Genetic Programming for Document Images Segmentation and Classification

by
Hayden Mulholland

Submitted in fulfilment of the
requirements for the degree of
Master of Science

at

School of Computer Science,
University Of KwaZulu-Natal
South Africa
December, 2007

SCHOOL OF COMPUTER SCIENCE,
UNIVERSITY OF KWAZULU-NATAL

The research described in this dissertation was performed at the University of KwaZulu-Natal under the supervision of Prof. Jules-Raymond Tapamo and Dr. Nelishia Pillay. I hereby declare that all material incorporated in this dissertation is my own original work except where acknowledgement is made by name, or in the form of a reference. The work contained herein has not been submitted in part or whole for a degree at any other university.

Signed: ______________________
Hayden Mulholland

Signed: ______________________
Prof. Jules-Raymond Tapamo

Signed: ______________________
Dr. Nelishia Pillay

Dated: December, 2007

My supervisors,

my colleagues,

my family,

and my friends.

Table of Contents

Abstract . . . . **viii**

Acknowledgements . . . . **ix**

Chapter 1 Introduction **1**

1.1 Context and problem description 1

1.2 Objectives of study 2

1.3 Contributions of the study 2

1.4 Plan of the dissertation 3

Chapter 2 Literature survey **4**

2.1 Image segmentation 5

2.2 Feature extraction 5

2.3 Classification techniques 7

2.4 Document image segmentation systems 9

2.5 Evolutionary methods 9

Chapter 3 Image segmentation methods and techniques **13**

3.1 Texture Segmentation 13

3.2 Feature selection 14

3.3 Feature extraction 15

3.3.1 Markov Random Fields 15

3.3.2 Discrete Wavelet Transforms (DWT) 16

3.3.3 Gabor Filtering 17

3.3.4 Grey Level Co-Occurrence Matrices and Haralick Features 19

3.4 Conventional Classification 23

3.4.1 K-Means 24

3.4.2 C-Means 29

Chapter 4 An introduction to genetic programming 34

4.1 Genetic populations 35

4.2 Genetic system control models 35

4.3 The representation of individuals 35

4.4 The primitives 36

4.5 The terminal set 36

4.6 The function set 36

4.7 Fitness cases 37

4.8 The evaluation of individuals 37

4.9 The fitness function 37

4.10 Selection and inverse selection methods 39

4.11 Genetic operations 40

4.11.1 Reproduction 40

4.11.2 Mutation 41

4.11.3 Crossover 41

4.12 Termination criteria 42

4.13 Population initialisation 43

4.14 Parallel populations 44

Chapter 5 Systems implemented 47

5.1 Genetic system 47

5.1.1 The concept 48

5.1.2 Quick clustering 50

5.1.3 The training module 50

5.1.4 The segmentation module 50

5.1.5 The genetic parameters 50

5.1.6 The training set 54

5.1.7 Human input 54

5.1.8 The terminal set 56

5.1.9 The splitting of training and input data 57

5.1.10 The function set 58

5.1.11 The representation of individuals 61

5.1.12 The fitness function 61

5.1.13 Finishing off 64

5.1.14 Example training and segmentation 64

5.2 Comparison system 68

5.3 Post-processing 69

5.3.1 Pass 1: fuzzy class removal 72

5.3.2 Pass 2: text component clean-up 73

5.3.3 Pass 3: noise culler 74

5.3.4 Pass 4: block assignment 76

Chapter 6 Experimental results and discussion 78

6.1 Data set 78

6.2 Genetic program creation 79

6.3 Methods used for comparison of results 79

6.3.1 Basic terminology 80

6.3.2 Confusion matrices 80

6.4 Measures derived from a confusion matrix 81

6.4.1 Receiver Operator Characteristic (ROC) graphs 82

6.5 Individual results 83

6.5.1 Result comparison 83

6.5.2 Run-time comparison 83

6.5.3 Qualitative comparison 88

6.5.4 Statistical significance 89

6.6 Interpretation of trained genetic programs: 91

6.6.1 Image 92

6.6.2 Text 92

6.7 Random individual generation vs. genetic programming based individual generation 92

Chapter 7 Conclusion and future work **94**

7.1 Limitations 95

7.2 Future work 95

Bibliography **97**

Appendix A Implementation details **105**

A.1 Program Design 105

A.2 The genetic system 106

A.3 The image processing system 109

A.4 The human interface classes 112

A.5 The filing 113

A.5.1 .PWN Cached image format 113

A.5.2 .GDT Genetic input format 114

A.5.3 .HDT Genetic algorithm format 115

Appendix B Comparison of results **117**

Appendix C Original images **123**

Appendix D K-Means based method results **131**

Appendix E Genetic programming based method results **139**

Abstract

Document image analysis systems automate the process of interpreting document images by combining information extracted from the document image. An important part of this information is obtained via image segmentation. This dissertation focuses on texture based document image segmentation and discusses some common methods of feature extraction and classification that are used to achieve the segmentations.

Genetic programming is a relatively fresh approach to the problem of document image segmentation and is considered to be a good technique for optimising existing algorithms. In this dissertation, a document images segmentation technique based on genetic programming is presented. The genetic programming based solution performs segmentations and combines them in a suitable manner to solve the document image segmentation problem.

The segmentations that the genetic programming based method combines to perform segmentation can be from the K-Means algorithm or even a combination of any segmentation algorithms that express their segmentations in a suitable format. For the purpose of this dissertation, the K-Means algorithm will be focused on.

From experiments performed using the Grey Level Co-occurrence Matrix(GLCM)/K-Means based method and the genetic programming method it appears that the genetic programming based method is capable of producing better and more consistent results than the GLCM/K-Means based method. The genetic programs also make more efficient use of the Haralick features due to the feature selection approach taken when creating the genetic programs. This results in far fewer Haralick feature spaces needing to be calculated in order to perform segmentations of comparable accuracy.

Acknowledgements

I would like to thank the following people and institutions for their assistance with the creation of this dissertation:

- My supervisors; Prof. Jules R Tapamo and Dr. Nelishia Pillay for all of their ideas, advice, help and support throughout the course of this project.
- My colleagues and Soren Greenwood for their help and ideas.
- My parents, sister and friends for all of their support.
- Claire McHendrie for all of her help and encouragement.
- Adriaan Tijsseling for the Gabor filtering libraries that have been adapted for the Gabor filtering used in this project.
- The Mediateam image database staff.
- All of the academics kind enough to make their publications easily available online.
- The NRF and the University of KwaZulu-Natal for their financial support.

Chapter 1

Introduction

1.1 Context and problem description

Document image analysis is a major field of research in image processing, with a large number of applications. Document image analysis systems automate the process of the interpretation of document images by combining information about the document image. Such information is obtained through the use of image segmentation, layout understanding, symbol recognition, and the use of rules based upon the application. The combined data is then used to interpret a document image in a way that is most suitable to the application[63, 96].

The problem that this dissertation focuses on is texture based document image segmentation. Texture based segmentation is the partitioning of an image into regions based upon the attributes of the textures in the image. Document image segmentation is an important part of document image analysis as it provides information from which the layout of a document is determined. The layout of a document image is in turn used to determine which regions to run optical character recognition (OCR) on, which regions to store or process as images, and which parts of the image simply contain background.

Texture based document image segmentation algorithms are generally based on feature extraction and classification techniques. Feature extraction techniques include Gabor filters, GLCM's (henceforth referred to as grey level co-occurrence matrices) and wavelet transforms. Classification algorithms include techniques such as neural networks and clustering techniques such as the K-Means and C-Means algorithms. In general, information about the document image is obtained via feature extraction algorithms, and then processed by a classification algorithm to achieve a segmentation.

Evolutionary algorithms are artificial intelligence techniques which entail the simulation of a population undergoing biological evolution. The population is comprised of possible solutions to a problem. This is achieved via the use of mechanisms inspired by evolution such as reproduction, mutation, genetic crossover, and natural selection[11, 32].

The use of evolutionary methods is relatively new in image processing and they are playing an ever increasing role in document image analysis. Evolutionary methods are frequently used to improve existing solutions to problems, as they are often able to create more computationally efficient and accurate algorithms.

There are many texture based document image segmentation implementations that combine a feature method and classification approach, and produce good results[10, 24, 25, 41, 56]. However, many of these systems are very computationally expensive, and most document image segmentation algorithms are not entirely accurate. Better solutions can also be reached by determining which features are the best to cluster for a particular group of images, or by finding new ways to combine different feature methods.

Evolutionary methods could lead to a more efficient solution to the problem of texture based document image segmentation, and such techniques could be adapted to other segmentation and classification applications. In this dissertation, a document image segmentation technique based on an evolutionary approach to the K-Means classification of Haralick features from GLCM's is presented.

1.2 Objectives of study

The objectives of this study are to:

(1) Survey and discuss the current state of texture-based document image segmentation.

