aug. 2 1984.

[2] D. Brzakovic and N. Vujovic. Designing a defect classification system: A case study. *Pattern Recognition*, 29(8):1401–1419, 1996.

[3] A. C. Evans, N. A. Thacker, and J. E. W. Mayhew. Pairwise representations of shape. In *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, volume 1, pages 133–36, The Hague, The Netherlands, Aug. 30–Sept. 3 1992.

[4] J. E. Graf, S. T. Enright, and S. I. Shapiro. Automated web inspection ensures highest quality nonwovens. *Tappi Journal*, 78(9):135–138, 1995.

[5] J.-L. C. Guay. A machine vision system for real time inspection of moving web. In *Intelligent Robots and Computer Vision VIII: Systems and Applications*, Proc. SPIE 1193, pages 204–228, 1989.

[6] R. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, Nov. 1973.

[7] R. M. Haralick, S. R. Sternberg, and X. Zhuang. Image analysis using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(4):532–550, July 1987.

[8] J. Iivarinen, M. Peura, J. Särelä, and A. Visa. Comparison of combined shape descriptors for irregular objects. In *Proceedings of the 8th British Machine Vision Conference*, volume 2, pages 430–439, University of Essex, UK, Sept. 8–11 1997.

[9] J. Iivarinen, J. Rauhamaa, and A. Visa. Unsupervised segmentation of surface defects. In *Proceedings of the 13th International Conference on Pattern Recognition*, volume IV, pages 356–360, Wien, Austria, Aug. 25–30 1996.

[10] J. Iivarinen and A. Visa. Shape recognition of irregular objects. In D. P. Casasent, editor, *Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling*, Proc. SPIE 2904, pages 25–32, 1996.

[11] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, 1995.

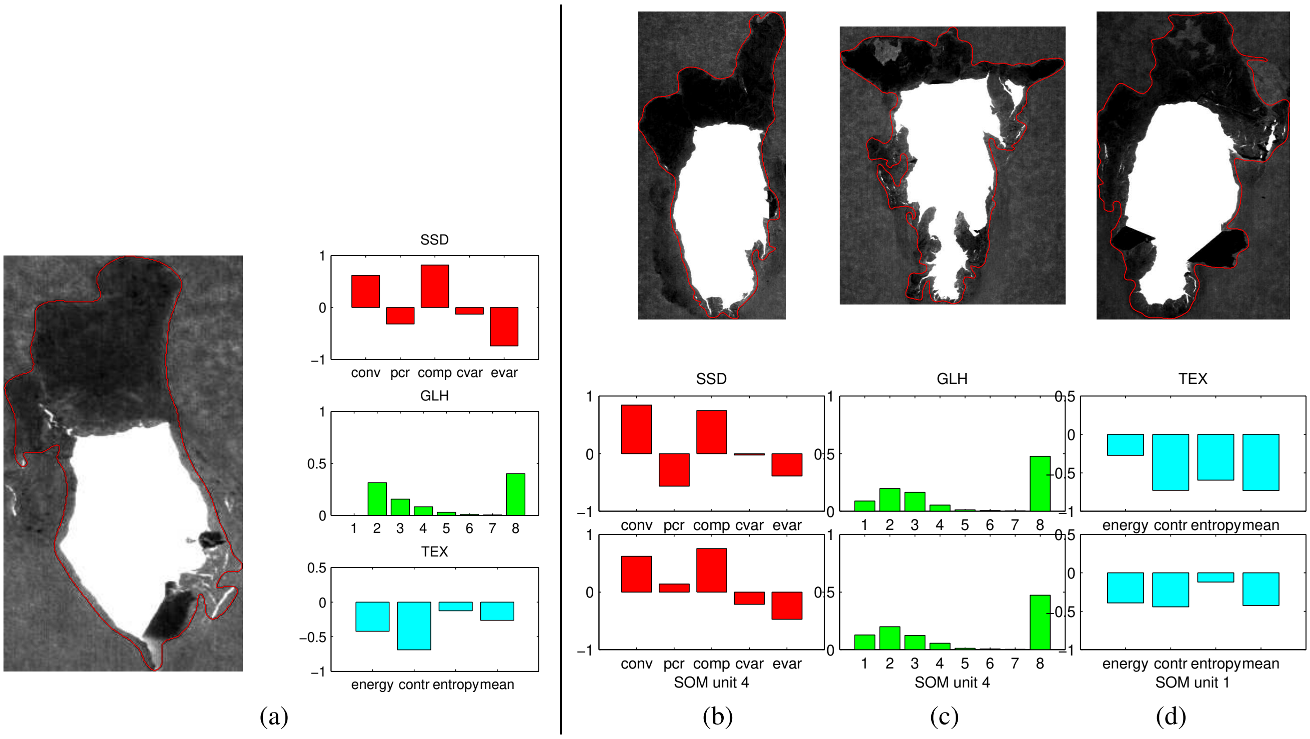

Figure 6. (a) A defect and its shape features (SSD), gray level histogram (GLH), and texture features (TEX). The best matching defects to the defect in (a) with respect to (b) the shape, (c) the gray level histogram, and (d) the texture features.

[12] S. Marshall. Review of shape coding techniques. *Image and Vision Computing*, 7(4):281–294, Nov. 1989.

[13] L. Mäkelin, J. Moisio, and J. Järvinen. From hole detection to image processing and information technology. In *Proceedings of 50th Appita Annual General Conference*, volume 1, pages 817–822, 1996.

[14] T. Pavlidis. A review of algorithms for shape analysis. *Computer Graphics and Image Processing*, 7:243–258, 1978.

[15] M. Peura and J. Iivarinen. Efficiency of simple shape descriptors. In C. Arcelli, L. P. Cordella, and G. Sanniti di Baja, editors, *Advances in Visual Form Analysis*, pages 443–451. World Scientific, Singapore, 1997.

[16] H. Ruuska and I. Akerberg. Practical inspection systems for nonwoven diaper linerfabrics. *Tappi Journal*, 78(9):135–138, 1995.

[17] K. Y. Song, M. Petrou, and J. Kittler. Texture defect detection: A review. In *Applications of Artificial Intelligence X: Machine Vision and Robotics*, Proc. SPIE 1708, pages 99–106, 1992.

[18] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. Chapman & Hall Computing, London, 1993.

[19] M. Unser and F. Ade. Feature extraction and decision procedure for automated inspection of textured materials. *Pattern Recognition Letters*, 2(3):185–191, Mar. 1984.

[20] A. Visa and J. Iivarinen. Evolution and evaluation of a trainable cloud classifier. *IEEE Transactions on Geoscience and Remote Sensing*, 35(5):1307–1315, Sept. 1997.

[21] A. Visa, K. Valkealahti, and O. Simula. Cloud detection based on texture segmentation by neural network methods. In *International Joint Conference on Neural Networks*, volume 2, pages 1001–1006, Singapore, Nov. 18–21 1991.

[22] I. Yläkoski and A. Visa. A two-stage classifier for wood boards. In *Proceedings of The 8th Scandinavian Conference on Image Analysis*, pages 637–641, Tromsoe, Norway, May 25–28 1993.