

Chemical Machine Vision: Automated Extraction of Chemical Metadata from Raster Images

Georgios V. Gkoutos,[†] Henry Rzepa,^{*,†} Richard M. Clark,[‡] Osei Adjei,[‡] and Harpal Johal[§]

Wolfson Laboratory for Informatics, Modeling and Visualization, Department of Chemistry, Imperial College of Science, Technology & Medicine, Exhibition Road, South Kensington, London, England SW7 2AY, Department of Computing and Information Systems, University of Luton, Park Square, Luton, Bedfordshire, England, LU1 3JU, and Ultra Electronics Control Division, Greenford, Middlesex, England, UB6 8UA

Received January 29, 2003

We present a novel application of machine vision methods for the identification of chemical composition diagrams from two-dimensional digital raster images. The method is based on the use of Gabor wavelets and an energy function to derive feature vectors from digital images. These are used for training and classification purposes using a Kohonen network for classification with the Euclidean distance norm. We compare this method with previous approaches to transforming such images to a molecular connection table, which are designed to achieve complete atom connection table fidelity but at the expense of requiring human interaction. The present texture-based approach is complementary in attempting to recognize higher order features such as the presence of a chemical representation in the original raster image. This information can be used for providing chemical metadata descriptors of the original image as part of a robot-based Internet resource discovery tool.

1. INTRODUCTION

The World Wide Web was designed as a common information space in which communication was implemented by sharing and exchange of text based information through a markup and linking language (HTML). In the first instance, there was a need for a presentational, human understandable, and readable language that was easy to construct and author, encouraging its general use. As a result its extension to a semantically meaningful markup in areas where complicated presentation was required was difficult. The solution arose with the introduction of graphical images through an early extension to HTML which was to allow transclusion of such objects, or images. Between 1987 and the early 1990s, the GIF (Graphics Interchange) Format¹ had become the most popular mechanism for archiving and exchanging computer raster images, and it became rapidly adopted as an alternative for marked up text-based content in HTML documents. More recent variations include formats such as JPEG (Joint Photographic Experts Group)² and PNG (Portable Network Graphics)³ etc. Data compression (lossless or lossy) was the main concern for all these graphic formats in order to minimize the increasing demand on bandwidth, rather than any attempt at creating machine processable semantics representing the meaning of the image.

The limitations of such approach were recognized later on, and modern vector formats emerged (e.g. SVG or scaleable vector graphics)⁴ where essentially lossless and bidirectional transformation between presentation (computer

screen, paper) and the underlying data model can be achieved. Vector images provide machine understandable information, and a search engine can in theory index and extract meta information for them.

However, the solemn presentational use, earlier on and still today, of raster images created using bitmap technologies resulted in the creation of a large amount of data containing no meta information relations to the semantic content they express. Although, such data are easily recognized and interpreted by trained humans, who are also able to read the descriptive information added to the images, these data could not be processed by machines, since no computer processable declarations of their content exists. A possible solution can arise from the addition and use of metadata.

Metadata can be broadly categorized⁵ as follows:

- **Navigational**

This is primarily to facilitate the location of associated documents and data.

- **Constraining**

This would be used to constrain the value and the behavior of the data.

- **Supplemental**

This adds information to the data, as for example the inclusion of an atom connection table as e.g. a SMILES representation to represent any molecular structures in an image.

- **Vocabulary**

This supplies “meaning” to the data; the metadata might refer to the context of the image (chemical structure, reaction diagram, instrumental diagram, etc.).

- **Function**

This indicates the function of the data and the possibility of binding appropriate software for processing the data.

* Corresponding author phone: +44 (0)20 594 5795; fax: +44 (0)20 594 5804; e-mail: h.rzepa@ic.ac.uk.

[†] Imperial College of Science, Technology & Medicine.

[‡] University of Luton.

[§] Ultra Electronics Control Division.

We have previously described ways of extracting metadata associated with some of the above categories by automated parsing of information from 2D and 3D molecule coordinate files,^{6,7} such as the MDL Molfile.⁸ However, in molecular sciences and particularly in chemistry, structural information available on the Web is too infrequently captured in such file formats. Indeed, we believe most such information lies in in-lined raster images. In this article, we briefly review four different, well-established ways of extracting information from raster images in the context of constructing automatic low-cost robot-based methods for traversing existing and “legacy-based” Web-based content. We will describe a preliminary application of chemical recognition making use of a technique known as “machine vision”, which has been tested in other nonchemical areas.

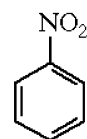
1.1. Current Methods for Abstracting Information from Raster Images. **1.2. Embedded Metadata in Raster Images.** The GIF and PNG image syntax includes invisible text fields that can carry metadata. Some online services and chemical structure editors can generate such images with included chemical descriptors such as SMILES strings, atom coordinates, and Molfile connection tables, and we have written parsers that can extract and perform added value operations on such images.⁹ Unfortunately, addition of such metadata to Web-server based chemical images has never been adopted to any significant extent.

An alternative is to use the `` invocation of an image file within an HTML document to add associated metadata via the element attributes. The `alt`="Descriptor" attribute is recommended for indicating the function of an image, whereas a formal description invoked as `longdesc`="URI" would point to the location of a long description of the image. This too has issues of adoption; the latter is very rarely used, while the former while more commonly adopted, is rarely meaningful. Furthermore, there is no easy way of validating the relationship between the “alt” or “longdesc” descriptor and the object so described. Indeed, the very “human readability” and reusability of HTML code often means that the “alt” descriptor is inherited from a previous use and left unedited.

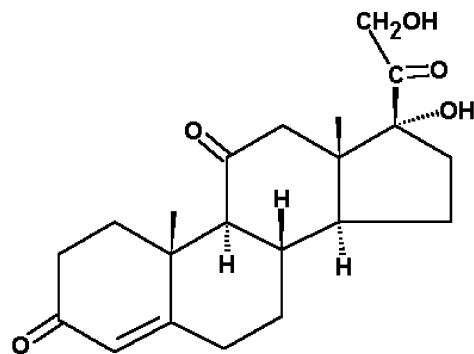
1.3. Kekule: Optical Chemical (Structure) Recognition. Transcribing a chemical structure in an appropriate software editor from a raster-based diagram is a costly and in human hands, highly error-prone process. Kekule¹⁰ is a software package developed in the early 1990s, which was designed to read scanned chemical-structure images and interpret them into a formal string representation. It is the chemical equivalent of an OCR (Optical Character Recognition) program. The system involves seven steps.⁵

- Scanning
- Vectorization
- Searching for dashed lines and dashed wedges
- Character recognition
- Graph compilation
- Postprocessing
- Display and editing

Character recognition (e.g. atom types) and graph compilation (bond linking) are two particular advantages of Kekule, but the system does require at least 300 dpi resolution scanned images (Scanning step) and several cycles of manual human intervention (postprocessing step) to achieve close to 100% accuracy in the transcription to e.g.



A. Nitrobenzene



B. Cortisone

Figure 1. An example of a simple and a more complex diagram, picked randomly from the Web, that both Kekule and CLiDE were tested with.

a Molfile.¹² The system was primarily designed for high-resolution images and the 72–96 dpi resolution normally associated with GIF raster images can significantly increase the human intervention process. The fully automatic generation of any chemical descriptor, whether at the level of achieving complete atom/bond accuracy, or indeed the coarser grained level of a molecule or class of molecule is for Web purposes, impossible to achieve. However, it should be noted that the Kekule program did surprisingly well, requiring only very little human intervention, for simple chemical diagrams (as judged by a human) (Figure 1 A) at 72–96 dpi retrieved randomly from the Web during our tests. More complicated images (Figure 1B), of lower quality, or those which were merely one component of a larger diagram, were difficult if not impossible to convert into sensible chemical descriptors.

1.4. CLiDE (Chemical Literature Data Extraction). CLiDE¹³ aims to automate the process of abstracting chemical information from the literature. It is sophisticated software that at its very basic level links chemical structures with the text segments of a publication via logical associations. It employs the Documental Format Description Language (DFDL) to describe the mutual relationships of logical objects and elements of the logical structure of a document. It uses advanced methods to identify the logical structure of a document, extract the features, and create logical descriptive relationships.¹⁴

As with Kekule, CLiDE also requires high-resolution images and human intervention to achieve sensible results and performed similarly with Kekule, while identical problems were recognized. The advanced methods employed prevented it from working automatically at the low level tasks useful for our purposes.