(2) Survey and discuss current genetic programming techniques and systems.

(3) Develop a novel texture based method for document image segmentation via genetic programming.

(4) Test genetic programs (trained by the developed genetic system) against a large data set of images, to determine the accuracy of the trained genetic programs.

(5) Compare the results obtained against results obtained via a conventional technique.

1.3 Contributions of the study

In this study, a novel document image segmentation process based on genetic programming has been developed. To train a genetic program, the implemented system uses a single training set, which consists of: the training image, segmentations of the image via a K-Means/Haralick feature based algorithm, and some user input. The trained genetic program is then able to accurately segment images with reasonably similar distributions of Haralick features (see chapter 6 for detailed results).

The system currently makes use of only the K-Means/Haralick feature based algorithm for training the genetic program. However, the framework is designed in such a way that it is possible to make use of, and combine, multiple segmentation techniques with the created genetic programs. For example, it would be possible to combine Gabor filters based segmentations with the Haralick feature based segmentations.

From experimentation performed, it has been determined that the genetic programming based segmentation method is on average more accurate than the use of grey level co-occurrence matrices with the K-Means algorithm, even over a very large and varied set of test images (see chapter 6 for more details on the results). Due to the feature-selection type nature of the method discussed, the genetic programming based method also makes more efficient use of features, potentially resulting in lower execution times.

1.4 Plan of the dissertation

The rest of the dissertation is organised as follows:

Chapter two contains a discussion of the current literature relating to this line of research. The current methods of classification and feature extraction relating to document image segmentation are then discussed in the third chapter.

Genetic programming is discussed in the fourth chapter, which is followed by a description of the implemented systems in chapter five. In chapter six, the methodology of the experiments is discussed, the results of the experiments are then presented and compared with other methods of document image segmentation.

The final chapter states the conclusions of this dissertation and discusses possible future work that would serve to further expand this line of research, as well as the limitations of the implemented genetic programming based system.

Chapter 2

Literature survey

In this chapter, applications of genetic programming that share characteristics with the genetic programming system to be developed are discussed. As far as has been established, there has been no application of genetic programming in previous works that takes the same approach to image segmentation as the system that has been presented in this thesis.

In terms of the scope of this study, a document is any printed or written item, as a book, article, or letter. The primary aim of document image processing is to recognise the graphics and text components in a document and to extract the required information from the text and graphic components[49]. Most of this processing is the type of processing that humans automatically perform when viewing a document. Such processing includes determining whether an area of a document is part of the text component or the image component of the document, recognising characters in the document and classifying text scripts [22].

At the moment, the field of document image processing is very well established. There is a large variety of applications that fall under the document image processing category, many of which can be performed by a variety of methods[49]. This includes applications such as:

(1) Text script recognition[22].

(2) Optical character recognition[8, 86].

(3) Document image segmentation[31].

(4) Text and background classification[31, 88].

(5) Circuit diagram segmentation (into line drawing and text components)[14].

(6) Document cleaning and restoration[92].

The field of document image analysis has two major categories; textual processing (which processes text components of document images) and graphical processing (which deals with the document's non-textual graphic components). Textual processing consists of finding bodies of text, determining text skew and recognizing the text via OCR[49], as well as possibly improving the OCR output by comparing it to prior knowledge of the document's language[77]. Graphical processing largely deals with tasks such as the recognition of non-textual line and symbol components within diagrams and tables as well as the discovery of pictures in documents.

2.1 Image segmentation

The segmentation of an image is a computer vision process involving the partition of an image into a number of regions according to a certain criterion of homogeneity. The goal of segmentation is generally to locate certain regions of interest in an image (such as regions of text or human faces). An example of a basic segmentation is the thresholding of an image into two particular sets of pixels based on pixel value. This would result in two different segments of the image.

Document image segmentation is image segmentation specifically carried out on document images. This form of segmentation occurs on two levels. On the first level, if the document contains both text and graphics, these are separated for subsequent processing[35, 96]. In general, segmentation algorithms are too basic to perform this type of segmentation perfectly. However, these algorithms are usually very general, predictable and efficient. On the next level, the text itself is segmented by locating columns, paragraphs, words, and characters. On the graphics segment, additional segmentation usually includes separating symbol and line components. A typical approach to document image segmentation is shown in figure 2.1.

2.2 Feature extraction

In the field of image segmentation, feature extraction is the process of simplifying the raw data of an image (the image pixels) into a smaller set of data which is more useful and manageable for subsequent steps of the image segmentation process. Feature extraction generally reduces the dimensionality of the data, which is important in data analysis since a smaller amount of more relevant data is made available[89].

Gabor filtering is a feature extraction technique that can be viewed as a sinusoidal wave

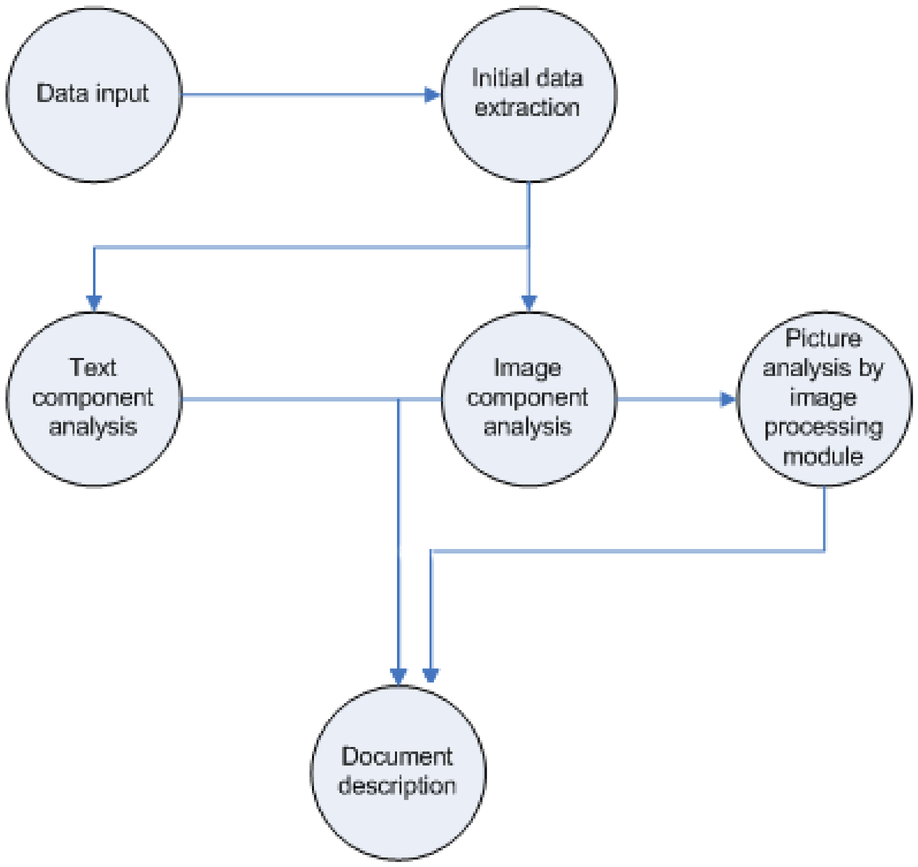


Figure 2.1: *The above diagram shows a typical layout of a document image processing system. The initial data extraction refers to any necessary binarisation, thresholding, noise reduction, or segmentation. After this has been done, the document is split into its image and text components, which are then processed separately. Once their processing is completed, all of the document's information is analysed[65].*

of a particular frequency and orientation modulated by a 2-D Gaussian function[9]. Gabor filters have recently been given a lot of attention due to the fact that the characteristics of certain cells in the visual cortex of some mammals can be closely approximated by these filters[9]. This method is ideal for use on document images as its filtering is highly directional and is performed along several different angles and frequencies. Gabor filtering (see fig. 3.2, in section 3.3.3) is useful because text is highly directional and filtering along particular angles can thus be very helpful for determining whether something is or is not text. Gabor filters have also been shown to possess optimal localisation properties in both the spatial and frequency domains and hence they are well suited for the purpose of texture segmentation[9].

Once filtered, the image can then be used for image segmentation. Sometimes thresholding the filtered image or merging the filtered image with other filtered images achieves better results.

This technique has been extensively used for target detection, fractal dimension management, document analysis[22], edge detection, retina identification[10], image coding and image representation[9, 19, 47], and clearly provides a powerful representation space for describing image textures[17].

The discrete wavelet transform, transforms discrete signals into a series of wavelets with a range of different frequencies and coefficients. This method has been used for purposes such as cleaning images, image compression, feature extraction and texture classification. Wavelet transforms have been used in many applications in document image processing; from removing document image interference[92] to segmenting images entirely[73]. Wavelets have the important property of describing patterns and repetitions in a document easily in the feature space that the document is converted to. This makes wavelets a very useful tool for texture segmentation.

