Chapter 1

Introduction

This thesis examines the problem of automatically decomposing images into sub-images using structure inferred by an algorithm. In order to address the problem it is first necessary to define an image and a sub-image as well as to outline the reasons for wanting to find such a decomposition and possible practical applications.

1.1 Hierarchical decomposition of images into sub-images

To provide an insight into the ideas behind the proposed problem, two analogous problems are introduced. Although these may at first seem irrelevant or off topic, they serve to provide an understanding of the adopted philosophy. In other words these problems outline the viewpoint from which the fundamental approach and specific methods have been tackled.

The approach used for both of the following problems is to automatically extract structure using general methods with no prior specific information about the content of the data. Each assumes the form of the data, for example discrete ordered symbols in the case of text and an array of binary pixels for stroke detection. Furthermore it is assumed that there may be some decomposition into *objects*, or subsets of the data, based on structure which can automatically be inferred by an algorithm or set of algorithms. Perhaps most importantly, the algorithm should be able to determine its own failure when structure is not found, leaving the identification or recognition of objects to other algorithms, hence avoiding misclassifications.

1.1.1 The structure of words in English text

The first problem involves the syntactic analysis of natural language text often examined using grammatical inference [1] [2]. At the lowest level, strings, such as the one in Figure 1.1, consist of a number of discrete symbols, referred to as characters. These characters are ordered and the text is structured based on this ordering.

The cat sat on the mat.
The@cat@sat@on@the@mat.

FIGURE 1.1: An example of English text with different word delimiters.

Taking the character 's' from Figure 1.1(a) a human can quite easily examine the local neighbourhood and determine that it belongs to the *word* 'sat'. But what is a word and could a computer program make this same inference? Obviously a very simple program could separate English text into words given a predefined definition of a word in the form of a set of simple specific and *human defined* rules. So some would say the problem has been solved, but that is not the point. The idea here is to design an algorithm which given a reasonable amount of training data will be able to automatically infer the structure or underlying grammar and decompose the data for further processing or classification.

To illustrate a simple but powerful approach consider the first three characters of the sentence 'The'. Given these characters one would expect the next character to be either a space, 'r', 'y' or 's' with reasonably high certainty. However, given the four characters 'The ' (including whitespace), it is difficult to predict what the next character will be. This predictability, which can be effectively measured using information theory [3] [4], provides a simple method for separating words from English text. So, given a simple memory model, the start of a new word is identified by the inability to predict the next character.

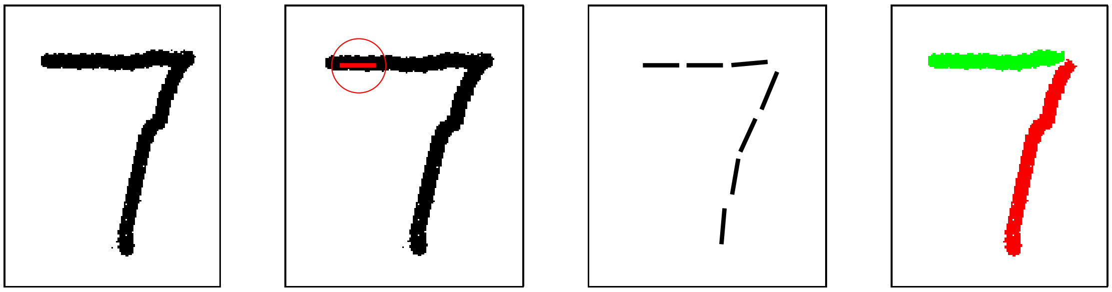
Although seemingly simple, catering for changes such as that illustrated in Figure 1.1(b), this method has some beneficial and very nice characteristics. It will automatically handle punctuation, inherently *learn* that new sentences start with a capital letter and be *surprised* when encountering a spelling mistake. The method will also work on foreign languages and even extract syntactical structure from non-natural languages such as codes and computer languages.

Once words have been identified these can in turn be combined into sentences using similar methods. Through analysis and clustering these discrete *words* can be grouped into objects corresponding to entities such as nouns, verbs and adjectives, and hence infer the grammar or syntax of the language. This is covered more completely in Section 2.8.2 (page 118).

In summary this method is automatic and general with few inbuilt assumptions. The data is modelled locally and deals with the transformation of characters to words to higher level entities in a hierarchical fashion. The problem deals with one dimensional data and will be considered a one dimensional problem.

