DESIGNING A DEFECT CLASSIFICATION SYSTEM: A CASE STUDY

D. BRZAKOVIC and N. VUJOVIC
Department of Electrical Engineering and Computer Science, Lehigh University, Bethlehem, PA 18015, U.S.A.

(Received 8 July 1994; in revised form 1 August 1995; received for publication 5 December 1995)

Abstract -- This paper describes designing and testing a vision system for web material inspection. The system incorporates five subsystems: sensing, detection, characterization, feature analysis and classification. Each subsystem contains adaptive image processing and pattern recognition algorithms that perform specific functions. A very difficult real-world case study of defect classification is detailed and examples of algorithms used by the five subsystems are outlined. The results are summarized and interpreted. Copyright © 1996 Pattern Recognition Society. Published by Elsevier Science Ltd.

I. INTRODUCTION

Quality control is the most important part of today's highly competitive industrial production. The high cost of manual inspection has prompted the development of on-line computer-based systems capable of performing inspection tasks. In addition to lowering overall manufacturing costs, such systems offer more consistent performance than manual operations. Most of the inspection systems presently in use perform defect detection; however, defect classification remains a research subject (see e.g. surveys by Newman and Jain,1 and Mothadi and Sanz,2 and work by Han et al.3). In many cases practical solutions to the inspection problems call for pairing human and machine vision (Sylla4). Particularly difficult defect classification problems are associated with web materials,* because of the required processing speeds and variability of inspection criteria (for comprehensive discussion see Guary5). The research issues in web inspection are widely open, and, in the absence of appropriate algorithms, development concentrates mostly on hardware, in particular sensory systems and host processors. Most frequently, sensory systems use CCD chips (e.g. Mills6); however, alternative strategies, including laser scanning (e.g. Neukermans7) and ultrasound systems (e.g. Zmola et al.8) are gaining ground. The selection of host processors varies from general purpose computers (e.g. Roberts et al.9) to customized hardware (e.g. Frenyi and Pongracz10).

This paper concentrates on research issues related to defect classification in web materials and discusses the general methodology as well as specific examples of the algorithms. The proposed inspection system consists of five interconnected subsystems: (i) sensing, (ii) defect detection, (iii) defect characterization, (iv) feature analysis and (v) classification. Each subsystem, in turn, consists of a number of algorithms. An algorithm is evaluated as an integral part of the whole system. The subsystems are linked by feed-forward and feed-back loops. The paper focuses on the interaction between the subsystems and, in particular, emphasizes the significance of the sensing subsystem, which can simplify the tasks of the other subsystems. A very difficult real-world case study is used to illustrate the specific algorithms. The case study involves five classes where the defect class membership is determined by destructive testing. The objective of our study is to evaluate the feasibility of replacing destructive off-line tests by on-line noncontact sensors and computer-based systems. It should be noted that the proposed inspection system and algorithms are general and applicable to other inspection problems.

The paper is organized as follows. The requirements for web inspection systems are summarized in Section 2. The proposed inspection system is described in Section 3, while the specific sensing strategies and defect characterization algorithms are detailed in Section 4. Section 5 discusses the performance of these algorithms based on the analysis in the feature space and summarizes the results.

2. ISSUES IN WEB MATERIAL INSPECTION

Web materials take many forms; however, there is a remarkable similarity in the requirements for web inspection technology which cut across the major industrial segments. Typically, materials are homogeneous and discrepancies from homogeneity are interpreted as flaws. Web material inspection primarily

* The term web materials refers to the materials produced in the form of continuous rolls. Web processing is used in many segments of industry, e.g. metals, papers, plastics and textiles.
relies on 2-D (two-dimensional) image understanding in contrast to parts inspection, which is inherently a 3-D image understanding problem.

Based on generic characteristics, web materials can be categorized into uniform and texture materials. In both categories inspection is presently performed either subjectively (by visual inspection) or objectively by destructive or nondestructive testing. Examples of uniform materials include metals, films, paper and various plastics. Defects in these materials are easily differentiated from the background, as seen in Fig. 1, and development of defect classification capabilities is currently the highest technical priority. Texture web materials can be divided into regular textures (e.g. printed textiles, printed currency) and random textures [e.g. nonwoven materials such as those shown in Fig. 1(b)]. In regular textures any discrepancy from a pre-determined pattern is viewed as a defect. Frequently, these problems require color processing with objective to evaluate color uniformity and consistency, which is essential in many materials, e.g. wall paper. Random texture materials require grading the overall and/or local texture quality. In some cases the quality of these materials is evaluated through performance tests, e.g. by measuring pressure drop across filter surface. Replacing performance tests by monitoring inputs from noncontact sensors is, in general, very difficult because detailed understanding of the relationships between defect characteristics, material microstructure and material performance must be achieved in order to determine proper sensing objectives and strategies. In this paper attention is on uniform web materials and consequently the following discussion is limited to inspection issues related to these materials.