One of the more frequently used methods for feature extraction in document images is Grey Level Co-occurrence Matrices (GLCM). This method is a part of the statistical group (as opposed to the less popular structural group) of texture analysis algorithms. Once a GLCM is calculated, Haralick features can be determined from it [29, 41]. These features include measures such as contrast, energy, correlation, inverse different moment, inertia, mean and variance. GLCM's and Haralick features have been successfully used for the texture based segmentation of colour images of blood smears on slides for the purposes of automated differential blood counts[81], images of irises[10] as well as many other kinds of images[60, 62].

2.3 Classification techniques

Data classification is the process of objects, or data, being grouped into classes based on some criteria. Classification is often used on the results of feature extraction algorithms in order to perform segmentation[58]. K-Means is a clustering algorithm which classifies points of data, in any number of dimensions, into a specified number of groups (K). The K-Means algorithm clusters points in the feature space by calculating the closest centroid (by a distance calculation) to each of the points to be clustered. The positions of the centroids are determined by the K-Means algorithm. Centroids to cluster data around are most commonly initially chosen at random. These centroids then move according to the average position of the closest points around them. Once the centroids no longer change position, the algorithm has converged and the data is considered clustered[13, 24, 57]. The problem with the K-Means algorithm is that it often gets stuck at local minima and the result is highly dependent on the initial choice of the prototypes. The K-Means algorithm also has a fuzzy version called the C-Means algorithm which in some applications will provide far better results due to its fuzzy nature[24]. Haralick features calculated from GLCM are frequently used with the K-Means algorithm for the purpose of segmenting the image in question. As experimental results have shown, this can be a fairly accurate way of segmenting an image into K segments[10, 56].

In several works, connected component analysis has been applied to a document image in order to segment images and text in documents. Once the connected components have been found, a set of rules is then used to determine which regions are part of the text component and which are a part of the image component. This technique has proven to be very robust and efficient during experimentation[35, 52]. Markov Random Fields (henceforth referred to as MRFs) constitute a machine learning technique that is used for signal classification. This technique has been used for the purposes of feature extraction and texture classification[10]. MRFs have also successfully been used for combining segmentations by colour methods and texture analysis based methods[50].

Neural networks may also be used as a texture recognition method. Neural networks deal with texture segmentation by determining the differences between the characteristics of the texture fields. The accuracy of the segmentation depends upon the discriminating power and robustness of the neural network. Developing a neural network that would work well enough to satisfy these two criteria is a difficult problem and is the major flaw of this method[98]. The segmentation method using split and merge, or pyramid linking has similar difficulty in the definition of a discriminant function which measures the homogeneity among the characteristics of the textures[98].

2.4 Document image segmentation systems

As an example, a standard document image segmentation system would take input in the form of a document image. The system would then run a feature extraction algorithm on the document image to summarise the data into a more manageable and useful form. The new feature space would then be passed on to a classification algorithm (such as the C-Means or K-Means algorithm), and then the classified data would be passed onto a post-processing algorithm in order to make use of the domain specific rules applying to document image segmentation.

An example of a more complex non-genetic document image segmentation system is that of *Dong et al*[31]. Their method takes a multi-resolution approach with block sizes of 16x16, 32x32 and 64x64, and then determines the probability of each block (at each scale) belonging to each class. Their system then passes that data onto a sequential Maximum *A Posteriori* estimator (MAP, a solution to a Bayesian estimation which aims to maximize the probabilities that all regions are correctly classified) combined with joint multi-context and multiscale (JMCM, as proposed in [33]) segmentation. The method of combining SMAP with JMCMS is performed in the same manner as done previously by *Cheng et al.* in [18].

2.5 Evolutionary methods

Genetic algorithms were formally introduced in the 1970s by John Holland[44] at the University of Michigan. John Koza later introduced genetic programming[53] which evolves a program or algorithm to be followed, usually with a different representation to genetic algorithms. Koza initially used Common Lisp; this was due to the ease with which programs can be defined in terms of functions, since LISP is a functional programming language[91].

Genetic programming (see section 5.1 for more details) is inspired by Darwin's theory of evolution. In this methodology, problems are solved by an evolutionary process, resulting in the best solution that has evolved through a process of survival of the fittest[64].

The continuing price/performance improvements of computational systems have made genetic programming attractive for many optimisation problems. Genetic programming is less susceptible to converging at local optima than many other search methods. However, it tends to be computationally expensive[76, 91]. Genetic systems, as search techniques, are effective at pruning the examined search space (in contrast with more brute force search techniques)[25].

The natural selection used in genetic programming is based upon the fitness of the individuals. The less fit individuals are generally the most likely to be removed. The fitness (or objective) function is a function that evaluates individuals, giving them fitness ratings. The fitness function is possibly the most important part of a genetic system as it dictates which possible solutions are replaced and which continue to evolve via the genetic operators (such as genetic crossover or mutation)[72, 76, 91].

The representation of genetic programs generally takes the form of a tree. These trees consist of branch nodes (referred to as function nodes) and leaf nodes (referred to as terminal nodes). Leaf nodes and branch nodes are collectively referred to as primitives[42, 54, 64, 72].

Function nodes usually take on the meaning of particular functions (such as addition, multiplication and subtraction). The immediate child nodes of a function node are the parameters of the function. For example; a function node representing addition would typically have two children, which the function node would add together to produce a result. The terminals in a tree would generally consist of input variables or constants (such as pi, X, Y etc.)[42, 54, 64, 72].

Evolutionary methods have recently become more widely used for many purposes in the field of image processing. Some fields and studies that are related to the field of research in this dissertation follow:

(1) Optical character recognition[8, 86].

(2) Object detection and extraction[54].

(3) Image classification[82].

(4) Medical image segmentation[26].

(5) Evolving wavelets for higher quality encoding of images (by *Grasemann et al* [39]).

(6) On-road vehicle detection schemes[87].

(7) Background removal in document images[36].

(8) General image segmentation[15, 25].

(9) Primitive extraction[78].

(10) Scene recognition[5].

(11) Image interpretation[43].

Genetic programming has also been successfully applied to texture recognition. The type of recognition described by *Song et al.*[84] can be used to recognise exact or very similar copies of the texture that the genetic program has been trained to recognise, however, it is not applicable when it comes to document image analysis since a class of textures are being recognized as opposed to an exact texture. In this dissertation we attempt to solve the problem of recognising subclasses of textures (background, image and text) as opposed to specific textures.

In [85] a system is developed which uses memetic algorithms for partitioning genes and gene products according to their known biological function based on genetic ontology. Memetic algorithms are a combination of genetic algorithms and local searches[97] and have been proven to be more efficient than genetic algorithms in certain problem domains[61].

In this study we use the K-Means algorithm to provide training data for the genetic system. The use of K-Means with genetic programming appears in a number of studies. For example: *Maulik et al.*[13] have implemented a system for clustering data via a genetic programming system based upon the K-Means algorithm (the KGA-clustering algorithm), which could have useful applications in the field of image segmentation (although in the study, an image processing application wasn't explored). Genetic programming combined with the K-Means algorithm has also been used for navigation systems[80]. In this application, the K-Means algorithm is used to cluster possible traversal nodes. The genetic programming is used for calculating routes. K-Means is used for partitioning numeric attributes in a genetic programming system for project organisation in[20]. Genetic programming has also been used as a way of benchmarking K-Means initialization methods; this is done by drawing up the probability distribution of the square error values of the resulting clusters from the K-Means algorithm and approaching its extremes by genetic programming. The number of iterations required for the K-Means algorithm to converge are then computed[67].

K-Means and genetic programming has also been used to segment colour images by partitioning the colour cube using the K-Means algorithm with genetic programming[74]. In [15], Bhanu, Lee and Ming propose a learning technique for image segmentation using genetic programming. The implemented system (for the purpose of document image segmentation) allows the segmentation process to adapt to different environmental effects such as lighting and weather conditions. The system implemented in this study differs in that it makes no use of colour information, instead, the textures are recognised via their statistical attributes.

In [25], *Chun et al.* present an image segmentation methodology using genetic programming based on a random search and a parallel test-and-go technique. The algorithm provides a method for image segmentation that does not require threshold values or critical parameters. The method in the implemented system differs in that it does not use genetic programming for the purpose of creating programs which combine a variety of statistical attributes of textures for segmenting (possibly) multiple document images.

Chapter 3

Image segmentation methods and techniques

In this chapter various classical methods and techniques for feature extraction and classification are discussed.

In our context, image segmentation (as illustrated in figure 3.1) is performed as follows:

(1) The input image is converted into a set of features known as a feature space.

(2) The feature space is processed by a classifier to determine an initial segmentation.

(3) The initial segmentation undergoes post-processing to determine the final result.