1.5. Raster to SVG Conversion. As noted earlier, SVG (Scalable Vector Graphics)⁴ diagrams are constructed from primitive shapes and paths rather than by describing pixel maps. SVG has many of the advantages associated with

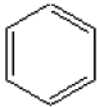
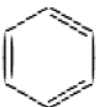
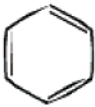



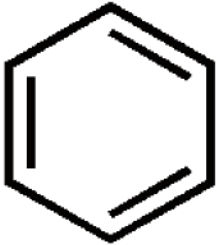
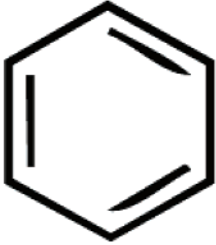
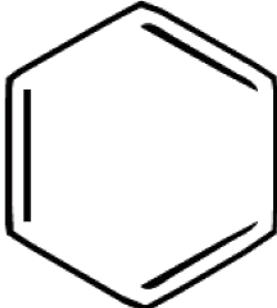
GIF Images	Celinea	Autotrace
A Benzene Ring  size: 1KB	CR2V conversion  size: 2KB paths: 86	Autotrace conversion  size: 1KB paths: 4
A Benzene Ring with wider lines  size: 2KB	CR2V conversion  size: 1KB paths: 4	Autotrace conversion  size: 1KB paths: 4
A bigger Benzene Ring with wider lines  size: 2KB	CR2V conversion  size: 1KB paths: 4	Autotrace conversion  size: 2KB paths: 4

Figure 2. CR2v and Autotrace conversion results.

HTML itself, including the ability to index the file, high-performance scaling, high-resolution printing, masking, animation, scripting, and linking. We chose to test two programs, which implement raster to SVG converters to test whether such R2V (Raster to Vector) preprocessing might prove a useful intermediate process.

CR2V. CR2V¹⁴ is a command line free program exporting to SVG and Adobe PostScript. It recognizes all major raster formats (BMP, GIF, JPEG, PNG, TIFF) and performs particular well with complex (nonsparse) images containing photos and colored graphics.

Autotrace. Autotrace¹⁵ is an OpenSource program for converting bitmap to vector graphics, which is optimized for conversion of images containing diagrams and lines, and hence is particularly suited for processing chemical structure images.

Both CR2V and in particular Autotrace perform well with vectorization of chemical images (Figure 2). A continuous chemical line diagram for example is mapped as a *path* under SVG. Benzene as cyclohexatriene will map to four paths, the basic ring hexagon and three double bond vectors. It should be possible, although not implemented here, to

further transform such paths to chemical structure recognition by creating for example a stylesheet able to recognize them and generate the corresponding added-value metatags, hence creating indexable and searchable fields for images.

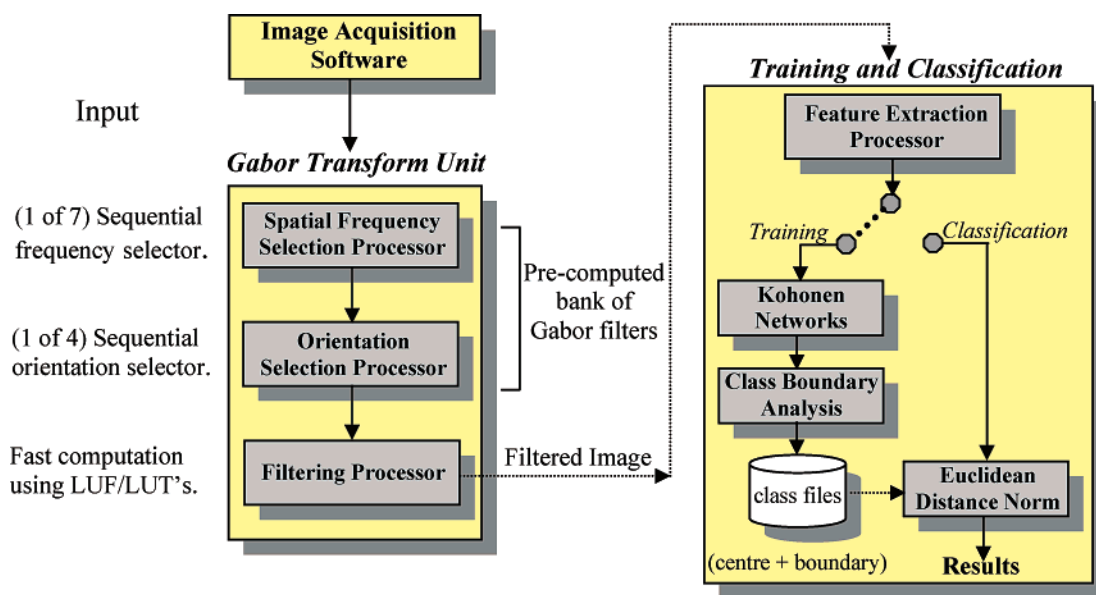
However, as the difficulty of such a process is expected to scale badly as the diagrams get more complex and when they are part of a more complex diagram, requiring OCR capabilities similar to the Kekule/CLiDE programs noted above (Figure 1) (Table 1).

1.6. Machine Vision. Machine Vision is a series of texture-based feature techniques that are employed to construct feature vectors composed from a number of features derived from a digital image. It has been applied to a number of areas for industrial, scientific, and medical texture recognition and discrimination problems. Examples of such application can be found in aerial image inspection,¹⁶ contour object detection,¹⁷ model based identification,¹⁸ military target estimation,¹⁹ face detection,²⁰ medical imaging,²¹ etc.

We have explored a system developed by Clark et al.²² with the intention of extracting general metadata from raster images in an automated way. Our intentions in the first stage

Table 1. Examples of More Complex Diagrams

chemical reactions	biological figures	protein figures
a reaction size: 7KB CR2V conversion size: 145KB size (compressed): 28KB Autotrace conversion size: 11KB	a protein figure size: 96KB CR2V conversion size (uncompressed): 145KB size (compressed): 53KB Autotrace conversion size: 4701KB size (compressed): 776KB	MHC complex size: 28KB CR2V conversion size: 108KB size (compressed): 30KB Autotrace conversion size: 1268KB size (compressed): 218KB
NMR spectra	1D NMR spectra	2D NMR spectra
NMR spectrum size: 5KB CR2V conversion size: 26KB size (compressed): 6KB Autotrace conversion size: 28KB size (compressed): 9KB	1D NMR spectrum size: 49KB CR2V conversion size: 37KB size (compressed): 12KB Autotrace conversion size: 129KB size (compressed): 53KB	2D NMR spectrum size: 115KB CR2V conversion (failed) size: size (compressed): Autotrace conversion size: 322KB size (compressed): 125KB

**Figure 3.** Design of a texture recognition system.

of this development is to provide a methodology able to provide answers to simple questions such as whether an image is likely to contain chemistry and furthermore distinguish those images to ones that contain ring systems. Our approach does not, by any means, replace the older methods described above. Instead it provides a methodology of extracting coarse chemical metadata without preprocessing the images, in a mostly automated way.

The machine vision system presented in this work uses a texture based feature extraction technique to construct feature vectors that are derived from digital images. The use of texture information in the form of chemical composition images overcomes the problems of obtaining a priori information about the context of each image. Although metadata can be encoded into some image formats (such as GIF), most images available regarding chemistry compositions do not have any associated structure information and would require expert intervention to define the semantic content of each image. Textures, however, provide measures of properties such as smoothness, coarseness, structure, and regularity,²³ and therefore a texture-feature based transformation, analysis, and discrimination methodology can be used

to recognize various groups of textures without the need for metadata or a priori information.

2. DESIGN OF THE TEXTURE RECOGNITION SYSTEM

The texture recognition system comprises two main units. These are the Gabor Transform unit and the Training and Classification unit as illustrated in Figure 3. A scripting method is used to control the units so that the location of the data and the texture class attributes can be identified during system training and classification.

The Gabor Transform unit accepts an input image in addition to performing three other software processes. The first two processes are used for spatial frequency and mask orientation selections. These processes operate together by using look-up tables to select a precomputed bank of even symmetric Gabor filters that consist of 64×64 masks with four orientations at 0, 45, 90, and 135 degrees, respectively.

The third is the filtering process that is responsible for the overall computation of the filtered image having selected the appropriate spatial frequency and the mask. In the filtering process, a transformation buffer is used to mount

the input image to allow the stage to eliminate unwanted artifacts that might appear at the edges of the resulting image from which the feature vector is computed.

The Training and Classification unit has three main functions. The first is to extract feature vectors from filtered images computed by the Gabor Transform unit. The second is to use the Kohonen Self-Organizing Feature Map (KSOFM) to compute cluster centers of feature vectors of textures belonging to same classes during training. The third involves the use of an Euclidean distance norm that is used to measure the distance of an unknown feature vector from the cluster centers during classification. The Euclidean distance norm is used in conjunction with the Class Boundary Analysis (CBA) processor. The class boundary analysis processor is used to compute a class boundary surrounding a cluster center beyond which a feature vector is rejected as not belonging to a particular class. In other words, the computation of class boundaries around class centers is useful in providing the system with an extra ability to discriminate.

The Training and Classification unit operates in two modes (i.e. training and classification modes). In the training mode, filtered images belonging to various classes of textural images are computed using the Feature Extraction Processor. These feature vectors are applied to the KSOFM and the CBA to compute the cluster centers and cluster boundaries. In the classification mode, an unknown textural image is applied to the Gabor Transform unit that subsequently computes a filtered form of the image. The filtered image is applied to the Training and Classification unit that computes the class to which the input image belongs using the class centers and class boundaries previously computed during the training mode. The class computation is achieved by using the Euclidean distance norm.