Fig. 1. Examples of two types of web materials: (a) uniform web material, 64 defects belonging to the same class; (b) texture material, two samples that show different performances in objective tests.
2.1. Issues in uniform web material inspection

At present, there are commercially available systems that can perform defect detection in uniform web materials at a reasonable cost. However, the problem of defect classification, i.e. determination of defect origin, is still an open research issue. The major obstacles in solving the classification problem are the following:

- **Extremely high data throughput.** A typical web material is 1–3 m wide and moves with speeds ranging from 200 to 2000 m min⁻¹. Consequently, data throughput for 100% inspection (when classifying defects of mm size) is tremendous and cannot be handled by the present general purpose hardware.

- **Inter-class similarity and intra-class diversity.** A single class of defects may vary widely in appearance and may have members that closely resemble defects in other classes. Therefore, the structure of a given class in a feature space may be of a very complex nature.

- **Large number of classes.** A typical defect classification problem involves a large number of defect classes; it is not unusual to deal with a few dozen to a few hundred classes.

- **Dynamic defect populations.** Small changes in the production process can result in entirely new classes of defects and a useful classification system should be dynamic with the ability for continuous on-line learning.

The first three items make initial system design very difficult, while the fourth item is a major obstacle in extending the useful lifespan of a developed system.

In order to illustrate data rates expected in web material inspection we consider a specific case. First, we analyse the problem of appropriate rate of data acquisition. For example, consider a 2 m wide web material which is moving at speed of 300 m min⁻¹ and defects of width 1 mm (both in horizontal and vertical direction). Assuming that a defect must be represented by a minimum of 2 pixels in both horizontal and vertical directions (i.e. spatial resolution of 0.5 mm pixel⁻¹), it is necessary to place eight Charge Coupled Device (CCD) cameras operating at 512 x 512 pixels to cover the cross web direction. In order to keep up with the moving web, which travels at rate of 5 m s⁻¹, it is necessary to acquire 20 frames s⁻¹. The defect detection subsystem receives in total \(8 \times 512 \times 512 \times 20 \approx 40 \times 10^6\) pixels s⁻¹.

Next we consider the problem of data processing and defect classification. The ultimate objective of an inspection system is to process data in real time; however, the concept of real time may be understood differently, depending on the specific task. Generally, it implies that the processing can keep up with data acquisition and material manufacture. The simplest task in web material inspection is recording the position of each defect within the web map, so that this position may be taken into account at a later time, e.g. when cutting material. In this case, it is sufficient that the processing rate be such that no major bottleneck be created by it; it may be adequate that the inspection cycle takes part of a second (e.g. 1/10 of a second). Recording defect locations calls for defect detection, which in turn can be achieved by thresholding. In simple cases this operation can be performed at
discussed data rates using off-the-shelf hardware; examples of systems performing real-time thresholding are discussed by Batchelor and Braggins[11] and Waltz.[12]

The previous considerations indicate that the defect detection problem can be solved in real time by using the available hardware and at a reasonable cost. Let us now consider the inspection system requirements for the defect classification problem on the same web material. In order to obtain useful shape and texture information required by many classification problems it is necessary to represent a defect by more than 2 pixels in each direction. When requiring defect representation by at least 20 pixels in each direction, the required acquisition rate increases 100 times. For 100% inspection in this case, 80 cameras working at the rate of at least 200 frames s$^{-1}$ are required. We point out that such cameras are readily available at a considerable cost (present state of technology is discussed by Giblom[13]).

3. INSPECTION SYSTEM DESIGN AND EVALUATION: GENERAL CONSIDERATIONS

In order to provide for designing and testing of various inspection systems we consider a development environment consisting of five subsystems: Sensing, Detection, Characterization, Feature Analysis and Classification. An efficient solution to an inspection problem requires a near-optimal design of each of the subsystems, as well as integration of proper control mechanisms connecting these subsystems. An inspection system must be evaluated as a whole based on its performance, which includes the accuracy rate of defect detection and recognition and the rate of false alarms. The additional evaluation criteria are system speed, cost and versatility.