1.1.2 Stroke detection in binary images

This problem deals with binary images consisting of black and white pixels. To simplify the problem for illustration purposes, the white pixels are assumed to be background and the black pixels the strokes or objects. Figure 1.2 shows the digit '7' which consists of two connected strokes.



(a) original (b) local region (c) line segments (d) strokes

FIGURE 1.2: Decomposition of the character 7 into strokes.

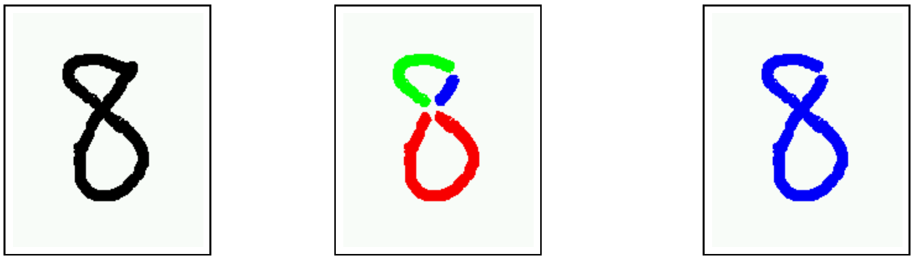
As with the analysis of natural languages, sections of the image are modelled locally as depicted in Figure 1.2(b). By choosing a small local neighbourhood centred on a black pixel (or point), a model corresponding to the shape and distribution of the neighbourhood is formed. Given the choice of an appropriate *resolution radius*, sections of the image which are part of a stroke are modelled by a small line segment, noise is discarded and intersections and solid regions are modelled appropriately.

Figure 1.2(c) shows the line segments modelled using this approach. Each model consists of measurements such as the location or centroid, the orientation and a measure of elongation. These are used to identify the strokes shown in Figure 1.2(d).

Although not immediately obvious, the process of amalgamating line segments into strokes bears many similarities to combining letters into words. Given two adjacent line segments it is possible to predict with reasonably certainty the location of a third, assuming that it is part of the same stroke. Hence within a particular stroke, the location of each line segment can be predicted in much the same way as the prediction of letters within words. Furthermore, the inability to predict line segments corresponds to the end of a stroke or the intersection between two strokes in a similar way to locating the delimiter between words.

Once strokes have been identified, similar methods are used to deal with the ends and intersections between them as illustrated in Figure 1.3. Since intersections are identified early on, these are used along with the ends of the initial strokes to produce results such as that shown in Figure 1.3(c). Similar techniques apply for handling occlusion or strokes with small missing sections.

An appropriate representation of strokes provides a powerful means for recognising or classifying hand-written characters and simple objects. Different variations of this stroke



(a) original (b) initial strokes (c) final strokes

FIGURE 1.3: Decomposition of the character 8 into strokes.

detection approach along with recognition techniques will be covered more thoroughly in Section 2.8.3 (page 120).

In summary, this general approach to stroke detection is syntactic and assumes only that the image (or pre-processed image) consists of black strokes on a white background. The image is modelled locally with a model that is hierarchical in nature, transforming pixels to line segments, to strokes and in turn into higher level entities or objects. Although it deals with two dimensional data, the method serves to identify one dimensional structure or objects and is hence also considered a one dimensional problem.

1.1.3 The question being addressed

The question posed is:

Can you decompose colour/grey scale images in a way analogous to

- *the decomposition of text into words and higher level entities*
- *the extraction of strokes from line images ;*

to find substructures and find relations between substructures.

As outlined on the previous pages such a decomposition should have a number of properties including those addressed below.

- **Generic.** The method should be general making as few assumptions as possible. Obviously for practical reasons assumptions or specific parameter values are often required. However these should not be built into the inherent framework of the approach. Should a different problem arise with similar characteristics it is desirable to make parameter changes and even minor algorithm changes rather than redesign the entire algorithm.
- **Automated.** Ideally all aspects of processing, training and recognition should be automatic without user intervention. Structure and even parameters should be inherently learnt. In practice certain knowledge may be required, such as the image

resolution, or parameter values or measurement types chosen as suitable for the application. However, the final algorithm should be completely unsupervised making decisions solely based on the data[1].