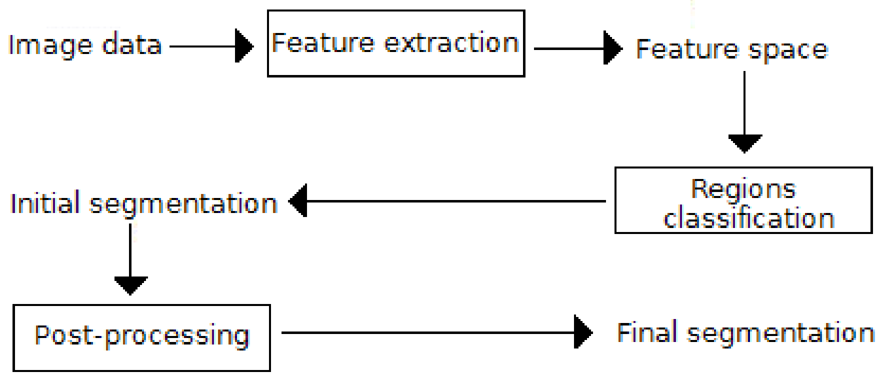


Figure 3.1: *The general operation of our texture-based image segmentation system.*

3.1 Texture Segmentation

Texture based segmentation is the process of segmenting an image based on the characteristics of its textures. Texture is an important property of images in computer vision. In document images, medical images, satellite images and many other types of images, different regions have different textures. As a result, texture based segmentation is a very important field as it allows one to extract and analyse particular regions of interest from images. Texture segmentation is generally considered to be an optimisation problem[55]. In this dissertation, a texture based approach is taken to the image segmentation problem.

Segmenting images by texture accurately is not a trivial task. There are various problems that can occur, for instance:

(1) Conventional edge detection algorithms have problems detecting the borders of textures since textures are made up of intensity fluctuations, which often give false positives to edge detectors. Also, texture boundaries do not appear the same way as conventional edges, so the texture edges often go entirely undetected by conventional edge detectors[41, 55].

(2) Problems with texture segmentation arise from the spatial characteristics of textures, such as determining the class of a texture near the boundary[55, 95].

(3) Large texture sample windows provide more accurate information about what class a texture belongs to. However, texture boundaries become more of a problem when it comes to determining the texture's class, and finding the boundary position. Smaller windows are better at determining border positions, but classification accuracy is lost. This problem is known as class position uncertainty[95]. One possible solution to this problem is to sample the texture with different window sizes[55].

3.2 Feature selection

In principle, more information is better when it comes to clustering data, when working under the assumption that there is nothing known about the representativity of sets of data. It would seem that one should use as many features as possible to represent a pattern. However, some features can simply contain noise as far as clustering is concerned and thus degrade the performance of the clustering algorithm. Also, the more features being clustered, the slower the algorithm runs[59].

Feature selection attempts to select the best possible subset of features out of a given group of features for clustering. The process of feature selection results in more accurate (since noisy features can degrade the performance of learning algorithms[75]) and more economical (in terms of storage and computation) classifiers[59].

Feature selection is particularly important for data sets with large numbers of features, such as problems in molecular biology may involve thousands of features [12], clustering web pages or other documents represented by key-terms[59].

3.3 Feature extraction

In our context, feature extraction is the process of transforming a feature space into a more manageable format, which is then used by a second algorithm to perform an image segmentation process (such as segmentation or pattern recognition). There are a large number of techniques that can be used to perform feature extraction on images. Some of the most frequently used texture based methods are covered in this chapter. They include wavelet transforms, Markov random fields, Gabor filters and GLCM's.

3.3.1 Markov Random Fields

Markov Random Fields constitute a machine learning technique that has been used for the purposes of texture classification and feature extraction[10]. Markov networks are similar to Bayesian networks in the manner in which they represent dependencies in the networks. However, Markov networks are able to represent dependencies that Bayesian networks are unable to, such as cyclic dependencies[27]. Markov networks were first used in the field of computer vision by Geman and Geman[38]. Markov random fields have been used for :

(1) Combining segmentations by colour methods and texture analysis based methods (such as Gabor Filters)[50].

(2) Image restoration[38].

(3) Segmentation of 3D scan data[4].

(4) Other applications such as post-processing[99].

Formally, Markov random fields model the joint probability distribution (as shown in eq. 3.1 and 3.2) of a set $\mathcal{X}$ of random variables and consist of:

(1) An undirected graph G = (V,E).
Each vertex v ϵ V represents a random variable in $\mathcal{X}$.
Each edge (u,v) ϵ E represents a dependency between the vertices u and v.

(2) A set of factors ϕ_k, one for each clique k in G. Each ϕ_k is a mapping from possible joint assignments (to the elements of k) to non-negative real numbers[27].

The joint distribution represented by a Markov network is given by:

$$P(X = x) = \frac{1}{Z} \prod_{k} \phi_k(x_{\{k\}}) \tag{3.1}$$

$$\text{where } Z = \sum_{x \epsilon \mathcal{X}} \prod_{k} \phi_k(x_{\{k\}}) \tag{3.2}$$

Z is the normalising constant and $x_{\{k\}}$ is the state of the random variables in the k^{th} clique.

The Markov blanket of a node v_i in a Markov network is every node with an edge to v_i (formally, all v_j such that $\{v_i, v_j\}\epsilon E$). Every node v in a Markov network is conditionally independent of every other node given the Markov blanket of v[27].

3.3.2 Discrete Wavelet Transforms (DWT)

The first DWT was discovered by the Hungarian mathematician Alfred Haar[40]. The discrete wavelet transform, transforms discrete signals into a series of wavelets with a range of different frequencies and coefficients.

The discrete wavelet transform has many applications in science. In particular, it is used for signal coding. This is done by representing a discrete signal in a more redundant form, often as a preconditioning for data compression (eg. JPEG compression). Wavelet transforms have been used in many applications in document image processing; from removing document image interference[92] to segmenting images entirely[73]. Wavelets have the important property of describing patterns and repetitions in a document easily in the space that it converts images to. This makes wavelets a very useful tool for texture based segmentation.

A wavelet function ψ (see eq. 3.3) can be used for representing and approximating functions by superimposing translated and dilated versions of ψ. The translated and dilated versions of ψ are denoted by $\psi_{j,i}$, where i and j are the translation and dilation parameters respectively. In image processing, the discrete case is generally used, where i and j only have integer values[39].

$$\psi_{j,i} = 2^{j/2}\psi(2^j x - i) \tag{3.3}$$

Decomposing a function into coefficients for each (j, i) pair is known as a wavelet transform. In the case of the components of the (j, i) pairs taking on integer values, the transform is known as a discrete wavelet transform (henceforth referred to as DWT). When computing the DWT of a function f, it is necessary to find a wavelet coefficient $\Upsilon_{j,i}$ for each j, i pair such that:

$$f = \sum_{j,i} \Upsilon_{j,i}\psi_{j,i} \tag{3.4}$$

$$\Upsilon_{j,i} = \prec f, \psi \succ = \int_{-\infty}^{\infty} f(x)\bar{\psi}(x)dx \tag{3.5}$$

It is required that the wavelet ψ is orthogonal, or that there exists a wavelet $\bar{\psi}$ which is the inverse of ψ. The inverse wavelet can then be used to determine the coefficients of the wavelet transform. The original wavelet can then be used to calculate the inverse DWT[39] (see equation 3.5).

3.3.3 Gabor Filtering

A Gabor function is a sinusoidal wave modulated by a 2-D Gaussian function. The algorithm applies the Gabor function to an input image, and a filtered output image is returned (for example fig. 3.2). Gabor filtering is a highly directional filtering technique. Since text is highly directional, these filters are very useful when applied to text extraction and recognition processes[46, 93].

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}) \cos(2\pi\frac{x'}{\lambda} + \psi) \tag{3.6}$$

$$\text{where } x' = x\cos\theta + y\sin\theta \tag{3.7}$$

$$\text{and } y' = -x\sin\theta + y\cos\theta \tag{3.8}$$

In equation 3.6:

(1) λ is the wavelength of the cosine factor.

(2) θ is the orientation of the normal of the parallel stripes of the Gabor function in degrees.

(3) ψ is the phase offset of the filter in degrees.

(4) γ is the aspect ratio of the Gabor function.