In this work we simulate the five subsystems and their interaction in software. Each of the subsystems is represented by a number of algorithms/programs and for each particular problem we consider different selection of algorithms with the objective of selecting the most appropriate one. In the design stage, the five subsystems are connected by feed-forward and feedback loops, as shown in Fig. 2. In the test phase, the detection, characterization and classification subsystems are usually connected by feed-forward connections only. The sensing subsystem is in the test phase connected by both feed-forward and feed-back loops to any or all three sub-systems. Feed-back loops are used when the sensing subsystem is triggered to provide more data or select a different sensing strategy. In the rest of this section the functions of five subsystems and their present software configurations are described. Specific examples of algorithms are detailed in Sections 4 and 5.

3.1. Sensing subsystem

This subsystem integrates various lighting, optics and sensing alternatives in order to allow for selection of the most appropriate acquisition strategy. In general, proper selection of sensing strategy guarantees simplification of detection and classification problems. In many cases, it is necessary to use multiple sensors and fuse acquired data which generally increases hardware cost but reduces the development cost of detection and classification subsystems and the overall cost of inspection. Unfortunately, the sensing subsystem receives the least attention in both industrial and academic environments even though sometimes it offers elegant and cost-efficient solutions to otherwise unsolvable problems (an example of such a solution is discussed in this paper).

![INSPECTION SYSTEM](image)

Fig. 2. Interaction of five subsystems in the design and test phases. Typically, the Feature Analysis subsystem is included only in the design stage.
A particularly important component of the sensing subsystem is illumination; when coupled with appropriate sensors, adequate illumination offers very efficient solutions to difficult problems of image acquisition. In many cases special types of sensors such as laser scanners may be more appropriate than ordinary CCD cameras. These sensors provide high resolution and are more suitable in some inspection tasks, but are very expensive. Issues related to sensor selection and 63 useful ways of illuminating objects are reviewed by Batchelor.\(^\text{14}\)

The software simulation of the sensing subsystem in the most general form includes models of material reflectance, light sources and sensors. Such simulations are helpful even in cases when there is readily available hardware, because they offer safe and cost-efficient evaluations prior to hardware tests. The subsystem used in this work is limited to modeling sensors, in particular, their resolution capabilities and sensor-based geometric transformation. The algorithms include multiresolution pyramids, column-row integrations (representation of image arrays by 1-D signatures), space-variant sensing and cortical projection.

### 3.2. Detection subsystem

The role of this subsystem is to detect defects and alert the characterization and/or sensing subsystems that an irregular event has been encountered. Typically, this subsystem works with the lowest resolution which permits defect detection. Many of the commercially available inspection systems perform exclusively the task of this subsystem (see Batchelor and Braggins\(^\text{111}\)).

In general, solutions to detect detection are task dependent and may involve thresholding and/or filtering (e.g. inspection of textiles proposed by Norton-Wayne \textit{et al.}\(^\text{113,114}\), texture characterization (e.g. Brzakovic \textit{et al.}\(^\text{116}\) or comparison with the existing model (e.g. currency inspection involving template matching Tobin\(^\text{117}\)).

The software simulation discussed in this work relies on various local and global thresholding techniques based on optimal threshold selection proposed by Otsu.\(^\text{118}\) In addition, it includes model-based threshold algorithms, which incorporate modeling normal intensity distributions, e.g. linear distribution with superimposed random noise and adaptive detection of discrepancies from the model.

### 3.3. Characterization subsystem

This subsystem performs feature extraction and the formation of feature pattern vectors that characterize defects. The objective is to represent defects by as few as possible features, while achieving good discrimination between defects belonging to different classes. The complexity of the tasks performed by this subsystem is determined by specific problems. The most fundamental issue is extraction of features that are invariant to expected deformations, e.g. defect position, orientation or size. Methodologically, this subsystem employs low-level and intermediate-level vision algorithms.

The software simulation developed in this work contains shape description methods, including various signatures (e.g. directional intensity signatures), transforms (including Hough and wavelet transforms, and Karhunen–Loeve expansion) and texture representation methods (e.g. Laws' measures and analysis of local oriented edges). In addition, this subsystem contains neural networks that perform feature extraction, including Kohonen self-organizing map.

### 3.4. Feature analysis subsystem

The feature analysis subsystem evaluates the effectiveness of the characterization subsystem and aids in designing/selecting classifier architecture. The primary role of the subsystem is to help understand the distribution of pattern vectors from various classes in the feature space, that is, to establish if the pattern vectors belonging to a class form clusters and if there is significant overlap between the classes. In cases when classes are well separated, this subsystem searches for further reduction of feature space dimensionality. An important part of this subsystem is visualization of pattern hyperspaces using interactive graphics and various mapping techniques, including neural networks.