- **Hierarchical and syntactic.** The approach should combine local measurements into higher level objects in a hierarchical fashion based on structure inferred from the image. In this way the algorithm will be syntactic, extracting the syntax of the image's components automatically. The ability to identify objects in a general way adds a desirable level of abstraction to an algorithm.

- **Computationally viable.** A low level of complexity is desirable. Since local measurements and techniques are proposed, the complexity during low level processing will not be based on the number of objects in an image. However, complexity considerations should be made at higher levels. Typically this is taken care of by the hierarchical nature, since the number of objects dramatically reduces as higher level objects are located (for example in Figure 1.2 the 21760 pixels become 12 line segments and 2 strokes).

The question posed is a two dimensional problem, as opposed to the analysis of natural language and stroke detection which are one dimensional in nature. In general it must deal with regions of an image, rather than simply embedded one dimensional structure. Furthermore, characteristics such as texture, light sources, contrast and shading must be considered where appropriate. Ideally these characteristics are automatically handled by the nature of the algorithm.

The question is very general and open, dealing with general structure in images. The relevance, practicality and scope of this problem is addressed in the remainder of this chapter.

1.2 Image definitions and scope

1.2.1 Definition of an image and sub-image

An image, as defined below, consists of a set from a discrete lattice of *pixels* specified by a unique position $(\mathbf{x}, \mathbf{y})$. The *image data*, $\mathbf{d}$, consists of a set of (at least one) real numbers based on measurements made during image acquisition. These measurements may correspond to light intensity, coloured light intensity, x-ray readings or any other

[1]The data may refer to the current image being processed, or parameters or measurements obtained indirectly via training data.

quantitative measure.

$$\textbf{Image definition:} \quad \begin{aligned} \mathbf{I} : P &\longrightarrow \mathbb{R}^{\mathbf{N}} \\ (\mathbf{x}, \mathbf{y}) &\longmapsto \mathbf{d} \end{aligned}$$

$$\begin{aligned} \text{where} &\quad P \subseteq \mathbb{N} \times \mathbb{N} \\ \text{and} &\quad \mathbf{N} \text{ is the dimension of the data } \mathbf{d} \end{aligned}$$

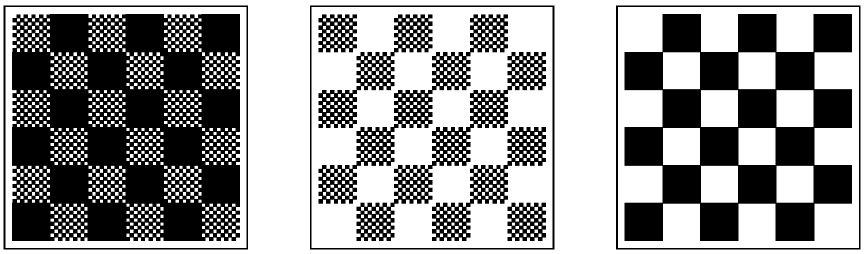
Note that this definition treats an image as a mapping into $\mathbb{R}^{\mathbf{N}}$ and makes no assumptions about the size or shape of the image. Typically digital images consist of a compact rectangular grid. This rectangular image, $\mathbf{I}_R$, is a special case of an image where

$$P = [0..\mathbf{X} - 1] \times [0..\mathbf{Y} - 1]$$

At this point it is important to note the use of a real number for each *separate* data measurement. This definition ensures that an appropriate metric can be used for analysing image data. Assigning the data to discrete values corresponding to labels, identifiers or symbols in an arbitrary order does not satisfy this definition of an image.

As a result of this general definition, a sub-image, which is a subset of pixel locations and their values, is also an image. This very general definition makes few assumptions concerning the nature of sub-images. The pixels need not have any particular spatial relationship. Figure 1.4 shows an example of an image containing two sub-images. In this example the sub-image regions are not restricted to a particular location, illustrating the need for such a general image definition.

$$\textbf{Sub-image definition:} \quad \mathbf{I_S} \subseteq \mathbf{I}$$



(a) Original Image (b) Sub-image 1 (c) Sub-image 2

FIGURE 1.4: The *checkerboard* structure. An illustration of sub-images with non-trivial spatial relationships.