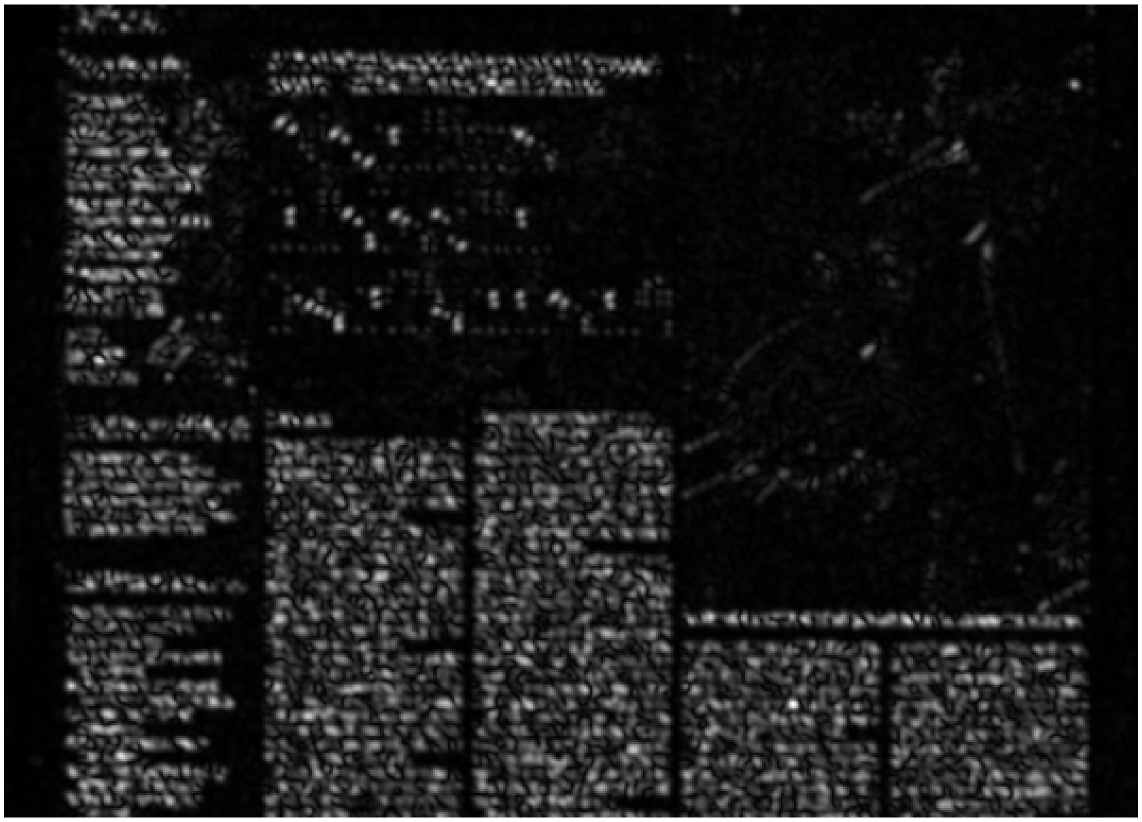


Figure 3.2: *Note the differences between the textual and non-textual regions of the image. This is due to Gabor filters (and text) being highly directional. This makes Gabor filters extremely useful as far as text extraction is concerned. The original image (figure C.1) can be found in appendix C.*

3.3.4 Grey Level Co-Occurrence Matrices and Haralick Features

The use of Haralick features extracted from GLCMs is currently one of the commonly used methods for texture analysis. This method is a part of the statistical group (as opposed to the less popular structural group) of texture analysis algorithms. Haralick features include measures such as contrast, energy, correlation, inverse different moment, inertia, mean and variance.

The way this is used upon an image is to segment the image into blocks of a given size and then individually assess the attributes of each block. This can then be used to perform texture based segmentation on the whole image. This is one of the techniques that will be used along with the genetic system.

Grey Level Co-Occurrence Matrices (GLCMs)

A co-occurrence matrix is defined over an image to be the distribution of co-occurring values at a given offset. This technique is mainly used in the process of measuring texture in grey scale and colour images[29, 41]. Let I_m be an image defined as follows:

$$\text{F: } I_m = \{P_{ij} \in \{0, 1, 2, ..., 255\}, 0 \leq i \leq (N-1), 0 \leq j \leq (M-1) \text{ where } N, M \in \mathbb{N}\}$$

Computing Haralick features consists of performing the transformation, f, from I_m to a feature space F which is a L x C matrix of Haralick features, calculated from co-occurrence matrices.

$$I_m \xrightarrow{f} F$$

$$F = (F_{ij})$$

$$F_{IJ} \in F_1 \times F_2 \times ... \times F_p$$

$$F_k \subseteq \mathbb{R}$$

$$\text{for } k = 1..p$$

$$0 \leq I \leq L - 1$$

$$0 \leq J \leq C - 1$$

where $L, C \in \mathbb{N}$

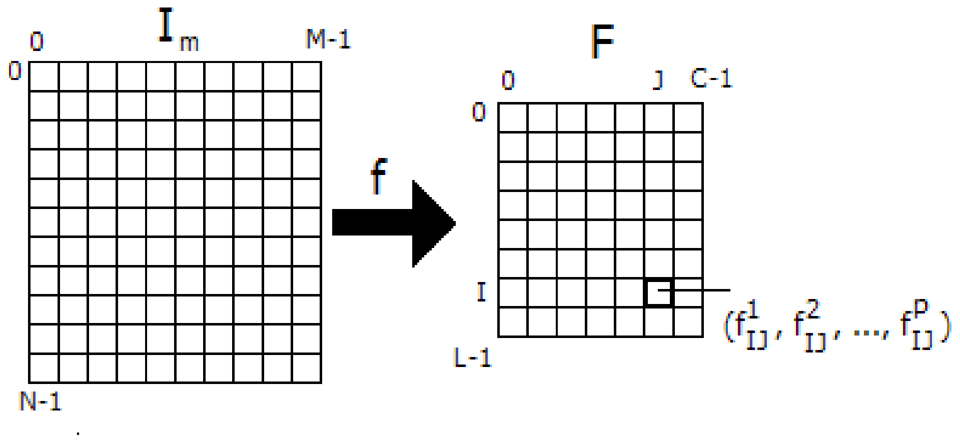


Figure 3.3: *The conversion of an image from the image space (I_m) to the feature space (F)*

The GLCM is generated from a function which observes the joint probability of two pixels with particular grey levels being a certain distance away (d) and in a particular direction (θ). For a given window of size $w \times w$ in an image of G grey levels we have the following matrix:

$$C_{\theta,d} = c_{d,\theta}(i,j)_{0 \leq i,j < G}$$

$c_{d,\theta}(i,j)$ is the frequency of the grey level j following the grey level i in the direction θ at the distance d. The above matrix is the co-occurrence matrix, from which Haralick features can be extracted.

Haralick features

Measures taken from a GLCM are often used to get a more useful feature space. Features generated from GLCMs are generally called Haralick features (named after Robert Haralick)[41, 29]. A list of some of these features follows:

Contrast (CON): This measure is used to evaluate the coarseness of a texture, and is also known as inertia. It is generally very useful for determining whether or not a region is text, since text is usually very high contrast.

$$CON = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(i-j)^2 p_{\theta,d}(i,j) \tag{3.9}$$

Entropy (ENT): Indicates the degree of non-homogeneity in a texture.

$$ENT = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} p_{\theta,d}(i,j) log[p_{\theta,d}(i,j)] \tag{3.10}$$

Energy (ENR): Indicates the degree of homogeneity in a texture.

$$ENR = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(p_{\theta,d}(i,j))^2 \tag{3.11}$$

Maximum Probability (MP): The maximum probability of a co-occurrence in a texture. This is useful for checking uniformity.

$$MP = \max_{(i,j)\in G\times G} p_{\theta,d}(i,j) \tag{3.12}$$

Inverse Difference Moment (IDM): Has a maximum value when all elements inside the kernel are equal[28].

$$IDM = \frac{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(i-j)^2 p_{\theta,d}(i,j)}{1+|i-j|} \tag{3.13}$$

Correlation (COR): Measures the degree of correlation of the pixels in the sub image being analysed.

$$COR = \frac{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(i-\mu_x)(j-\mu_y)p_{\theta,d}(i,j)}{\sigma_x\sigma_y} \tag{3.14}$$

$$\text{where } \mu_x = \sum_{i=0}^{G-1} i \sum_{j=0}^{G-1} p_{\theta,d}(i,j)$$

$$\mu_y = \sum_{i=0}^{G-1} j \sum_{j=0}^{G-1} p_{\theta,d}(i,j)$$

$$\sigma_x^2 = \sum_{i=0}^{G-1}(i-\mu_x)\sum_{j=0}^{G-1} p_{\theta,d}(i,j)$$

$$\sigma_y^2 = \sum_{i=0}^{G-1}(j-\mu_y)\sum_{j=0}^{G-1} p_{\theta,d}(i,j)$$

Mean (MN): Measures the average grey level in a texture. This is useful for discriminating between textures based upon average brightness.

$$MN = \frac{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1} f(i,j)}{N \times N} \tag{3.15}$$

Standard Deviation (STD): This measures the deviation from the mean in a grey level texture. Text usually has a high standard deviation. Standard deviation is calculated from variance (VAR).

$$VAR = \frac{\sum_{i=0}^{G-1}\sum_{j=0}^{G-1}(f(i,j)-Mean)^2}{N \times N} \tag{3.16}$$

$$STD = \sqrt{VAR} \tag{3.17}$$

3.4 Conventional Classification

The purpose of feature extraction in document image segmentation is to extract more meaningful and reduced information from an image. Such information can then be used for classifying regions.