The major components of the software implementation of this system include clustering (based on K-means and ISODATA algorithms), combinations of clustering and hierarchical structures for reducing the complexity of the analysis task, feature selection (using minimization of entropy and Karhunen–Loeve expansion) and neural-network approaches (ART2 and Kohonen self-organizing maps).

### 3.5. Classification subsystem

A classifier labels defects. Typically, it requires a learning stage in which it is supplied by input from the characterization subsystem and corresponding class label. The classifier design/selection follows from feature analysis; the complexity of feature vector distribution dictates selection of a specific classifier strategy, e.g. multilayer neural network is appropriate when feature pattern vectors form clusters of arbitrary shape while single-layer, single-node classifier suffices in the case of linearly separable classes. Further considerations in selecting the classifier are the amount of data available for learning and system verification, and future dynamics of defect population, including new trends within existing classes, as well as the introduction of new classes into the feature space. Advanced issues include real-time learning and novelty detection.

The software simulation used in this work contains adaptive algorithms based on connectionist, Bayesian and fuzzy-set theory approaches. The most commonly used algorithms include feed-forward perceptron, k-
nearest neighbor, Parzen windows, counter-propagation neural network, ART2 and radial basis neural network.

4. CASE STUDY: SENSING AND CHARACTERIZATION SUBSYSTEMS

In the following we consider a case study emphasizing the problem of defect classification. The focus is on solving the problems of inter-class similarity and intra-class diversity and providing a solution that can be implemented near real-time using presently available hardware. The study considers a problem where, at present, defect classification is determined by objective destructive testing. The reliability of the physical test is such that the true class membership is known with probability of ca 0.85. No solution to the studied problem is known at this time and our objective is to determine the feasibility of automating defect classification in this case.

The case study concentrates on a web material that is typically produced in long sheets of width 0.8 to 1.2 m, moving at speeds in the range 200–600 m min$^{-1}$. The defect size is of the order 1 mm. The study considers only some aspects of problem encountered in practice, and concentrates on five defect classes. The most concerning problem is considerable overlap between classes in the visual appearance of the defects. This point is illustrated in Fig. 3, where a portion of the samples belonging to different classes is shown; each

![Fig. 3](image)

*Fig. 3. Examples illustrating inter-class diversity and intra-class similarity in the case study. Each class is represented by up to 64 defects. Attention should be paid to variability in the visual appearance of defects in each of the classes and to the similarity between some of the defects in different classes.*
class is represented by a subset of up to 64 samples. It is also important to note that the class representatives are unevenly distributed over the classes; e.g. class 2 has 155 representatives, and at the other extreme, class 3 is represented by only 47 samples. Such an uneven membership poses fundamental theoretical and practical problems. The class membership is listed in Table 1.

Two approaches, based on the sensing strategy, are considered for solving the defect classification problem in the case study. The first approach involves conventional sensing and the emphasis of the work is put on selecting appropriate characterization algorithms so that different classes show separability in the feature space. In the second approach, the emphasis is put on changing the sensing strategy in order to simplify the task of the characterization subsystem. In Sections 4.1 and 4.2 we discuss the specific algorithms used by the sensing and characterization modules and then in Section 5 we discuss the results based on the analysis in the feature space.

### 4.1. Approach 1: Conventional sensing

This approach uses conventional imaging techniques in which a web material is sampled uniformly

<table>
<thead>
<tr>
<th>Class</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>79</td>
<td>155</td>
<td>47</td>
<td>104</td>
<td>74</td>
</tr>
</tbody>
</table>
and data are represented in standard matrix form. The five subsystems are connected in a feed-forward manner and the characterization subsystem bears the burden of the inspection task. Therefore, further discussion concentrates on the characterization subsystem.

4.1.1. Characterization subsystem. We have considered various approaches to characterizing defects for this case study. Generally, standard shape descriptors were of little use because there was no consistency in obtained values within a class and even more importantly, such descriptors showed sensitivity to noise and, thus, were impractical to use. We have studied in detail three characterization algorithms: invariant moments (based on the work of Hu(19)), spatial-domain intensity signatures and wavelet-based signatures (Sari-Sarraf and Brzakovic(20)). None of the three approaches clustered data consistently with the classification objectives (Brzakovic et al.(21)). In the following we detail the spatial-domain intensity signature approach, because it generated the most interesting results, i.e. grouped defects in the feature space similar to the way humans would, based on the visual content of images.