1.2.2 The class of images. What data are we dealing with?

Although numerous image types are examined, from application specific grey scale images, to newsprint and real-world colour scenes, the scope of images is still limited. Tone images[2] are not considered appropriate due to their nature and simplicity. Although these techniques tend to work reasonably, it is much more appropriate to design specific algorithms with in-built knowledge. For example, the segmentation of an MRI image of the brain can obviously benefit by asserting prior knowledge about the contents of the image and its expected structure [5].

Textured images with specific structure, such as the ultrasound image in Figure 1.5 [6], are inappropriate for similar reasons. It is very difficult to segment this image into meaningful regions without knowledge of the application as well as the data acquisition method. The radial nature of this ultrasound image makes it appropriate for custom made algorithms rather than generic ones.

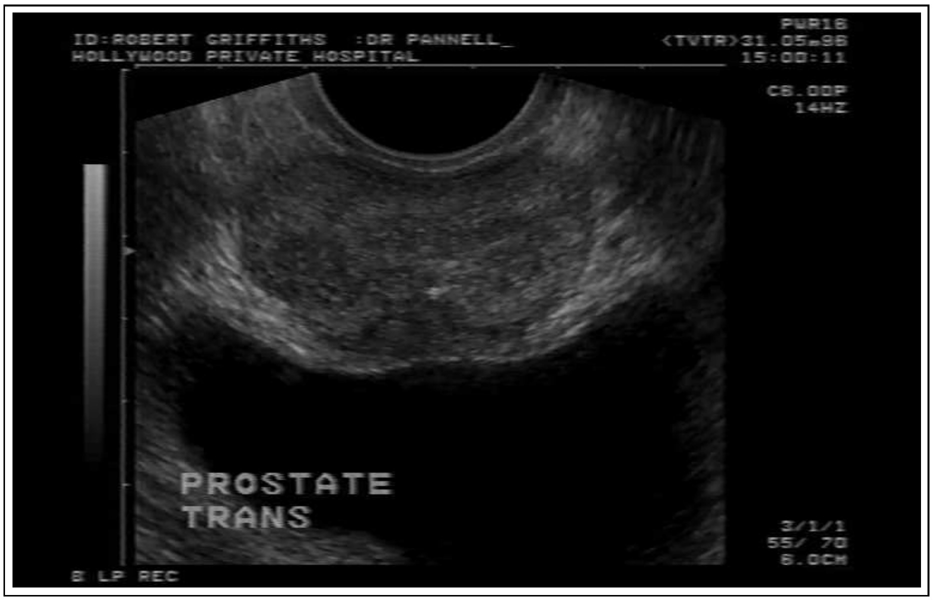


FIGURE 1.5: An ultrasound image of the prostate. This is an example of the data which is not appropriate for the generic decomposition of images into sub-images.

Other than such highly studied restricted problems, most images depicting real-world data are considered within scope. In general any image containing an arbitrary number of real-world measurements and some spatial relationship is appropriate for processing. No further restrictions are imposed.

Perhaps the most important point is that the data is real rather than artificial or theoretical in nature. Hence, concentration is not placed on locating theoretical edges or well defined lines, but rather the decomposition of the image into objects which are useful for classification or recognition in practical applications.

[2]Those which can be classified purely by grey scale values. Refer to Section 2.5 for more details.

1.2.3 Examples of legitimate sub-images

The first two examples were presented and discussed in Section 1.1.

Text strings as images

Although a string of text is not strictly an image, it can be represented as one according to Definition 1.2.1 with $\mathbf{X}$ equal to the string's length, $\mathbf{Y} = 1$ and the data, $\mathbf{d}$, being $\mathbf{N}$ dimensional with a binary value used for each of the $\mathbf{N}$ discrete symbols[3]. Using this definition sub-images consisting of space separated *words* can be identified by locating connected components of binary value 0 in the appropriate dimension of the data. Although the automation of this process is non-trivial the structure is perhaps more obvious than one might first think by noticing the frequency and regularity of the *space symbol* as shown in Figure 1.6.

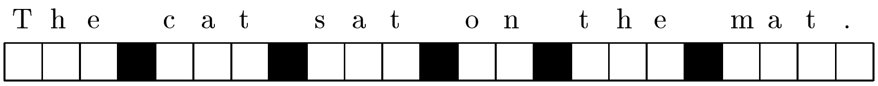
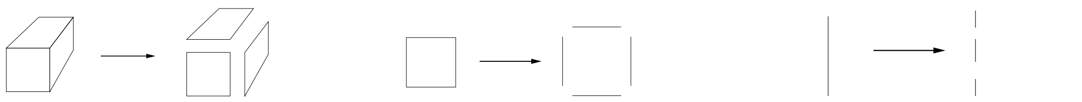


FIGURE 1.6: The *space symbol* dimension of an image representing the string 'The cat sat on the mat.' The words, separated by connected white *pixels*, are a simple example of sub-images.