Data clustering entails the discovery of groups of data or the grouping of similar objects. Each of these groups is called a cluster, which are regions in which the density of objects is locally higher than in other regions[67].

There are two main categories of clustering algorithms:

(1) Hierarchical clustering: the clustering is presented in a tree-like form. The root node of the tree represents a cluster containing all points in the space to be clustered. Every descendant node in the tree represents a sub-cluster of the points that its parent node contains.

(2) Partitional clustering: data is split into a number of different clusters.

Clustering algorithms generally take as input a proximity matrix containing the similarities/dissimilarities between all pairs of points, or a pattern matrix, where each item is described by a set of features [67].

Image segmentation can be formulated as a clustering problem [37, 79]. Other applications of clustering include document clustering [45] to generate content related data for information access or retrieval [16], market segmentation[21], as well as applications in medical and biological fields[12, 59].

In the process of document image segmentation, generally the document undergoes feature extraction and then a classification technique is applied to the extracted features to determine which segment each region belongs to. In this section two of the conventional classifiers shall be discussed:

(1) The K-Means algorithm, which is one of the most used data clustering algorithms.

(2) The C-Means algorithm, which is the fuzzy version of the K-Means algorithm.

3.4.1 K-Means

The K-Means clustering algorithm converges quickly and is an effective way to cluster data[48]. The name is due to there being K total cluster centres that result from the algorithm.

This technique can be applied to any application that involves organising large amounts of data into a given number of clusters. The K-Means algorithm is often used as an exploratory data analysis tool. In one dimension, it is a good way to categorize variables into K buckets. With speech understanding it has been used to convert waveforms into one of K categories. It has also been used in the past for optimising colour palettes in 4,8 and 16 bit images. K-Means is also used to segment data as a part of a compression scheme[23, 48].

The K-Means algorithm (see algorithm 1) chooses K random points in the data as the prototypes for the centroids (in as many dimensions as needed). The points in the data that are closest to each of the K centroids are calculated. The locations of the K prototypes are then changed to the average location of all of the vectors that are closest to each prototype. The process of determining which points are closest to which of the K centroids and the relocation of the centroids continues until the K centroids no longer change position. At this point the data points closest to each centroid are considered to be part of that centroids' cluster, and the clustering is complete. This process is illustrated in figure 3.4.

Figure 3.5 shows some post-processed K-Means algorithm based segmentations. See appendix D for more full sized post-processed K-Means algorithm based segmentations. The K-Means segmentations shown in figure 3.5 and appendix D are segmentations that were performed on all of the Haralick features available, all angles (0,45,90 and 135 degrees) and distances 1 and 2.

K-Means initialisation methods

The K-Means algorithm can be initialised in the following ways[67]:

(1) The MacQueen Approach (henceforth referred to as MA) was proposed by MacQueen in 1967. K random seeds are chosen out of the points to be clustered. The rest of the points are assigned to the nearest centroid. The position of each centroid

Algorithm 1 The K-Means Algorithm

```
(1) Choose k initial centroids by one of the initialisation methods. C = $\{c_1, ..., c_k\}$.
(2) For each i $\in \{1, ..., k\}$, the cluster $C_i$ is the set of points in the data points to be
    clustered $X$ that are closer to $c_i$ than they are to $c_j$ for all $j \neq i$.
(3) For each i $\in \{1, ..., k\}$, set $c_i$ to be the centroid of all points in $C_i$. The new position
    of $c_i$ = $\frac{1}{|Ci|} \sum_{x \in Ci} x$.
(4) Repeat steps 2 and 3 until the positions of the centroids no longer change[7].
```

is recalculated after each additional point is assigned to it.

(2) The Forgy Approach (henceforth referred to as FA) was proposed by Forgy in 1965[3]. K random seeds are chosen out of the data to be clustered. The rest of the data is assigned initial membership to its closest seed's segment.

(3) The Kaufman Approach[51] (henceforth referred to as KA) was proposed by Kaufman and Rousseeuw in 1990. This initialisation algorithm is described in detail in algorithm 2.

(4) Random: The points to be clustered are partitioned at random. This is the most common initialisation method due to its simplicity and effectiveness. Random initialisation is used in our implementation.

The KA and random initialisation methods make the K-Means algorithm more effective, and make the K-Means algorithm behave in a more robust fashion. KA also has a more desirable behaviour in that it doesn't result in bad partitioning as often as the other three methods[67]. KA initialisation has also been shown to cause the K-Means algorithm to converge faster on average than the other methods. Random, however, is considered to be the default initialisation method for the K-Means algorithm. This is because of its good performance and very simple implementation.

The application of K-Means to document images

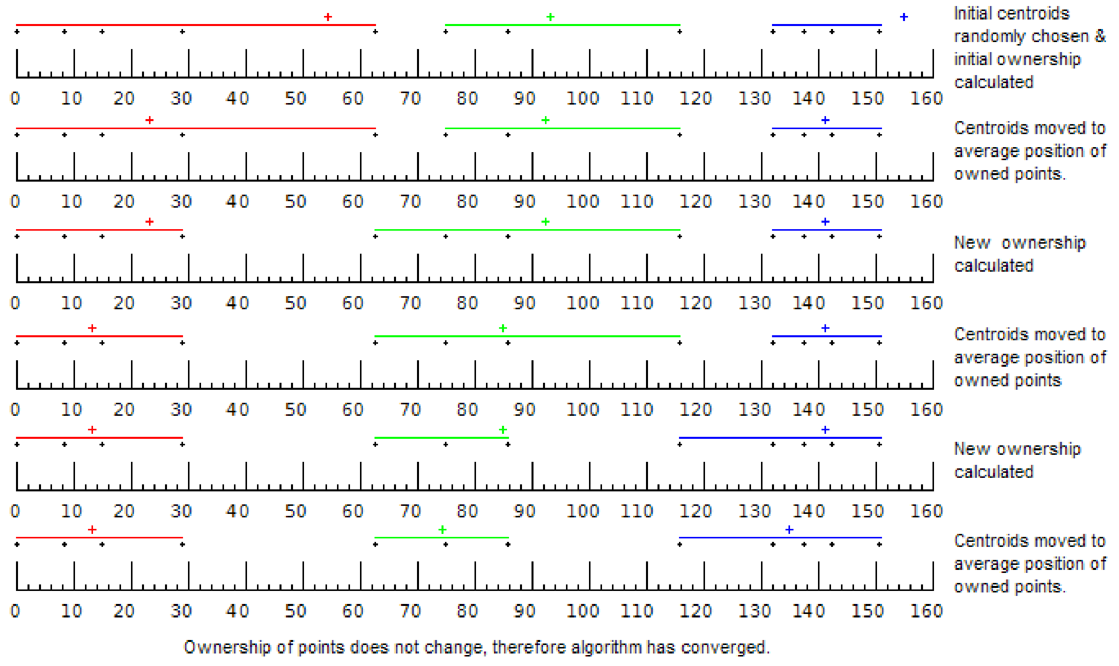

Initial centroids randomly chosen & initial ownership calculated
Centroids moved to average position of owned points.
New ownership calculated
Centroids moved to average position of owned points
New ownership calculated
Centroids moved to average position of owned points.
0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160
Ownership of points does not change, therefore algorithm has converged.

Figure 3.4: *A one dimensional clustering, using the K-Means algorithm.*

One of the approaches to segmenting document images via K-Means is to cluster the feature space into three segments (text, image and background). After this has been performed, post-processing is generally used to clean up the segmentation.

Advantages of using the K-Means algorithm

The advantages of the K-Means algorithm are as follows:

(1) Fast convergence speed (it has been observed that the number of iterations is typically much less than the number of points[6]).

(2) The algorithm is general, and is thus suited to a large number of clustering applications.

(3) The algorithm is fairly simple to implement.