The spatial-domain signatures were motivated by our desire to capture the shape, orientation and intensity variations of a defect simultaneously. For an image, \( f(x, y) \), an element \( s_b \) of the signature, \( S \), is defined as:

\[
S_{sb} = \frac{1}{K} \sum_{r=1}^{K} f(x_0 + r \cos \theta_b, y_0 + r \sin \theta_b),
\]

where \((x_0, y_0)\) is the center of gravity, angle \(\theta_b\) varies in predetermined steps, \(\delta\), and \(K\) is determined by the physical edges of the considered image region. An \(n\) element signature is defined as \(S = (s_0, s_{\theta_1}, s_{\theta_2}, \ldots, s_{\theta_{n-1}})\), where \(\Delta = 360\). We have used \(\Delta = 5\) in our experiments, thus acquiring 72 element pattern vectors. It should be noted that the backgrounds were normalized prior to signature generation in order to countereffect differences in image acquisition conditions. Examples of three signatures that reflect shape characteristics of defects are shown in Fig. 4.
Our next objective was to reduce the dimensionality of pattern vectors by extracting specific features from the signatures. However, the signatures did not show directly distinct features that could be used in further data reduction. Thus, we have subjected the entire signature to the feature analysis subsystem in the hope that the feature subset selection can be performed using standard methods. This analysis is described in Section 5.

4.2. Approach 2: Smart sensing

This approach illustrates the importance of sensors and shows that a change in the sensing strategy can facilitate and improve inspection system performance. The issues regarding sensing strategies and, in particular, “smart sensing” have yet to capture the attention of both the research community and practitioners in the web inspection industry.

The overall architecture of the web inspection system in this case incorporates the five subsystems connected by feed-forward and the feed-back loop between the detection and sensing subsystems. Furthermore, the sensing subsystem provides an input directly to the characterization subsystem when alerted by the feed-back loop.

4.2.1. Sensing subsystem. In the proposed approach the sensing strategy is determined by the defect detection subsystem. Similar to target tracking systems, the inspection system acquires a broad view (low-resolution image) and based on that determines the presence of a potential defect. Defect detection is performed using an adaptive thresholding algorithm,
described by Otsu. \cite{otsu1979thresholding} This is a robust adaptive algorithm that allows detection of an object (a bright blob) on a noisy background. The presence of a potential defect alerts the sensing subsystem through the feedback loop. Then, the space-variant sensor foveates on the region specified by the detection subsystem to provide a detailed view of a potential defect. The switching of the sensing strategy is motivated by the desire to reduce data throughput and make the on-line web inspection feasible. Furthermore, the space-variant sensing, combined with the cortical projection, simplifies the task of the characterization subsystem when an irregular event has been encountered.

The space-variant strategy simulated in this work is based on the concept of foveal vision and sensor description by Massone \textit{et al}. \cite{massone2010holographic} and Sandini and Dario. \cite{sandini2004conversion} The fovea is placed over the center of mass of the detected defect and is combined with cortical projection to provide an input to the characterization subsystem. The basic topology of the sensor is shown in Fig. 5(a). The high-resolution fovea is placed at the sensor center and is surrounded by three concentric layers. Each of the layers comprises $M$ concentric rows and in Fig. 5(b) the rows of layer 3 are shown with different shading for $M = 5$. In this study we have simulated sensor with $M = 16$. Sensing elements (cells) are distributed over each row and three such elements for the three layers are shown in Fig. 5(c). Different shapes of the sensing elements are proposed in the literature, but the common characteristics for each implementation are:

- the integration area increases for the cells that are farther away from the sensor center, that is, the resolv-
Designing a defect classification system

Fig. 4. Examples of signatures acquired in the spatial domain: (a) and (b) large horizontal defects, and (c) small, fairly faint defects.

ing capability of the sensor is higher for cells closer to the sensor center and decreases for peripheral cells;

- the highest resolving capability is in the sensor center (fovea);
- angular resolution is the same for each layer and each row inside a layer.

This sensor organization is particularly suitable for obtaining a cortical projection of an image, a geometrical transformation associated with the human visual system. Cortical projection simplifies effects of some distortions, e.g. rotation and scaling, which is very important for the class of inspection problems considered in this study. In addition, some features are very easy to extract in cortical images since their extraction requires integration along only one of the coordinate axes.