Although not really applicable for natural language, this does illustrate a simple example of a sub-image. Furthermore such a representation may be useful for analysing structure in codes with only a handful of symbols.

The structure of cubes

As depicted in Figure 1.7 a binary image of a cube consists of three sub-images corresponding to quadrilaterals. In turn, each quadrilateral contains four strokes which are made up of line segments or pixels as discussed in Section 1.1.2. This decomposition, which is theoretical in nature, is studied by McLaughlin [7] [8].



(a) Cubes to quadrilaterals (b) quadrilaterals to strokes (c) strokes to line segments

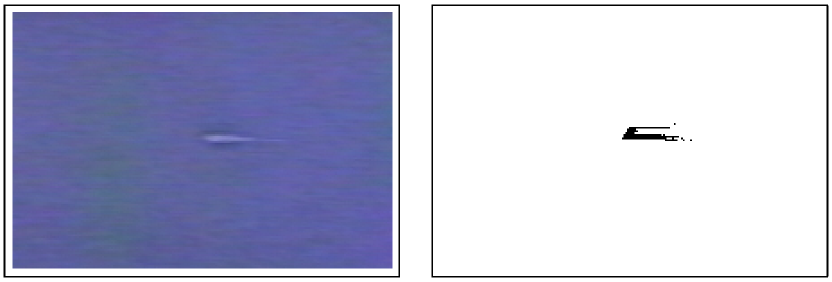
FIGURE 1.7: The hierarchical structure of cubes consisting of quadrilateral, strokes, line segments and, implicitly, pixels.

[3]Note that it cannot be defined as a one dimensional image since a discrete symbol cannot be assigned to a real number meaningfully.

The detection of anomalies

This is perhaps one of the most general and useful of the simple examples of sub-images. For images containing regular structured backgrounds and foreground objects which stray from this regularity, two sub-images result. These correspond to the background region and non-background, or *other region*. In the general case the *other region*, which may cover multiple locations of the original image, contains information pertaining to meaningful content. The characteristics of this region are typically not well predicted by the rest of the image and are hence considered anomalous.

Figure 1.8(a) shows a section of manufactured fabric[4] containing an anomaly. In this example, the background characteristics are highly homogeneous and the anomaly clearly stands out to the human observer. Hence, the decomposition into two sub-images is appropriate. Furthermore the anomaly contributes to a large proportion of the image's information content even though it is relatively small.



(a) original image segment (b) location of sub-images

FIGURE 1.8: The decomposition into sub-images corresponding to image background and an anomaly or object.[4]

Newspaper print

Newspapers contain a collection of pictures and text on a white or black background. The entities *picture*, *text* and *background* are examples of legitimate sub-images. Figure 1.9 shows a grey scale newspaper image and the corresponding picture sub-image.

Natural scenes

This final example is most diverse and general. In principle, natural scenes can be decomposed into objects, although any particular decomposition is debatable and open to interpretation. Hence, colour images of real-world scenes can be separated into sub-images corresponding to meaningful objects, such as that illustrated in Figure 1.10.

[4]All fabric images are from Natural Textiles Australia and have been captured from video shot by an amateur with a hand-held recorder.

(a) original image segment (b) sub-image corresponding to pictures

FIGURE 1.9: A newspaper image containing picture, text and background sub-images.

(a) original (b) three possible sub-images

FIGURE 1.10: A natural scene and its decomposition into sub-images.

1.3 Reasons for thinking images often do decompose hierarchically

Consider an image of an old lady standing in front of a shop. The shop is painted in a cream glossy paint and has two windows, a solid wooden door, and a red tiled roof. On the door is a large brass handle below a notice board containing three notices, one of which is a poster containing a picture of a man above some obscure text. In the background the sky is blue with two dominant clouds. A well maintained lawn surrounds the shop, extending back to a wooden fence.