To control the operations of the system in an automated fashion, a system script is applied. The script describes the attributes of a given data set. These attributes include the locations of the digital image data, the number of texture classes in the set, the number of images per class, and the name of each class of textures. This information is used to automatically locate input data, control system operations, and derive intermediate output files. The script defines the data attributes using a scripted-language that is parsed by the machine vision system during operation. The ability to describe the attributes of each data set allows the system to be adapted for many applications without the need to modify internal system parameters.

3. THEORY AND IMPLEMENTATION OF GABOR WAVELETS

The receptive fields (RF) of the mammalian visual cortex and Gabor wavelets are localized in the spatial and frequency domains and consist of a sinusoidal wave that is modulated by a Gaussian envelope function.²⁴ The two-dimensional Gabor wavelet is given by Jain et al.²⁵

$$h(x,y) = \exp\left\{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\} \cos(2\pi\mu_0 x + \phi) \quad (1)$$

where σ_x and σ_y are the standard deviations of the Gaussian envelope along the x and y directions, respectively, μ_0 is the frequency and ϕ is the phase of the sinusoidal wave along

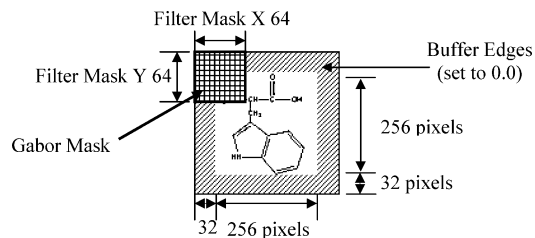


Figure 4. Mounting an image in a transformation buffer.

the x -direction (at 0° orientation). To create filters at various orientations the generated filter can be rotated by using two-dimensional rigid body transformation techniques. Alternatively, the filter may be computed by substituting the values of x, y from the following equations to directly compute a filter with a desired orientation:

$$x_\phi = x \cos \phi + y \sin \phi \quad (2)$$

$$y_\phi = -x \sin \phi + y \cos \phi \quad (3)$$

The first method of filter construction is preferred since it gives rise to fast computation of a single filter that can be duplicated and oriented at any angle when required. This method therefore saves recomputing a filter for every orientation. The filters are then stored in the system, and a *look-up-table* is used to load them when needed by the scripting process that controls this operation in the system.

With respect to the computation of Gabor wavelets, the following radial frequencies are used in the manner of Jain et al.²⁵ to form a bank of filters:

$$1\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}, 8\sqrt{2}, 16\sqrt{2}, 32\sqrt{2} \text{ and } 64\sqrt{2} \quad (4)$$

These computed frequencies are 1 octave apart, and with a selection of 4 orientations and 7 radial frequencies, a total of a bank of 28 Gabor filters are used for our experiments.

3.1. Enhancing the Transformation Stage. An application of the Gabor transform to several hundreds of images can prove to be a computationally expensive process.²⁶ A brief presentation of the methods developed to enhance the transform stage in order to reduce the computational time is discussed in this section. These methods consist of software programs that permit the use of a transformation buffer, look-up-frames (LUF), and look-up-tables (LUT) to select regions of interest in the transformed image for further processing.

3.2. Mounting Each Image in a Transformation Buffer. To avoid the introduction of unwanted artifacts at the image edges during transformation, a buffer mounting technique is used as shown in Figure 4. The buffer size is set so that at least half of the filter mask can overlap the edges of the image data. This allows the pixel values at the very edges of the image to be used by the filter mask during the transform operation. Using this technique, the image is mounted centrally in the transformation buffer, and the surrounding buffer edges are initialized to a minimum pixel value (i.e., set to 0). This technique ensures that the entire image data can be used in the transformation process.

3.3. Look-up-Frames and Look-up-Tables. In the selection of features, using a random number of feature points as discussed earlier, often there are areas of the original image that are not selected as part of the feature region. Therefore

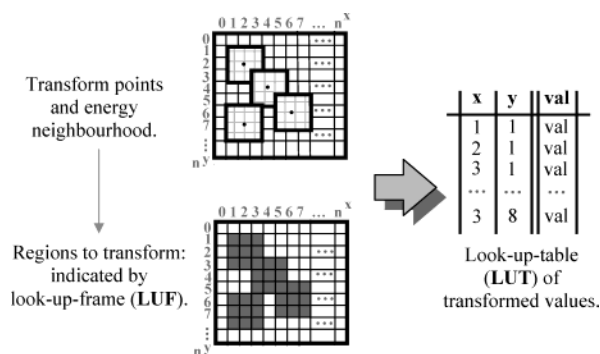


Figure 5. Method of using look-up-frame (LUF) and look-up-table (LUT).

such regions within the image are not regions of interest (ROI) and do not need to be transformed.

Figure 5 depicts the process where a number of feature regions are selected, and their corresponding neighbors are used by an energy function to compute the feature vectors. The neighboring pixels are determined by the size of the energy window. The selected pixel location and its neighbors can often result in overlapping regions of interest when deriving features as shown in Figure 5. It is also shown that some areas of an image may often not need to be computed as they are not selected to be regions from which to form an element of a feature vector.

A look-up-frame (LUF) and its corresponding look-up-tables (LUT) are applied to avoid the computation of areas in the image that are not used during the formation of a feature vector element. The transformation of an entire image, including the nonselected regions, can inhibit computation in real-time when transforming several hundreds of images.²² The two-dimensional look-up-frame is used to indicate the selected regions of interest (ROI) within an input image as shown in the example shown in Figure 5. These regions are appropriate for computing the elements of the feature vector.

To compute the regions of interest (ROI), a random selection of pixel locations is initially chosen. The neighboring pixels are then selected according to the size of the energy window that will be used by the energy function, later described in section 3.4. The energy function uses the values of the selected pixel and its neighbors in deriving an element of a feature vector. The required feature points and corresponding neighbors are computed, and the locations are stored in a two-dimensional binary LUF. In the LUF, a value of 1 denotes an area to transform in original image, and a value of 0 indicates that no transformation is required. Following the initialization of the LUF, only the pixels at the necessary (x,y) positions are transformed to reduce the time required during transformation. Information regarding the location and the transformed pixel values at these regions are correspondingly stored in the LUT for later use.

Figure 5 also demonstrates how there might be occasions whereby the energy windows overlap each other at several feature locations. In this situation, a pixel location that has already been transformed and stored in the LUT does not need to be recomputed. The pixel locations indicated in the LUF and the values stored in the LUT therefore help to identify areas of an image that require computation once only. The integration of a LUF and LUT further enhanced the computations during the transformation stage by remov-

ing the need to compute unnecessary regions if they have already been computed and stored in the LUT.

3.4. Computing Feature Vectors. After the transform operation, the image is thresholded. This process is based on a nonlinear function that resembles the processing of data used in sigmoidal activation functions as applied in artificial neural networks. The nonlinear function $\psi(t)$ is given in eq 5 as

$$\psi(t) = \tanh(\alpha t) = \frac{1 - e^{-2\alpha t}}{1 + e^{-2\alpha t}} \quad (5)$$

The function is used to highlight both light and dark regions of features from the transformed image. Previous studies by Jain et al.²⁵ found that a value of the constant $\alpha = 0.25$ resulted in a rapidly saturating threshold that enables the function to act as a detector of feature clusters.

In the feature extraction process, an energy function is applied to a number of random pixel locations. The pixel values are obtained from the *look-up-table* (as described in section 3.3). The energy function allows features to be computed as the average absolute deviation from the mean value. This is computed using small energy mask windows centered at each feature point. This operation therefore enables the system to form a measure of texture energy around each selected pixel value.

The energy function²⁷ is defined as

$$e_k(x, y) = \frac{1}{M^2} \sum_{(a,b) \in W_{xy}} |\psi(r_k(a, b))| \quad (6)$$

where $\psi(\cdot)$ is the nonlinear function given in (eq 5) and W_{xy} is a two-dimensional window of size $M \times M$ centered at pixel (x,y) in the threshold image. In experiments conducted in this work, the size of the energy window was modified as necessary with a smaller window giving energy readings of fine-scale features and a larger window giving larger-scale energy features. The energy function is used to compute the elements of the feature vector for each selected (x,y) region. Values for each transformed pixel location are obtained from the *look-up-table* (LUT). This process results in the output of a one-dimensional feature vector consisting of a selected number of feature elements.

4. CLASS BOUNDARY ANALYSIS (CBA)

The Class Boundary Analysis (CBA) process is used to compute the mean and variance of texture classes.²⁸ These metrics provide convenient means with which to measure classes in the incremental learning system. The mean of a class is the vector computed as the sum of all the elements in the class divided by the number of patterns in that class.

The mean of a class is denoted as

$$\mu_k = \frac{\sum_j^n x_j}{n} \quad \forall_k = 1, 2, 3, \dots, M \quad (7)$$

where x_j is the j th element in the feature vector, n is the number of patterns in the class, and k is the index to the elements in the vector, with M as the length of the vector.

The variance²⁹ is computed as

$$v_k = \frac{\sum_j^n (x_j - \mu_{kj})^2}{n - 1} \quad \forall_k = 1, 2, 3, \dots, M \quad (8)$$

where x_j is the j th element in the feature vector, n is the number of patterns in a training class, k is the index to the elements in the vector, and μ_{kj} is the mean of an element j of a class as computed in eq 7. In this work, the largest variance in the vector is used to provide the boundary limits of that class and can be used to aid the discrimination between classes with similar classification measurements.