(4) There are a large number of expansions to the K-Means algorithm that can be made

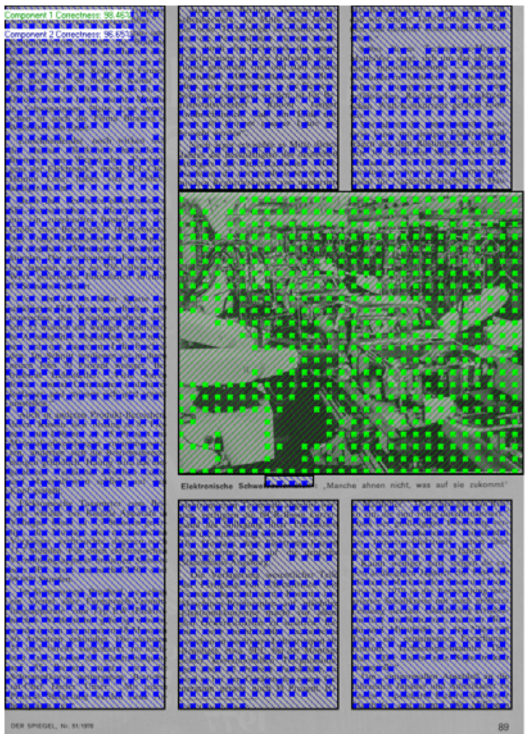
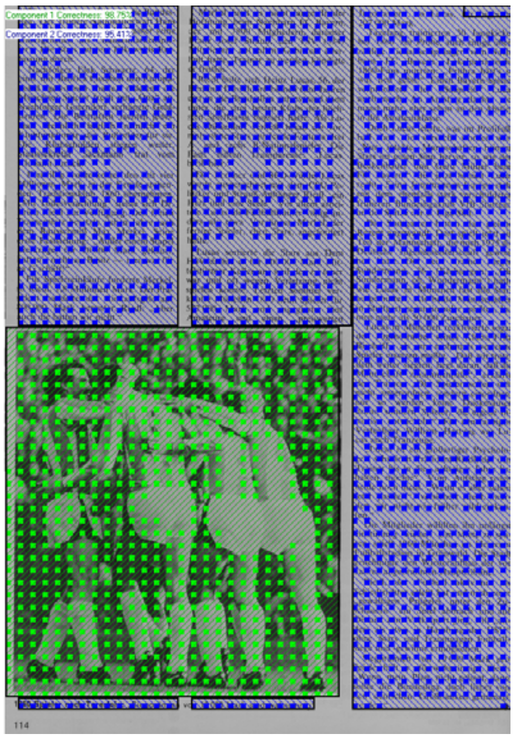



Figure 3.5: *Images 528 and 534 segmented by the post-processed K-Means algorithm. See appendix D for the full sized images and appendix C for the original images.*

in order to make it more accurate or faster[7].

(5) Due to the speed of the K-Means algorithm, it can be run several times on a data set to determine an optimal clustering.

(6) The algorithm works well when data is naturally clustered.

Disadvantages of using the K-Means algorithm

The disadvantages of the K-Means algorithm are as follows:

(1) The algorithm is not guaranteed to return a global optimum and can converge on local maxima[6].

(2) Prior knowledge of a particular application is necessary in order to determine what

Algorithm 2 The KA initialisation method

```
(1) Select the most centrally located point to be clustered as the first seed
(2) For all non-selected points $(w_i)$ DO
    (2.1) Calculate $C_{ji} = max(D_j - d_{ji}, 0)$ where distance $d_{ji} = \| w_i - w_j \|$ and $D_j = min_s d_{sj}$ with $s$ being one of the selected seeds
    (2.2) Calculate the gain of selecting $w_i$ by $\sum_j C_{ji}$
(3) Select the unselected point $w_i$ which maximizes $\sum_j C_{ji}$
(4) Unless there are K selected seeds, Goto (2).
(5) Assign each non selected point to the cluster represented by the nearest seed.
```

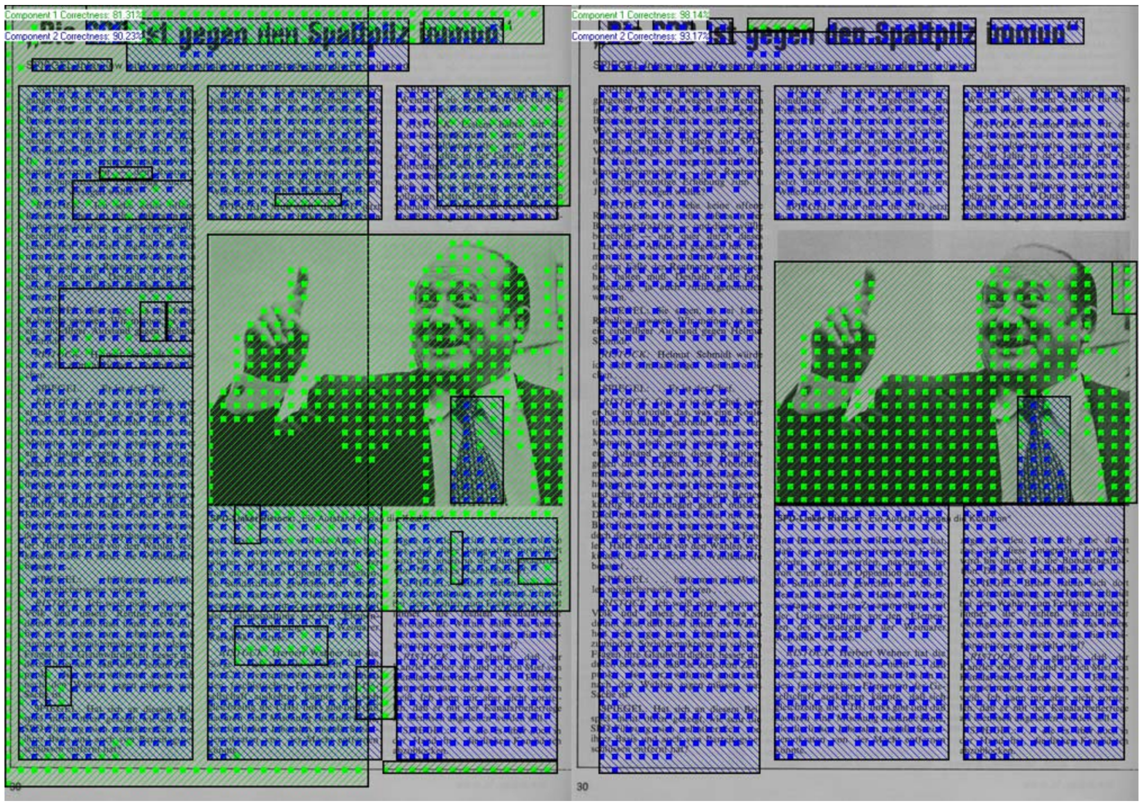

Component 1 Correctness: 81.31%
Component 2 Correctness: 90.23%
Component 1 Correctness: 98.14%
Component 2 Correctness: 93.17%

Figure 3.6: *Image 548 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm. These images demonstrate the inconsistency problems that can occur when using the K-Means or C-Means (see section 3.4.2) algorithms. See appendix D for the full sized images.*

each cluster is (for example: determining whether a cluster is text, image or background).

(3) Since the algorithm needs to know the number of clusters to expect, situations can occur in which there are fewer clusters than expected. This may lead to incorrect results (figure 3.7 demonstrates this problem).

(4) The K-Means algorithm's results depend on initial starting conditions (initial centroids and ordering)[68]. This may cause the K-Means algorithm to return inconsistent results since random centroid selection is one of the preferred initialization methods (figure 3.6 demonstrates this problem).

(5) The quality of the final solution is largely dependent on the initial set of clusters. In practice, the results may be much poorer than the global optimum[6].

(6) If the data is not naturally clustered, the K-Means algorithm may return poor results.

3.4.2 C-Means

The C-Means algorithm is the fuzzy version of the K-Means algorithm. Unlike the K-Means algorithm, this algorithm outputs the percentage ownership of each vector by each cluster. For instance, a region can be 20% a part of region B and 80% a part of region C (see figure 3.8).

The algorithm works in a very similar manner to the K-Means algorithm. Instead of one ownership array (representing the ownership of the vectors in the space being clustered), it has C ownership arrays. The values in the arrays are the percentage of ownership for the vector by a cluster. Figure 3.8 illustrates how partial ownership of points by clusters could be distributed in a single dimension clustering.

The initial centroid positions can be chosen randomly, or by one of the algorithms for seeding the K-Means algorithm (see section 3.4.1). Every round the new locations of the centroids are determined by using the average location of the vectors that are owned or partially owned (in which case the contribution by the vector is weighted by its partial ownership) by the centroid having its position calculated. This causes the C-Means algorithm to produce different output to the K-Means algorithms (given the same starting conditions) since the partial ownership by each cluster is used to calculate new centroid