Cortical projection. The retino-cortical projection is a geometrical transformation and can be described by a conformal mapping of the polar \((\rho, \theta)\) plane onto the Cartesian \((\log(\rho), \theta)\) plane. This relationship can mathematically be described by considering a point \(z(\rho, \theta) = \rho e^{j\theta}\) in the retinal plane that maps to the point \(w(u, v) = u + jv\) in the cortical plane as:

\[
w = \log(z) = \log(\rho) + j(\theta + 2\pi k) \tag{2}
\]

and \(k\) is an integer. When taking into account only the principal branch of the logarithmic function, the representation of a cortical projection point is:

\[
u = \log(\rho), \quad v = \theta. \tag{3}
\]

Cortical projection images are used for feature extraction in this study, because they emphasize some important features of web defects that are not easily visible in the images acquired by conventional sensors.* The obvious advantages of this transform regarding computer vision applications, such as shape recognition in binary images, have already been demonstrated by Sandini and Dario.\(^{23}\) Our work considers gray-level images, where the objects vary in shape and size. In this particular case, the transformation offers the following advantages:

- It allows reduction of the effects of defect size. In particular, it allows disregarding the central part of the defect if it does not exhibit significant intensity variations. This, in turn, allows the size-invariant representation of particular types of defects. Disregarding the central part of the object when using conventional sensors is of no practical use, because it has no effect on size variations. In contrast, as seen in Fig. 8, disregarding central parts of objects yields identical representations in objects of varying size.
- The orientation of an elongated bright defect translates into the presence of a bright, narrow ridge

*The term conventional sensors refers to sensors with uniform spatial resolution and matrix organization of sensing elements.
Fig. 5. Smart sensor. (a) Basic structure: the sensor comprises high-resolution central fovea and three concentric layers. (b) Structure of a layer: each of the layers comprises several circular rows (in this case five circular rows are shown with different shading for layer 3). (c) Sensing elements: each circular row contains the same number of sensor elements (cells), that is, the angular resolution of each row is the same. However, the overall resolving capability decreases for rows that are farther away from the sensor center.

along the $\theta$ axis and the position of the ridge reveals the specific orientation. Therefore, by integrating the columns of the transformed image and by detecting the location of the peaks it is possible to determine the defect orientation. In contrast, a conventional image requires at least integrating both columns and rows, and in small objects of arbitrary orientation the two signatures may not reveal defect orientation.

4.2.2. Characterization subsystem. Defect characterization was carried out on thresholded cortical images. The thresholding was performed by a version of the optimum thresholding algorithm proposed by Otsu (18) that allowed separating an image into three regions: the background, bright object and any other regions that are darker than the background (such regions do exist in these images but are not apparent to the human eye due to low contrast). The relationship between hardly visible dark regions and easily visible bright regions is the dominant intra-class characteristic. It should be noted that we were not aware of this relationship before studying the cortical projection images. Examples of the thresholded images used for feature extraction are shown in Fig. 7(b) [correspond-
Fig. 6. Geometrical objects (upper row) and their cortical projections (lower row). Notice that scaling [(a) and (b)] and rotation [(a) and (c)] effects are shifts along the coordinate axes.

Fig. 7. Part of defects in class 2: (a) cortical projections of examples in Figure 3, (b) thresholded images in (b) using two threshold values, thus separating dark and bright portions of the objects and the background.
Feature extraction is divided into the characterization of bright and dark regions. In both cases, three features are extracted and together they form a six-element pattern vector that is used for classification. The features are as follows:

- Feature 1: the maximum $\rho$ coordinate of the bright/dark pixels [see Fig. 9(a)]. This measurement relates to the shape of the defect in normalized thresholded images, i.e., it relates to "roundness" of the object.
- Feature 2: object orientation measured at the location of a bright/dark ridge [see Fig. 9(b)].
- Feature 3: 10-point width of the least mean-square error (LMSE) fit. This feature is related to overall object shape. The width of the fitting function is

![Graphical representation of extraction of features](image)

Fig. 9. Graphical representation of extraction of features: (a) extraction of Feature 1, (b) extraction of Feature 2 and (c) extraction of Feature 3. Features for the dark and bright portions of images are extracted similarly.
Table 2. First-level clustering results indicating high overlap between the classes when using Approach 1

<table>
<thead>
<tr>
<th>Cluster</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>4</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>2</td>
<td>10</td>
<td>3</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>L</td>
<td>2</td>
<td>7</td>
<td>1</td>
<td>27</td>
<td>2</td>
<td>16</td>
<td>1</td>
<td>6</td>
<td>10</td>
<td>4</td>
<td>0</td>
<td>2</td>
<td>5</td>
<td>12</td>
<td>0</td>
<td>6</td>
<td>30</td>
<td>7</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>A</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>28</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>42</td>
<td>2</td>
<td>7</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>18</td>
<td>3</td>
<td>0</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>S</td>
<td>5</td>
<td>1</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>13</td>
<td>25</td>
<td>10</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

measured on the line that is parallel to the tangential line at the extremum point and is 10 units away from it [see Fig. 9(c)]. The convex and concave fits are labeled by the opposite signs.