4.1. Euclidean Distance Norm. The Euclidean distance norm is used to measure between the central point of a class of textural features and an unknown vector. In terms of the central point, suppose \mathbf{x}_i denotes a one-dimensional feature vector with real components where $\mathbf{x}_i = [x_{i1}, x_{i2}, x_{i3}, \dots, x_{in}]$, then \mathbf{x}_i defines a point in n -dimensional Euclidean space.

The Euclidean distance norm can therefore be used to identify the highest level of similarity between texture classes and an unknown vector. The Euclidean distance between \mathbf{x}_i and \mathbf{x}_j is given by

$$D_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| \quad (9)$$

$$= \left[\sum_{k=1}^n (x_{ik} - x_{jk})^2 \right]^{1/2} \quad (10)$$

where x_{ik} and x_{jk} are the k th elements of the vectors \mathbf{x}_i and \mathbf{x}_j . These vectors are close if the Euclidean distance D_{ij} is small.

The similarity between vectors is determined as the reciprocal of the Euclidean distance where the smaller the distance between vectors, the greater will be the similarity. Thus, similarity is defined in the system as

$$\xi = \left(\frac{1}{D_{ij}} \right) \quad (11)$$

This metric is used to measure unknown vectors to all cluster centers. The shortest distance provided a measure of the class the vector belongs to. For some feature vectors where pattern discrimination is difficult, the variance (class boundary) of the pattern set can be used as a means to reject or accept the unknown vector as belonging to a particular class.

4.2. Self-Organizing Kohonen Network. Self-organizing networks use competitive learning paradigms in which neurons compete among themselves during training in a winner-takes-all basis. In this method, the stimulation of a particular pattern during training causes a neuron with the largest dot product to be assigned as the winning node.³⁰

Self-organizing networks adjust themselves according to the winning and neighboring neurons as input patterns are associated during the training process; however, only the winning neuron and its neighbors are actually permitted to learn.³¹ The Kohonen self-organizing map therefore allows similar patterns to be clustered to their focal center. Figure 6 illustrates a $N \times N$ Kohonen unit with the winning node and its neighborhood.

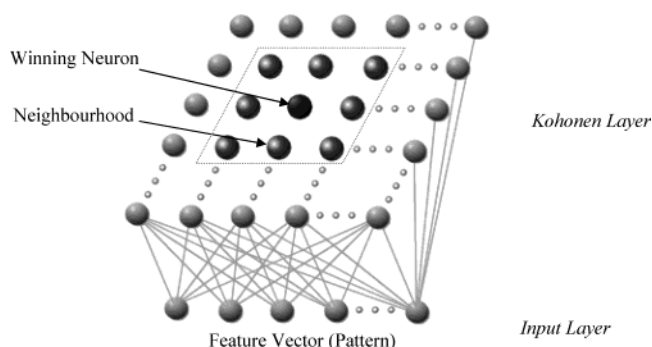


Figure 6. Clustering using Kohonen Unit (each layer is interconnected).

Feature vectors are applied to the Kohonen network, and weight values are adjusted in the neighborhood of the winning node. The change in weight values is given as

$$\Delta W_{ij} = \alpha (X_j - W_{ij}) \quad (12)$$

where W_{ij} denotes the interconnecting weights, X_j denotes the input pattern, and α is the learning rate.

The learning rate is started at a value of 1.0 and slowly decreased down to 0.1 during the training cycle. The radius of the neighboring neurons is also suitably decreased during training.

This cycle is repeated for all iterations and once completed, the winning neuron denotes the centroid of the cluster. The cluster center of each class provides measurements with which to compute the classification of an unknown vector using the Euclidean distance norm.

Figure 7 shows a schematic view of the incremental learning system. Each texture class can be independently represented to allow the addition of new classes without the need to modify existing knowledge.

Suppose an existing subnetwork or texture class required modification, only the patterns associated with that class would need to be clustered without the need to retrain all information. Should a new class of textural patterns be added to the vision system, training patterns for that class can be independently clustered. Hence, an incremental learning paradigm can provide a useful method where new information can be continually integrated into a given application.

5. EXPERIMENTS AND RESULTS

We created a data set of images arranged in three class groups a list of which is presented below.

- Ring System Group
- Nonring System Group
- Nonchemistry Group

These samples were obtained from chemical Internet resources as raster images of varying quality and resolution, since the ultimate objective of this investigation was to outline the use of this techniques for the recognition of textual information from embed raster images located on the World Wide Web (Figure 8). Ultimately, such classification of metadata descriptors could form part of the ChemDig procedure⁹ for a fully automated generation of metadata. Although, here we chose only two classifications for demonstration purposes, chemistry nonchemistry, ring nonring systems, the system could be further modified to generate metadata for other sets, such as aromatic, nonaromatic, etc.

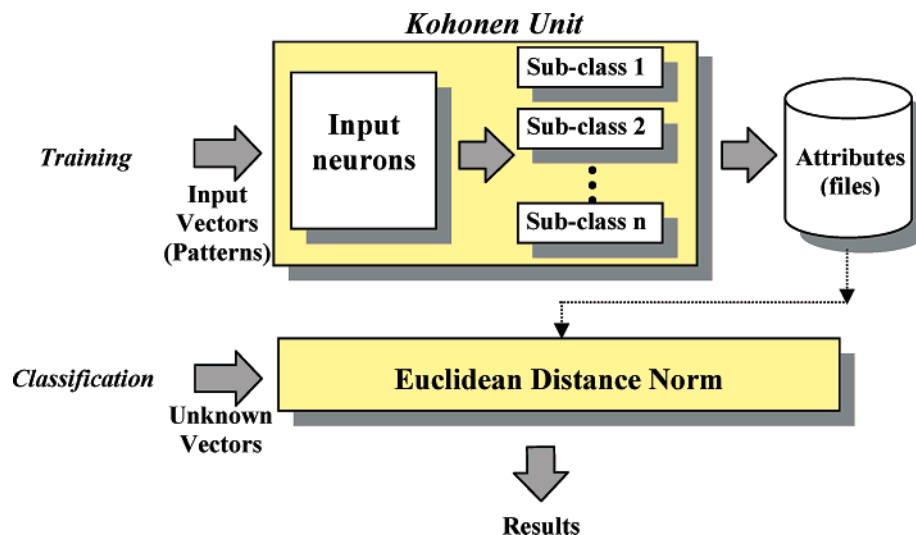


Figure 7. Topology of Kohonen unit and subnetworks for ILA.

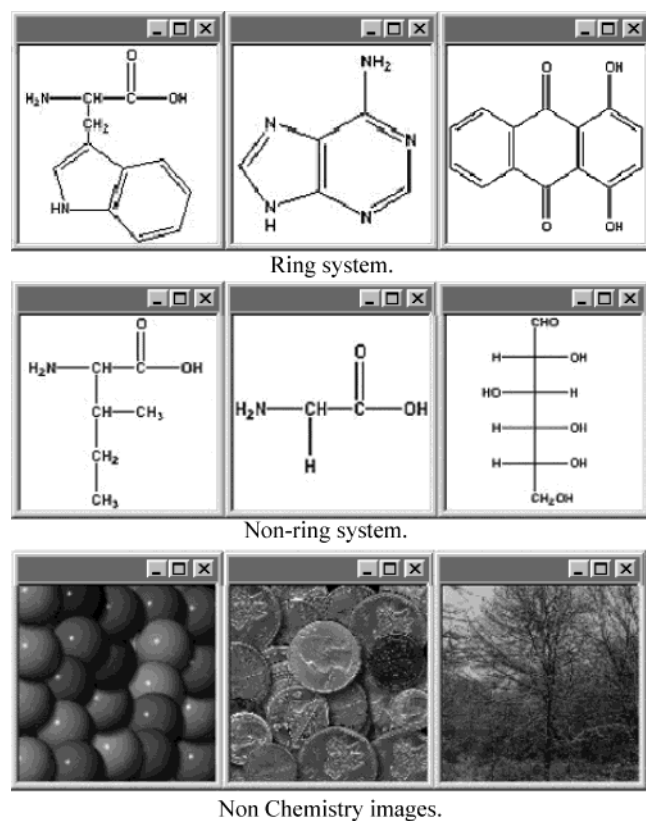


Figure 8. Examples of ring system, nonring system, and nonchemistry images.

The experimental parameters investigated were the number of features selected per image, the size of the energy mask (as used for feature energy computation), and the range of frequency channels used in the Gabor transform.

Through the analysis of the results obtained using a range of parameters, in this application, we can show that the machine vision system can be fine-tuned to better accommodate the complexity of data for a particular application. This therefore allows the system to be applied to a wide range of texture discrimination problems through careful selection of transformation parameters.

To measure the recognition rates of the system during the experiments, the classification metric is defined in

terms of the recognition rate

$$E_s = S_e/S_t * 100 \quad (13)$$

where S_e is the total number of samples incorrectly classified, S_t is the total number of test samples, and E_s is the misclassified error rate.