The above description allows a human to quite easily reconstruct a scene in their own mind. This *reconstructed image* makes use of high level abstract objects and incorporates personal knowledge and experience. Obviously the image will not be unique, differing from one person to another based on assumptions or choices made. The image may however be considered correct if it can justifiably match the natural language description, or the image model. This is obviously open to interpretation since the description lacks precision. No mention is made about the size of the shop, the angle of the roof or the meaning of phrases such as *dominant* and *well maintained*. However, even with a precise description the reconstruction is still open to different interpretations of the information. Such representation and reconstruction issues are discussed in Chapter 3.

Now consider the inverse problem where a person is asked to describe the original image. The description, “It’s a lady outside a shop” would be an appropriate one, although it inherently ignores the background objects. When asked for more information a descrip-

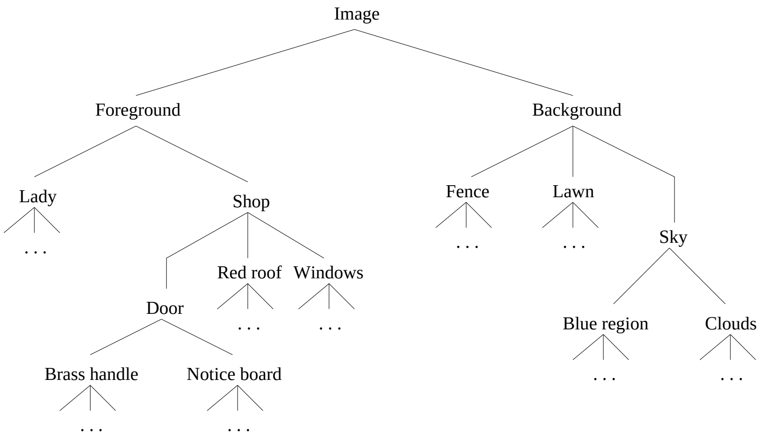


FIGURE 1.11: The structure of a hierarchical decomposition of a scene into meaningful regions.

tion of the background objects and a more detailed description of the prominent objects could be expected.

The very nature of the above communication suggests an underlying hierarchy in both the meaningful content and the structure of images. A simple analysis or description includes the most prominent or highest level objects. Subsequent descriptions account for the details and lower level structure of individual objects. A possible decomposition of this example is shown in Figure 1.11.

Similar observations can be made about all of the image types presented in Section 1.2.3. For example, a newspaper image such as Figure 1.9 consists of pictures and sections of text on a white background. The sections of text can be decomposed into paragraphs, paragraphs into words and in turn words into characters, where the words and characters are defined according to spatial separation.

In summary, the very way people describe, think of, and possibly store image information suggests an inherent hierarchical structure. This can also be illustrated through countless examples of various image types.

1.4 The objectives

The main objectives of this research are summarised below.

- Generic feature extraction
- Automatic extraction of structure
- Classification and recognition
- Image reconstruction

1.4.1 Generic feature extraction

The nature or framework of the feature extraction process should make as few assumptions about the properties of the image as possible. The aim is to make the approach adaptable to both resolution or scale changes, and changes in raw measurements.

For example, if a set of images is scaled in size, the feature extraction process will either adapt automatically or require a simple parameter change. Although this might sound trivial, many feature extraction methods use predefined masks or specific features at the single pixel level and cannot be adapted in this way. A common example is the 3×3 pixel mask used by the standard Sobel edge detector.

Changes in the type of images may lead to the need for measurement changes, with possible consideration for intensity, texture, colour, local spatial frequency or any other characteristic. The objective is to use a feature extraction framework which is easily adaptable to changes in the measurement made and also supports multiple measurements.

1.4.2 Automatic extraction of structure

Given a generic set of feature vectors, for arguments sake $\mathbf{P} \subset \mathbb{R}^{\mathbf{N}}$, structure is to be automatically extracted. The identification of this structure allows sub-images to be found.

In order to keep this process automatic it is desirable to develop a number of methods to extract structure from $\mathbf{P}$. It is also necessary to develop measurements to ascertain the success of each method. Such a set of tools and measurements provide a framework for hierarchically extracting structure from the image.

An example is the use of the expectation maximisation (EM) algorithm applied to Gaussian mixture modelling [9] [10] [11] to find clusters in datasets. One possible measure of success is obtained by using the *log-likelihood* measurement and the distribution of Mahalanobis distances from each data point to its respective cluster model. This measure allows the algorithm to automatically decide whether obvious structure, in the form of clusters, exists. Perhaps most importantly this method relies only on a set of data points in Euclidean space ($\mathbb{R}^{\mathbf{N}}$) to determine whether the set should be separated in prominent clusters.