5. ANALYSIS IN THE FEATURE SPACE AND DESIGN OF THE CLASSIFIER

The goal of the characterization subsystem is to compress information into feature pattern vectors which can then be analysed with respect to the classification goal in order to select a proper classifier architecture. Considering that each of the classes is represented by a relatively small number of samples, the standard statistics are only marginally useful. Generally, the basic statistics show that there are only a few clear outliers in the entire population of defects in both approaches and further analysis has proceeded with the complete data set. This analysis included clustering and data visualization. We have performed an interactive search for the optimal number of clusters, inherent in the data, based on the K-means and ISODATA algorithms. The results of this analysis are as follows:

**Approach 1.** A coarse, initial clustering was performed in order to determine the global characteristics of data that may allow for further clustering. In this case 20 clusters represented well the full data set. There was a mixed class representation in the majority of the clusters. A further search on individual classes revealed that some of the classes were distributed over as many as 18 clusters. Table 2 shows the distribution of class population among the obtained clusters. The defects associated with a cluster were examined for visual homogeneity, showing an immediate separation of the defects into large, bright defects and small, low-contrast defects. Thus, at the first pass, there appeared to be the necessary relationship between the characterization and the visual characteristics of the defects. The next step in the analysis was to determine the amount of resolution necessary to separate the existing clusters into more homogeneous groups. The technique utilized for this was a visually guided variant of the ISODATA clustering algorithm. The general strategy was to incorporate the ISODATA features of cluster deletion, splitting and merging guided by the visual information in the images associated with the clustered vectors. The obtained clusters contained fairly homogeneous populations only when reduced to a very small size containing 1–4 elements. The essence of the problem in this case is that the pattern vectors have
high dimensionality and no obvious features could be extracted from the signatures. Consequently, the high dimensionality of the feature space forced sparse class representations. However, it should be pointed out that the initial clustering results show high correlation with human perception. As illustrated in Fig. 10, the clusters group defects based on their size and shape.

Approach 2. In this case 10 clusters represented well the full data set. Each class was represented by two clusters and each of the clusters corresponded well to a single class. Therefore, this analysis suggested that each of the classes was represented by few compact clusters and that no significant overlap between the classes existed. The same result was confirmed through visualization techniques. Furthermore, it was indicated that the reasonable class separation between any two classes (for a two class subproblem) can be achieved by using up to two features, e.g., features 1 and 2 extracted for the bright regions when separating classes 1 and 2. This point is illustrated in Fig. 11, where the feature pattern vectors representing classes 1 and 2 are shown in the 3-D feature space.

Since there was no need to design a special type of classifier, we have experimented with the existing classifiers, including perceptron (generating quadratic boundaries), Bayesian classifier and self-organizing map network. Different classifiers performed consistently, yielding on average 85% correct classification. The overall system performance was evaluated in the
Designing a defect classification system 1417

Patterns In Feature Space

\[ \text{Patterns In Feature Space} \]

(a) 3-D space and (b) projection onto 2-D space (top view). The second view shows clear separation between the two classes.

Fig. 11. Two views of the 3-D feature space for a two-class subproblem using Approach 2: (a) 3-D space and (b) projection onto 2-D space (top view). The second view shows clear separation between the two classes.

following way. The classification was obtained by using 90% of the class members (randomly chosen) for training and the remaining 10% for testing. The experiments were repeated for different configurations of the training and test populations, and the results were averaged over 10 trials.

In summary, in Approach 1 the feature analysis revealed that it was impractical to attempt building the classifier. In contrast, the same analysis has shown that the classification task can be achieved in Approach 2, yielding accuracy of classification consistent with reliability of destructive tests.