The recognition is given as

$$R_s = (100 - E_s) \quad (14)$$

A vector is correctly classified if it is recognized correctly with a higher percentage of recognition rate. In the experiments to find the optimum parameters for this application, a training/test set ratio of 50:50 was selected. To ensure that the system had not previously seen the pattern under detection, only the training sets were used to train the system, whereas the test sets were used for classification.

5.1. Identification of Optimum Parameters. Three experiments are designed to identify the optimum parameters for the system. The aim of the first experiment is to derive the optimum length of the features vector. The second experiment identifies the most suitable size of the energy mask. The third experiment is concerned with the identification of the most effective frequency channel used to transform structured chemical images. During the identification of optimum parameters, the data was partitioned into ring system and nonring system compounds with a training/test set ratio of 50:50. Details of these experiments are discussed in sections 5.1.1, 5.1.2, and 5.1.3.

5.1.1. Derivation of Optimum Features for Chemical Composition Diagrams. Previous experiments by Clark et al.²⁸ used feature vectors that consisted of 4000 elements per image. In our opinion, the length of each feature vector is too large and hence has adverse effects on the computational time. To improve the system performance, the first experiment is designed to find the optimum length of a feature vector for the representation of chemical composition diagrams.

In this experiment, the number of features selected per image is increased from 100 features up to 4000 features in steps of 100. For each feature size, the recognition rate is calculated and the results are plotted as in Figure 9.

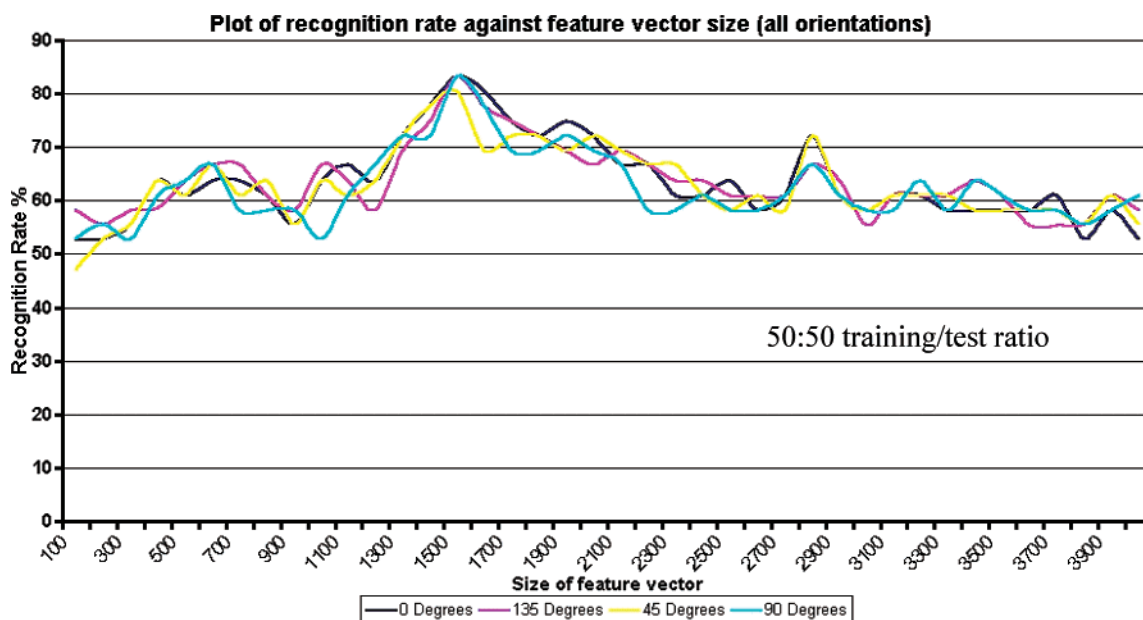


Figure 9. Plot of recognition rate against feature vector size.

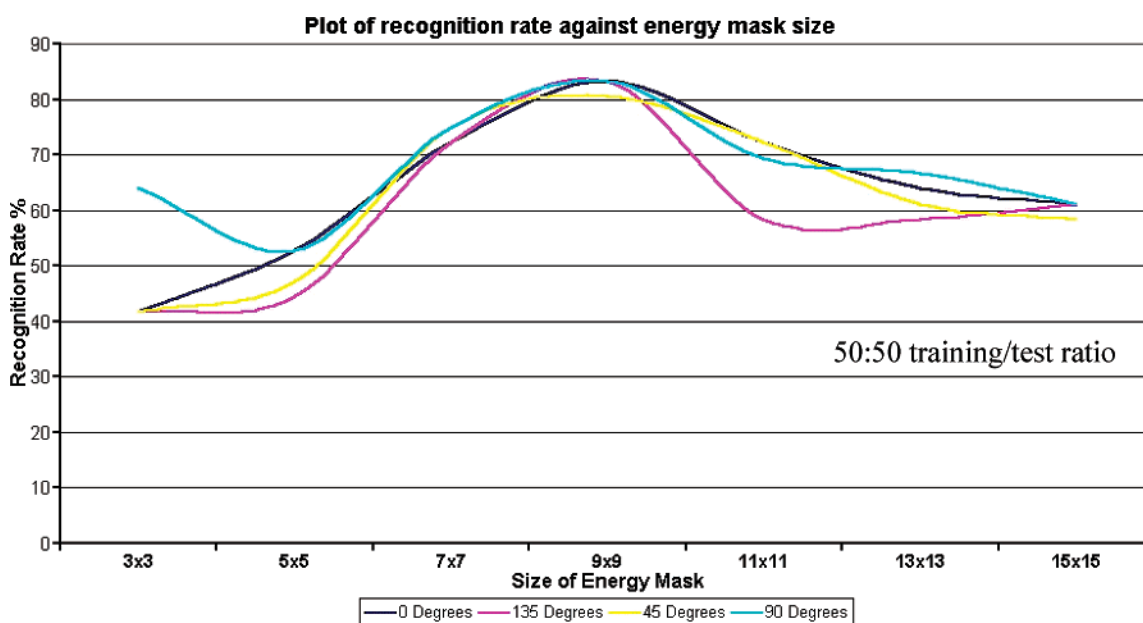


Figure 10. Plot of recognition rate against energy window size.

Figure 9 shows a graph of the size of the feature vector against the recognition rate. The results show that a high recognition rate of the order of 80% is observed when the feature vector contains about 1500 elements for various filter orientations. This therefore demonstrates that a decrease in the size of the feature vector of 62.5% is attainable.

5.1.2. Derivation of Optimum Energy Mask for Chemical Composition Diagrams. The size of the energy mask used in the computation of a feature vector element can be adjusted to suitably detect textural features. A larger mask can detect large-scale features from a texture, and a smaller mask can be used to detect small-scale features.³² In the computation of several hundreds of features per image, the computational time required is increased with the use of a larger mask size. This experiment is designed to find the optimum size of the energy mask for use with chemical composition diagrams.

In this experiment, the energy mask size is varied from 3×3 to 15×15 , and the recognition rate is recorded.

Figure 10 shows a graph of the size of the energy mask against the recognition rate.

Figure 10 demonstrates that an energy mask of 9×9 has a higher level of recognition rate when applied to chemical composition diagrams.

5.1.3. Derivation of Optimum Frequency for Chemical Composition Diagrams. All frequency channels of the Gabor transform were used in previous experiments by Clark et al.^{22,28} However, it is often observed that some frequency channels do not produce high recognition rates. To improve the system, the optimum frequency channel is investigated for the successful discrimination of chemical composition diagrams.

In this experiment, seven spatial frequency channels are applied. Previous authors, such as Jain et al.,³² have shown

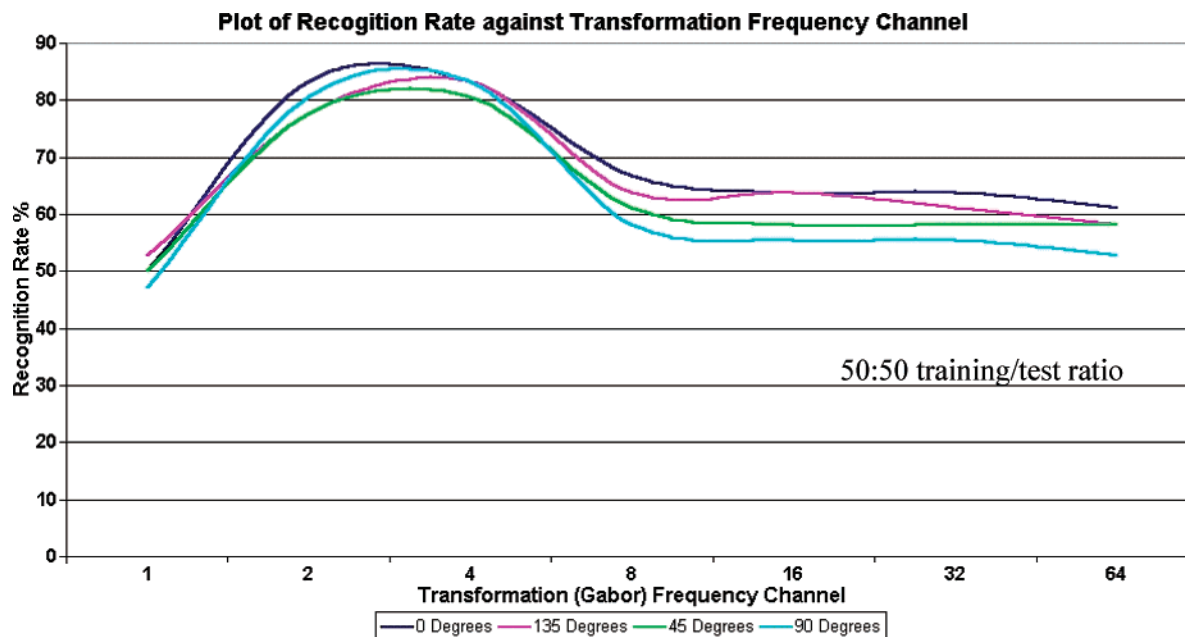


Figure 11. Plot of recognition rate against frequency channel.