1.4.3 Classification and recognition

The objective is to use the structure and knowledge automatically inferred from the image to identify, classify or recognise objects in images. The identification of objects may be done based on the content of a single image being analysed. However, classification and recognition are typically performed based on training with a pre-labelled set of images or sub-images. In general these rely on an appropriate representation for use in a classification method, such as K-nearest neighbours or a neural network.

1.4.4 Image reconstruction

It is desirable to reconstruct an image or components of it. Obviously this is useful for image compression applications where compression is obtained by encoding structure in an intelligent way. However, it is also very useful in understanding and analysing the effectiveness of the underlying algorithms and automatic methods. As outlined by the description of a scene in Section 1.3 the reconstruction of an image from a higher level representation will differ from the original image due to both information lost during the representation phase and the inadequacies of the reconstruction phase.

For example, a representation describing a roof as red with tiles on it and precisely defining sizes, shapes and tiling distribution allows a very good reconstruction. However, if one of the tiles is cracked and this information is left out of the representation, or disregarded as noise, the reconstruction will omit this detail. In this case an insight is provided into the decomposition algorithm and the effectiveness of object representations. This can be extremely useful since this effectiveness governs the success of higher level processing or object classification.

1.5 General uses and applications

The generic analysis and decomposition of images provides a powerful framework for image segmentation and object recognition. This framework is useful for numerous practical applications. As previously outlined, although methods are useful for image segmentation and region classification of medical images, these applications benefit substantially from specific techniques using well defined prior knowledge and hence are not considered here.

The following applications are discussed.

- Anomaly detection and classification
- Segmentation of newspaper or mixed-mode images
- Content based image retrieval
- Scene analysis and robot applications
- Image compression

1.5.1 Anomaly detection and classification

In production line manufacturing the screening of materials is often required. Traditionally, cheap labour is used to perform this task. There are several disadvantages to these systems.

- The work is low paid, conditions may be difficult and the work is tedious and repetitive.

- Results are subjective and may vary not only between individuals, but also be based on the recent level of quality of the materials being produced. On a very good day, a product with a slight fault could to be deemed defective, whereas after experiencing severe defects, an unacceptable product may be passed.

- The cost for companies is higher than a fully automated system. This often makes the process economically unviable in countries without cheap labour, perhaps also adding additional transportation overheads.

Automatic screening is an attractive alternative, since it is cheaper and produces more consistent results. Obviously the location of anomalies is not sufficient without a classification system to determine whether the flaw is a serious problem or an acceptable feature. For example, a crease in the fabric is possibly acceptable, but a hole in the fabric is not.

In summary, a real-time automated system is required which not only locates anomalies, but also classifies them based on prior training.

1.5.2 Segmentation of newspaper or mixed-mode images

First, the segmentation of images containing pictures, text and other types of regions allows an algorithm to 'read' a newspaper. Text regions are decomposed further and interpreted using optical character recognition (OCR) systems. Pictures are dealt with appropriately.

This type of segmentation is also useful in assisting algorithms which attempt to correct the orientation of an image by rotating it. These algorithms generally assume that the image consists of text, using specific methods to determine the image's relative orientation. The removal of pictures from the original image serves to improve such algorithms.

A further application is one of effective mixed-mode compression. Current image standards such as *jpeg* and *gif* [12] have advantages and disadvantages with a tradeoff between image size and quality.

The *jpeg* image format makes use of lossy compression via the discrete cosine transform (DCT) to provide an image which looks acceptable to a human, even though it is missing the high frequency information or fine detail. Hence a picture of a person or natural scene benefits from an impressive compression rate while still being acceptable to the human observer. On the other hand, regions of text will lose clarity and often be unreadable.

In contrast, the *gif* image format is lossless in terms of spatial information, but compromises the raw compression rate and number of colours. Hence, text regions will appear with clarity, but pictures will be unnecessarily large and seem rough and computerised due to the lack of colour resolution.

By locating regions of picture and text, the appropriate compression regime can be applied to different sections of the image. This approach not only allows compression of static images, but is also useful for compressing video images for displaying remotely.