5.1. Practical issues in using Approach 2 on-line

Considering that Approach 2 yields promising results, this section reviews its processing requirements. In Fig. 12 we show the stages involved in the interactions between sensing, detection and characterization subsystems. First, the conventional sensor acquires low-resolution data [Fig. 12(a)], which are then subjected to thresholding performed by the detection subsystem [Fig. 12(b)]. The presence of a defect triggers alternative sensing strategy. One approach to handling this is acquiring a high resolution image using a conventional sensor [Fig. 12(c)] and then simulating in software the smart sensor [Fig. 12(d)] and thresholding the resulting image [Fig. 12(e)]. Alternatively, steps shown in Figs 12(c) and (d) can be implemented directly in hardware. The software implementation offers an advantage in that it guarantees that the transformation will be generated relative to the object’s center of gravity.

Assuming 1 m wide web moving at 400 m min\(^{-1}\) and spatial resolution 0.5 mm pixel\(^{-1}\) for detection, it is necessary to place four cameras operating at 512 \(\times\) 512 across the web and to acquire 25 frames s\(^{-1}\). Consequently, the detection system receives 25 \(\times\) 4 \(\times\) 512 \(\times\) 512 pixels s\(^{-1}\). In the simplest case, defect detection requires using a predetermined threshold and in more complex cases it is necessary to employ local or global adaptive thresholding. Simpler thresholding can be handled as an integral part of acquisition with new generation of sensors. Defect

Fig. 12. Detection and classification of defects. (a) A low-resolution image of the moving web with a defect; (b) defect from (a) detected by the optimum thresholding algorithm; (c) high resolution image of (a); (d) cortical projection image of the defect in (a); (e) thresholded image in (d), which is used for feature extraction. Images in (a) and (b) are enlarged four times for displaying purposes.
classification requires at least 10 times higher resolution; however, assuming that defects are rare events, the high-resolution images are acquired only occasionally and required processing may take as long as 1 s (or more) per defect. The classification part requires first obtaining the cortical projection. Two steps are required for this task: (i) it is necessary to determine center of defect's gravity and (ii) perform the transform. The first step can be accomplished during acquisition with new types of sensors and the second step can be implemented as a tabular lookup, because the relationships between coordinates in two representations is predetermined. Further processing requires thresholding the cortical image and feature extraction. Extraction of features 1 and 2 amounts to adding pixel values along columns and comparing their values, while feature 3 requires more complex least-squares fitting. Extraction of feature 3 can be approximated by simpler calculations utilizing the distance between two dominant peaks. The final step of the classification procedure requires labeling the defect by the classifier, which can be carried out very quickly, in particular when using neural networks. From this analysis it follows that two types of processing are performed, the detection and characterization/classification. Both types require relatively simple operations and thus can be implemented on relatively inexpensive hardware. The control aspects related to sensor switching strategies require keeping up with the moving web and controlling its speed.

6. DISCUSSION AND CONCLUSIONS

The problem of replacing off-line destructive testing by automated analysis and noncontact sensors is, in general, very difficult. The major difficulty is caused by the fact that sensed data do not show visual correlation to class membership. The approaches capitalizing on the direct utilization of well-known image processing and pattern recognition methods are of little use here because of the nature of the data. The results obtained by these methods strongly correlate to the visual appearance of defects, but there is little correlation to results of destructive tests. In contrast, changing the sensing strategy can significantly simplify the problem and improve results. We attribute the success of Approach 2 primarily to encoding information pertaining to dark regions, which carry particularly important information about the defect origin. It should be pointed out that the existence of small dark regions was not apparent in the spatial domain and was enhanced by the method of cortical projection.

REFERENCES

About the Author—DRAGANA BRZAKOVIC has received her B.Sc. in electrical engineering from the University of Belgrade, Yugoslavia, in 1976. The balance of her studies was completed at the University of Florida, Gainesville, where she received her M.E. and Ph.D. degrees in electrical engineering in 1979 and 1984, respectively. In the period 1984–1992 Dr Brzakovic has been working in the Department of Electrical and Computer Engineering at the University of Tennessee, Knoxville. Since September 1992 Dr Brzakovic has been an Associate Professor in the Department of Electrical Engineering and Computer Science at Lehigh University, where she is teaching courses in pattern recognition, image processing and computer vision. Her current research interests include the areas of image understanding, knowledge-based pattern recognition and applied artificial intelligence, specializing in texture analysis.

About the Author—NENAD VUJOVIC received his B.Sc. in electrical engineering from the University of Belgrade, Yugoslavia, in 1991, where he specialized in the area of telecommunications. After graduation he attended the University of Tennessee, where he received the M.S. degree in electrical engineering in 1992. He is now a Ph.D. student in the Department of Electrical Engineering and Computer Science at Lehigh University. His research interests are image processing and pattern recognition.