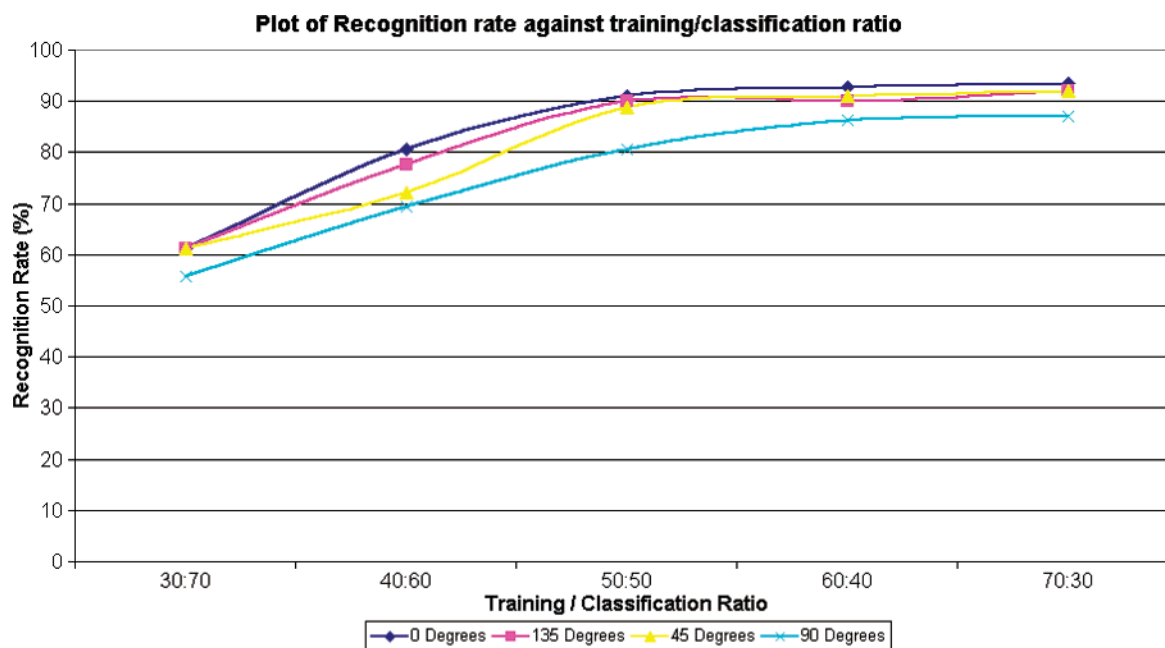


Figure 12. Plot of recognition rate against training/classification ratios.

that the frequency bandwidth of simple cells in the visual cortex is about 1 octave apart; therefore, in this experiment the frequencies are set at 1 octave apart. The data were partitioned into a training/test set ratio of 50:50, and the recognition rate was recorded for each of the seven frequency channels. The most effective frequency channel is then identified as the channel that produces a higher level of recognition.

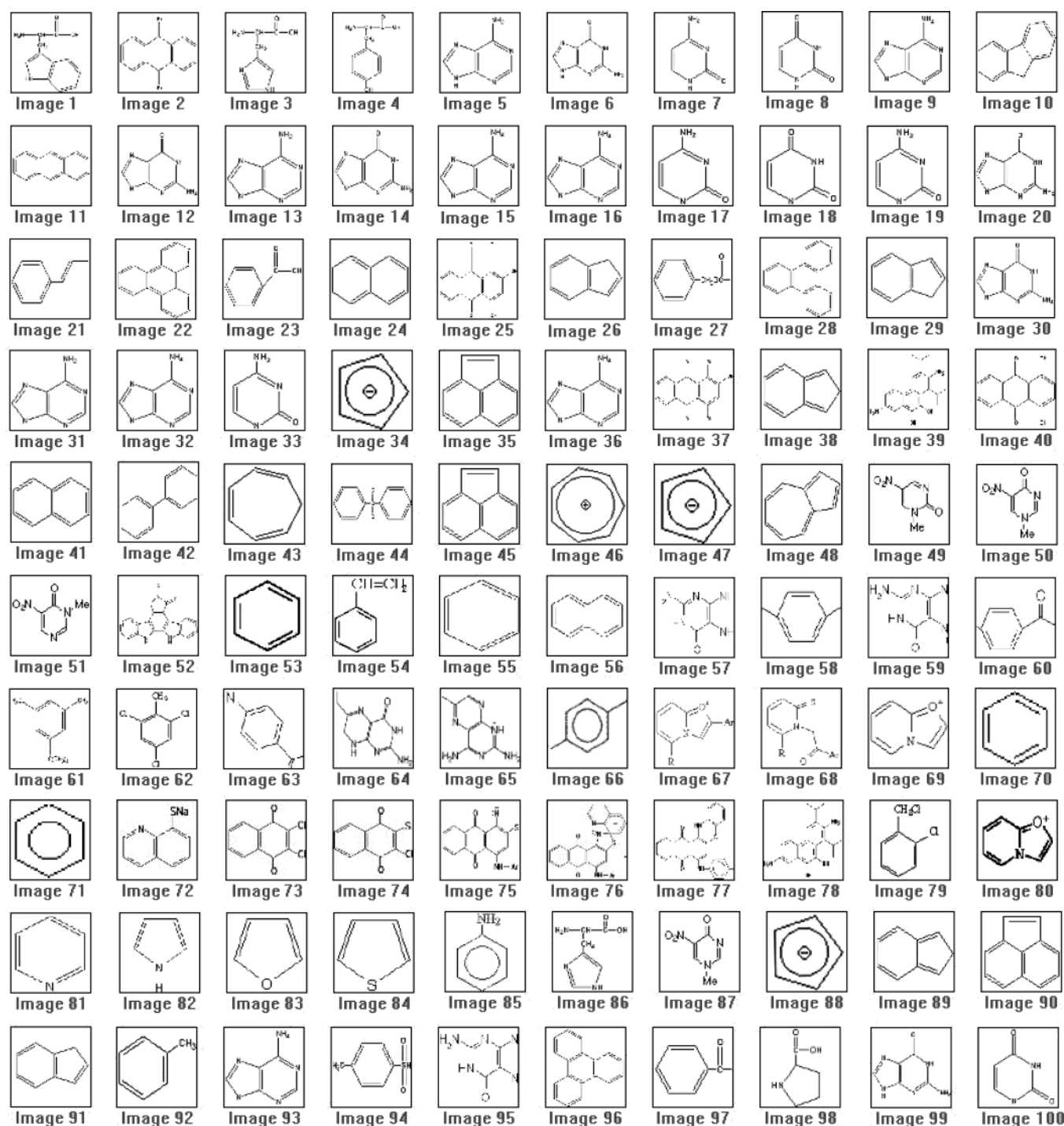
Figure 11 shows a graph of Gabor frequency channels against the recognition rate.

In Figure 11, channels $2\sqrt{2}$ and $4\sqrt{2}$ produce a higher level of recognition. However, $4\sqrt{2}$ produces and increased recognition level for all Gabor filter orientations. The remaining frequency channels show a much lower recognition rate and therefore did not discriminate as effectively between the chemical composition classes.

5.2. Detection of Ring System, Nonring System, and Nonchemistry Textures. The identification of the optimum parameters allows the machine vision system to be fine-tuned in order to produce suitable levels of recognition for chemical composition diagrams.

The objective of the experiments described here is to determine the discrimination ability of the system with respect to chemistry and nonchemistry textures. The data set consisted of three texture classes and three hundred images. The data included an additional class of nonchemistry data. This additional data set is included to verify if the system is able to appropriately distinguish between chemistry and nonchemistry textures. The nonchemistry data consisted of computer generated and noncomputer generated textures and geometric objects. The chemistry texture classes consisted of ring system and nonring system compounds (see Figure

Chart 1



8). Charts 1–3 show samples used to represent each texture class in the experiments.

To measure classification, the data sets were divided into five experimental groups and the recognition rate was recorded.