1.5.3 Content based image retrieval (CBIR)

With the rapid growth of the *world wide web* and massive on-line image databases there is an ever increasing need for the automatic storage and fast automatic retrieval of image data based on content. Traditional systems making use of manually entered text annotations are slow, labour intensive and subjective. Alternatively, many automated systems make use of simple measurements such as colour percentages which do not capture the true content of the image. While such systems are useful for index images they offer no method for intelligently searching or querying images. Even queries for images similar to another sample image produce unsatisfactory results.

The decomposition of images into regions corresponding to objects provides a powerful means for scene analysis and use in CBIR. Through the identification of meaningful objects, images can be automatically labelled. In the simple case the characteristics of objects, regions of the image, or in general, sub-images are stored. These might include information about location, region size, colour, intensity, texture and location relative to other objects in the image. Such information is appropriate for database searches based on a query image.

More advanced systems can then make use of this simple information along with pre-labelled images, or image objects, to train a system to automatically recognise objects. This would ideally lead to a system capable of returning images containing horses or aeroplanes with red wings.

1.5.4 Scene analysis and robot applications

Autonomous robots rely on a reasonable description of the objects surrounding them in order to navigate effectively. For example, a robot which delivers mail around a building with narrow corridors, blue carpet, white walls and dark grey labelled doors [13], requires a method of decomposing the scene in order to develop or follow a predefined map. The decomposition of the scene into sub-images corresponding to the carpet, doors, walls and other obtrusive objects, such as people and other robots, allows the necessary information for the robot to function autonomously.

1.5.5 Image compression

Most standard compression techniques make use of either information theory or projections onto a given set of basis functions. Examples are *gzip*, which uses *lz* compression [12, p196–208] and *jpeg* [12] which projects an image onto a set of orthonormal cosine functions more commonly known as the cosine transform. As outlined in Section 1.5.2 the intelligent segmentation of an image based on content allows for much more effective and acceptable lossy compression.

In the general case, sending a description of the content of an image may be far more efficient than sending the standard compressed image. Obviously this is highly dependent on the ability to describe the image concisely based on its structure and the ability to reconstruct[5] the image without unacceptable loss of information. However, an image compression scheme should be able to benefit from such intelligent decompositions. Of course the compression rate may vary considerably from one image to the next, thus probably just providing an alternative compression method rather than a substitute.

1.6 Thesis outline

The remainder of this thesis tackles the example image types, general objectives and applications introduced in this chapter.

Chapter 2 provides a general background to traditional methods of image analysis and more sophisticated approaches to image segmentation and classification. Spatial frequency, texture, colour representation, colour image segmentation and intelligent image retrieval systems are discussed. These background topics are concluded by a discussion of syntactic methods.

A more precise description of the approach is given in Chapter 3 through the introduction of a process known as *The UpWrite* and a corresponding *DownWrite* process. These processes provide a framework for the automated image decomposition problem.

Chapters 4 to 6 illustrate three examples of applying the framework of the *UpWrite* to restricted or simplified problems. The first examines grey scale images with homogeneous background characteristics and a number (usually at least one) of foreground objects which differ from the background in some way. This is mainly illustrated through the identification, higher level analysis and classification of defects in painted steel. Image reconstruction is also presented.

The second restricted problem (Chapter 5) deals with segmenting grey scale newspaper images into regions corresponding to text, pictures, background and region edges. The approach, or framework, used for this problem is very similar to the detection of defects in painted steel images as highlighted by the similar naming of each chapter's sections.

Chapter 6 follows the same framework, again highlighted by the section headings. This deals with the problem of segmenting colour images of natural real-world scenes into homogeneous regions corresponding to objects. Through the initial identification of spatially disjoint homogeneous regions, higher level techniques are used to combine regions into objects. This amalgamation process makes use of region characteristics such as intensity, texture and colour, in combination with region location and modelled variations in characteristics such as simple lighting variations. Once again image reconstruction is illustrated showing the effectiveness of the region models.

Conclusions, an indication of original content and directions of future work conclude this thesis.

Throughout this thesis results and illustrations have been produced and displayed using

[5]Refer to Section 1.4.4 on page 13 for more detailed discussion.

custom programs written in standard ANSI C. In Figures where this is not the case, the tools used or source is clearly stated. Unless otherwise stated all results and illustrations were processed using a Silicon Graphics Onyx workstation and run on one of its four MIPS R-10000 processors.