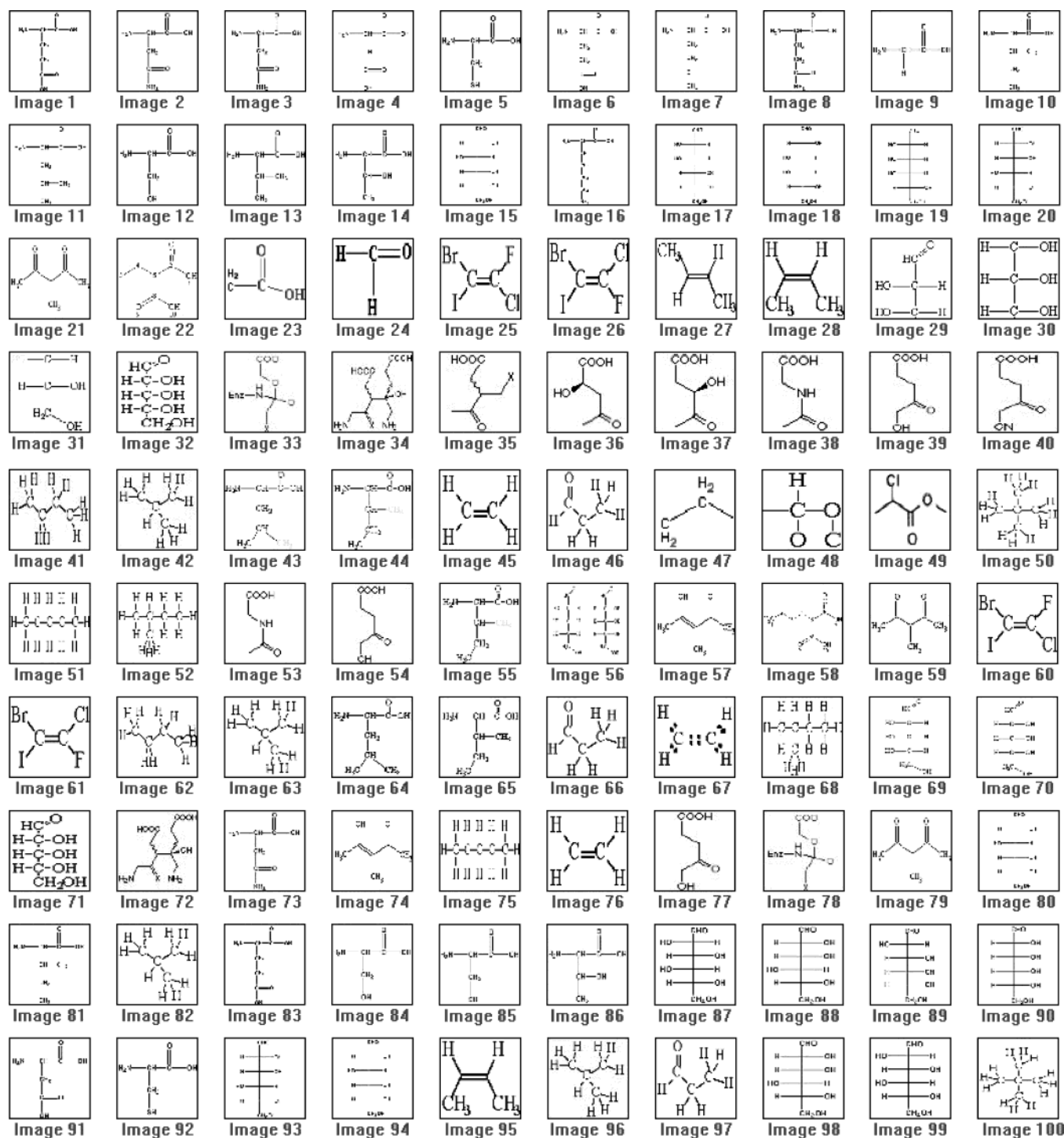
Figure 12 shows that the system is able to achieve a recognition rate of up to 91% using a 50:50 training/test set ratio. The graph also shows a smooth increase in the recognition rate as the training ratios are increased from 61% for a 30:40 ratio and up to 92% with a 70:30 ratio. These experiments demonstrate the ability of the system to distinguish between chemistry (ring system and nonring system samples) and nonchemistry data.

Table 2. List of Misclassified Images (Charts 1–3)

type of images	image number	misclassified type
ring system	59, 60, 64, 65, 68, 77, 79, 80, 90	nonring system
nonring system	50, 66, 67, 71, 72, 74, 82, 93, 96	ring system
nonchemical images	51, 57, 76, 77, 79, 82, 83, 92, 94	chemical images

In Table 2, a list of all misclassified images is presented. The first row corresponds to ring system samples that were misclassified as nonring and is the most interesting. It should be noted that images such as 59 and 95 are almost identical (according to human perception). Such images were misclassified due to the similarity of system measurements

Chart 2. Nonring System Texture Class



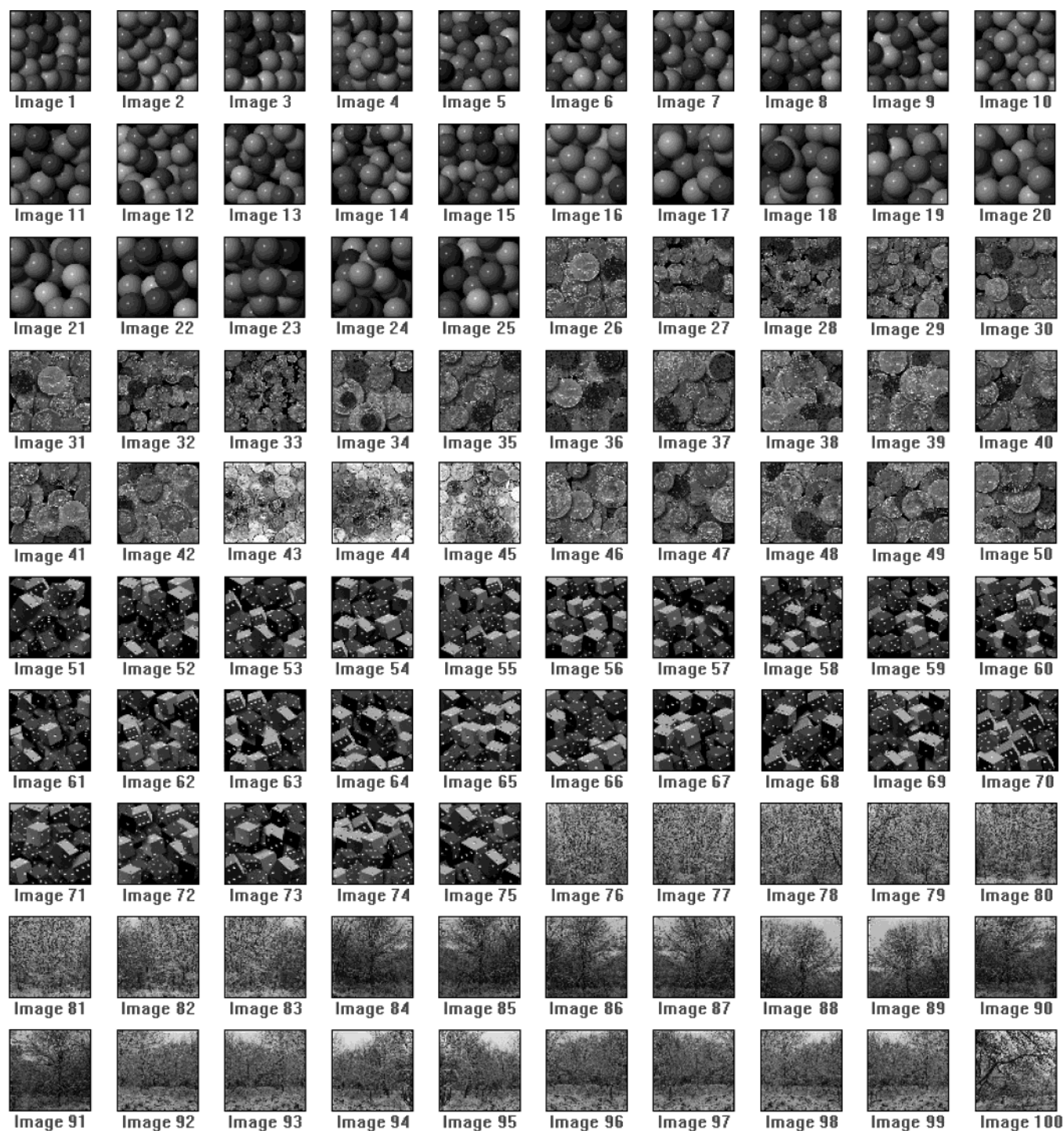
between these textures resulting in the misclassification between the border of ring and nonring system classes. These preliminary results suggest that further future work will be required to improve the chemical perception and training of the system so that it recognizes the boundaries of images that clearly belong (according to an expert human) to one category rather than another.

The second row (Table 2) represents the nonring system samples that were misclassified as ring systems. Here an explanation might arise from the fact that some of them could mimic the benzene ring outline and hence confuse the system. Again, a possible solution for such a mistake could arise by further training the system creating larger databases and force

it to classify such images as nonring systems. The third row corresponds to the list of false positives, i.e., nonchemical images classified as chemical. Although a list of false positives exists, no false negatives, i.e., chemical images classified as nonchemical, were identified by the system. The reason for this is that since the sample of nonchemical images is potentially so large, it would always be possible to select a set of nonchemical that looks absolutely nothing like the chemical structures.

Some recognition problems can arise from the similarity between chemical representations that consist of features such as lines, connected edges, and character clusters. This is often the case where similar discrimination levels from the

Chart 3. Nonchemistry Texture Class



boundary distance and class boundaries are unable to aid classification to the correct texture class. In attempting to simulate the use of machine vision for the recognition of Internet based images, a varying level of detail between chemical samples was also employed that can account for a minor number of misclassifications between texture classes.

It should be worthwhile in the future to look at the possibilities of adding weighting factors to the training set, so in effect the system will be giving a higher weighting factor to an image according to e.g. the number of rings it contains or possibly due to the importance of any given ring. Thus a system containing five rings could be considered more important than one with only a single-ring system. Although

at the moment a 91% of recognition is achieved, which is a considerably high percentage, the possibility of tuning the system to make choices in combination with databases containing classified images as judged by experts could probably boost the rate to a more satisfying level.

Application of the resulting output from the system can be used to as part of Chemdig robot-based index engine⁹ that can employ such output to form metadata declarations in associated html headers and alt image descriptors. The incremental learning paradigm employed in the system further allows for the addition of new texture classes from which to discriminate unknown data without the need to retrain any existing knowledge. In this context, should we

wish to further refine a texture class such as aromatic ring systems and nonaromatic ring systems, these texture classes could be incorporated to allow further discrimination between such groups and therefore generate additional metadata descriptors.

In general, machine vision techniques can be applied to a number of recognition problems and through optimization can derive high performance levels for the identification of texture samples, but of course it can never achieve 100% recognition levels (something probably beyond the scope of a human being as well!).

6. CONCLUSIONS

In the context of digital raster images, the most effective goal is to extract embedded chemical information, including atom and bond connection tables that allow postprocessing added value operations. However, previous work in this area has demonstrated the difficulty in obtaining such information from digital raster image data. The use of raster image information will continue to be used on the Internet to represent data such as chemical composition diagrams, and alternative techniques will need to be employed in order to discriminate the contents of such images.

Kekule and CLiDE concentrated their efforts on generating chemical metadata information to derive atom and bond connection tables from line diagrams and had to be conducted as a process with a zero error rate. Such tasks require human supervision/intervention as well as high-resolution scanned images. Both options are not feasible for extracting metadata for raster images on the Internet, since they are of low resolution and an automated low cost approach is essential.

Here, we have outlined and demonstrated an alternative methodology using machine vision concepts. The experiments have shown that a machine vision system can discriminate textural features between different data groupings. This technique allows the identification of chemistry data and further, ring system and nonring system compound information from digital raster images. The use of a texture based machine vision system avoids the need for additional semantic information (metadata) regarding to the contents of raster images and can be applied to a variety of texture identification problems.

The use of machine vision in a novel application for texture discrimination of chemical composition diagrams has shown to be successful in a number of investigative experiments. However, for use on the Internet, further work should address the use hybrid machine vision approaches where a level of semantic knowledge can be utilized in addition to texture identification requirements. Such an approach can be achieved through the embedding of XML extensions with description for image contents or analysis of associated text linked with the image on a Web page.

ACKNOWLEDGMENT

One of us (G.V.G.) thanks Merck Sharp and Dohme and the EPSRC for the award of a studentship.

REFERENCES AND NOTES

- (1) Description of GIF format is available at <http://www.w3.org/Graphics/GIF/>.

- (2) Description of JPEG format is available at <http://www.w3.org/Graphics/JPEG>.
- (3) Description of PNG format is available at <http://www.w3.org/TR/REC-png>.
- (4) W3C-SVG. Available online: <http://www.w3.org/Graphics/SVG/SVG-Implementations>
- (5) Berners-Lee, T.; Fischetti, M. *Weaving the Web: The Original Design and the Ultimate Destiny of the World-Wide Web*; Orion Business Books: London, 1999.
- (6) Gkoutos, G. V.; Kenway, P. R.; Rzepa, H. S. *New. J. Chem.* **2001**, 25, 635–638.
- (7) Gkoutos, G. V.; Kenway, P. R.; Rzepa, H. S. *J. Chem. Inf. Comput. Sci.* **2001**, 41, 253–258.
- (8) Specifications of the Molfile formats. Available online: <http://www.mdol.com/>.
- (9) Gkoutos, G. V.; Leach, C.; Rzepa H. S. *New. J. Chem.* **2001**, 5, 656–666.
- (10) McDaniel, J. R.; Balmuth, J. R. *J. Chem. Inf. Comput. Sci.* **1992**, 32, 373–378.
- (11) Selter, C. Review of Kekule 1.1, **1994**. Available online: <http://www.macworld.com/1994/11/reviews/847.html>.
- (12) Ibison, P.; Jacquot, M.; Kam, E.; Neville, A. G.; Simpson, R. W.; Tonnelier, C.; Venczel, T.; Johnson, A. P. *J. Chem. Inf. Comput. Sci.* **1993**, 33, 338–344.
- (13) Simon, A.; Johnson, A. P. *J. Chem. Inf. Comput. Sci.* **1997**, 37, 109–116.
- (14) CR2V: Raster to Vector Conversion. Available online: <http://www.celinea.com/>.
- (15) Autotrace. Available online: <http://autotrace.sourceforge.net/>.
- (16) Kolbe, T. H.; Plümer, L.; Cremers, A. B. Identifying Buildings in Aerial Images Using Constraint Relaxation and Variable Elimination. *IEEE Intelligent Systems* **2000**, 33–39.
- (17) Mokhtarian, F. Silhouette-Based Isolated Object Recognition through Curvature Scale Space. *IEEE Trans. Pattern Analysis Machine Intelligence* **1995**, 17, 5, 539–544.
- (18) Michel, J.; Nandhakumar, N.; Velten, V. Thermophysical Algebraic Invariants from Infrared Imagery for Object Recognition. *IEEE Trans. Pattern Analysis Machine Intelligence* **1997**, 19, 1, 41–50.
- (19) Sim, D.; Park, R.; Kim, R.; Lee, S.; Kim, I. Integrated Position Estimation Using Aerial Image Sequences. *IEEE Trans. Pattern Analysis Machine Intelligence* **2002**, 24, 1, 1–17.
- (20) Adjei, O.; Vella, A. Recognition of human faces based on fast computation of circular harmonic components. *The Third International conference on Multimodal Interfaces, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences; Beijing, P. R. China, 2000*; pp 160–167.
- (21) McInerney, T.; Terzopoulos, D. Deformable Models in Medical Image Analysis: A Survey. *Medical Image Analysis* **1996**, 1, 2, 91–108.
- (22) Clark, R. M.; Adjei, O.; Johal, H. Machine Vision: an incremental learning system based on features derived using fast Gabor transforms for the identification of textural objects. *Vision Geometry X, Proceedings of SPIE*; San Diego, California, U.S.A., 2001; Vol. 4476, pp 109–119.
- (23) Kulkarni, A. D. *Artificial Neural Networks for Image Understanding*; Van Nostrand Reinhold: 1994; pp 75–91, ISBN: 0-442-00921-6.
- (24) Daugman, J. G., November, High Confidence Visual Recognition of Persons by a Test of Statistical Independence. *IEEE Trans. Pattern Analysis Machine Intelligence* **1993**, 15, 11, 1148–1161.
- (25) Jain, A.; Bhattacharjee, S. Text Segmentation Using Gabor Filters for Automatic Document Processing. *Machine Vision Applications* **1992**, 5, 169–184.
- (26) Pichler, O.; Teuner, A.; Hosticka, B. J. An Unsupervised Texture Segmentation Algorithm with Feature Space Reduction and Knowledge Feedback. *IEEE Trans. Image Processing* **1998**, 7, 1, 53–61.
- (27) Ratha, N. K.; Jain, A. K.; Lakshmanan, S. Object Detection in the Presence of Clutter Using Gabor Filters. *Proc. SPIE, Applications Digital Image Processing XVII* **1994**, 2298, 612–623.
- (28) Clark, R. M.; Adjei, O. A.; Johal, H. Machine classification of textures using incremental learning based on the mean and variance of the multidimensional feature space. *International Conference on Mechantronics and Robotics* 2000, Saint Petersburg, Russia, 2000, Vol. 1, pp 27–33.
- (29) Chatfield, C. *Statistics for technology – A course in applied statistics*, 3rd ed.; Chapman & Hall: 1983; pp 100–121, ISBN: 1-412-25340-2.
- (30) Negnevitsky, M. *Artificial Intelligence – A guide to intelligent systems*, 1st ed.; Addison-Wesley: 2002; ISBN: 0-201-71159-1.
- (31) Anderson, B. Kohonen Neural Networks and Language. *Brain Language* **1999**, 70, 86–94.
- (32) Jain, A.; Farrokhnia, F. Unsupervised Texture Segmentation Using Gabor Filters. *Pattern Recognition* **1991**, 24, 12, 1167–1186.