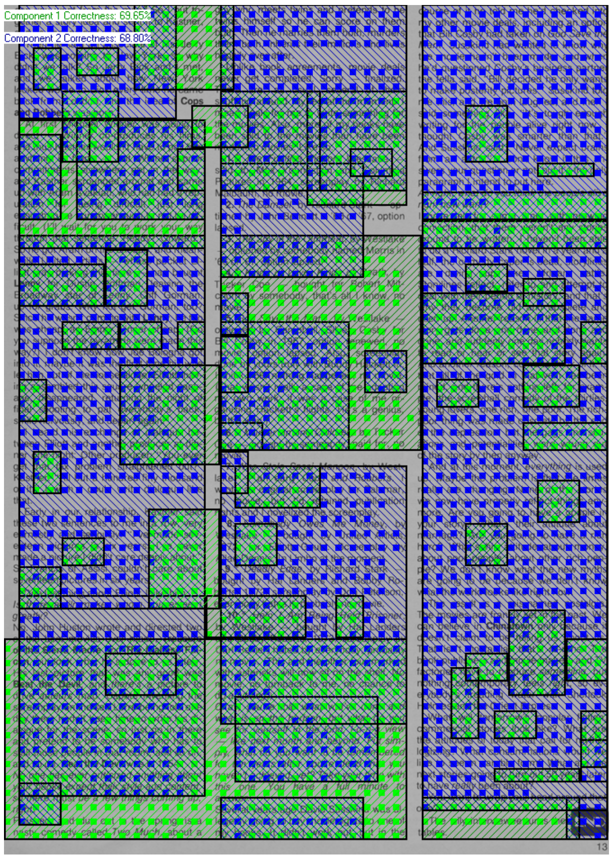

Component 1 Correctness: 69.65%
Component 2 Correctness: 68.80%

Figure 3.7: *This image demonstrates the problem that occurs when attempting to segment two natural clusters (text and background) into three clusters (text, image and background) via the K-Means or C-Means (see section 3.4.2) algorithms. When comparing the results of the post-processed K-Means based algorithm in section 6, five segmentations of the same image are used to determine the post-processed K-Means based algorithms average accuracy. See appendix D for the full sized image.*

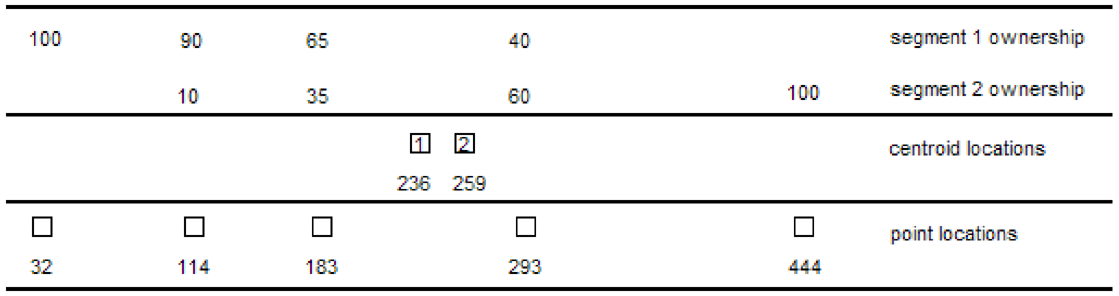


Figure 3.8: *This figure illustrates how the C-Means algorithm uses degrees of ownership instead of absolute ownership (like the K-Means algorithm). The algorithm is executed in the same way as the K-Means algorithm, the only difference in the execution is that degrees of ownership are used to calculate new centroid positions.*

positions. The C-Means algorithm also produces better results, especially when it is an application requiring information with degrees of certainty. The C-Means algorithm runs slower than the K-Means algorithm due to the increased number of operations required for using C ownership arrays to calculate centroids.

This algorithm is used in similar conditions to the K-Means algorithm, especially when an uncertainty value is necessary, or where more accuracy is desired at the expense of computation time. C-Means is described in pseudo-code in algorithm 3.4.2.

Advantages of using the C-Means algorithm

The advantages of the C-Means algorithm are the following:

(1) The algorithm is fairly fast, although slower than the K-Means algorithm.

(2) The algorithm is very general, and is thus suited to a large number of clustering applications.

(3) The algorithm is fairly simple to implement.

(4) There are a large number of expansions to the C-Means algorithm that can be made in order to make it more accurate or faster[7].

Algorithm 3 The C-Means Algorithm

The principle of the C-Means algorithm is the following: for each point x in the space being clustered into C clusters, there is a coefficient ($u_k(x)$) giving the degree to which it belongs in the k'th cluster. The sum of the coefficients for each point is generally 1, so that $u_k(x)$ can be considered to be the probability to which the point belongs to cluster k[66, 90]. The parameter $m(> 1)$ is used as a weighting co-efficient for the fuzzy values. When m is close to 1 the cluster center closest to an examined point is given a lot more weight and the algorithm produces similar results to the K-Means algorithm.

$$\forall x \quad | \quad \sum_{k=1}^{C} u_k(x) = 1 \tag{3.18}$$

$$u_k(x) = \frac{1}{d(\mathrm{centroid}_k, x)} \tag{3.19}$$

$$\mathrm{centroid}_k = \frac{\sum_x u_k(x)^m x}{\sum_x u_k(x)^m} \tag{3.20}$$

where $d(x, y)$ is the distance from the vector x to y.

```
(1) Choose the initial centroids randomly.
(2) Repeat
    (.1) The degree to which points belong to clusters is calculated (shown in equation
         3.19) and then normalised (using equation 3.18)
    (.2) New centroids are calculated (using equation 3.20).
(4) Until (Ownership of points no longer changes)
```

(5) The C-Means algorithm works well when the data is naturally clustered.

(6) The C-Means algorithm produces more accurate results than the K-Means algorithm.

Disadvantages of using the C-Means algorithm

The disadvantages of the C-Means algorithm are as follows:

(1) The algorithm is not guaranteed to return a global optimum and may converge on local maxima.

(2) Prior knowledge of a particular application is necessary in order to determine what each cluster is (for example: determining whether a cluster is text, image or background).

(3) Since the algorithm needs to know the number of clusters to expect, situations can occur in which there are fewer clusters than expected. This leads to incorrect results [for example: an input image in which three clusters are expected (text, background and image), but instead with text and background, but no image].

(4) Due to the random nature of the C-Means algorithm it is not entirely consistent.

(5) The quality of the final solution is largely dependent on the initial set of clusters. In practice, the results may be much poorer than the global optimum.

(6) If the data is not naturally clustered, it may return strange results.

Chapter 4

An introduction to genetic programming

Genetic programming systems are artificial intelligence systems that attempt to evolve solutions to a given problem. In the case of genetic programming, the generated solutions are known as genetic programs. Problems are expressed to a genetic programming system in terms of a heuristic function, which is used to rate individuals for the purpose of applying natural selection, and the possible building blocks of the solution (known as primitives).

A description of the overall algorithm used by genetic programming systems to evolve individuals (genetic programs) is given in the algorithm 4.

Algorithm 4 The steady-state control model

This algorithm is commonly used by genetic systems that follow the steady-state control model. However, there are many variations of each step.

```
(1) Initialize the genetic system.
      – Create the genetic population (or populations) and create the specified number
        of individuals in each genetic population as randomly generated individuals.
(2) Determine the fitness value of every individual in the population.
(3) WHILE (termination criteria are not met [Eg. maximum number of rounds])
      .1 Perform natural selection.
      .2 Create a child by applying the genetic operators to parents selected in (3.1),
         then remove the least fit individual from the population.
      .3 Calculate the fitness of the new individuals created in step (3.2).
      .4 Increment the round count.
(4) Return fittest individual in the history of the population.
```

4.1 Genetic populations

A genetic population is a collection of individuals. Populations may be of a fixed size, or a variable size depending on the control model being used by the genetic system.

4.2 Genetic system control models

(1) **Generational control model**

The generational control model is the most common control model used by genetic systems. The size of the population (or populations) is fixed throughout the run and the number of generations is fixed. In every generation, a new population is created by the application of the genetic operators to the individuals of the old population.

(2) **Steady state control model**

In the case of the steady state control model, the population (or populations) of the genetic system are of a fixed size and there is not necessarily a set number of generations. The system makes use of inverse selection methods for determining which members of the population are replaced[69]. Algorithm 4 shows the basic working of a genetic system using the steady state control model.

(3) **Varying population size model**

In the case of the varying population size model, the population size is not constant. The population (or populations) may shrink or grow to best suit the state of the system. For instance, if the system nears an optimal solution, the population size is decreased. However, if the system nears a local optimum, the size of the population is increased to maintain genetic diversity[69].

4.3 The representation of individuals

Genetic programming systems most commonly encode individuals in the population as parse trees. Other methods of encoding individuals include a linear encoding and a graph encoding. All representations have their advantages in different situations. The graph type representation in particular is suited to applications which involve recursion, looping and more complex control structures[30, 70]. The linear structure has been shown to converge to solutions very quickly, but lacks the complexity of the other representations[69]. The implemented genetic system makes use of parse trees, so parse trees shall be focused on in the discussion of genetic programming.

4.4 The primitives

Individuals in genetic populations are represented as structures composed of building blocks known as primitives. The leaf nodes of the genetic individuals' tree are known as terminals. The valid terminals (as defined by the problem description to the genetic system) in a genetic system are known as the terminal set. The non-leaf nodes of the tree are known as function nodes. The valid function nodes (as defined by the genetic systems problem description) in a genetic system are known as the function set.

It is in terms of the primitives of the genetic system that the possible solutions (genetic programs) must be written. The set of primitives must be chosen to satisfy Koza's[53] closure and sufficiency properties. A set of primitives is said to satisfy Koza's sufficiency property if it is possible to express the solution to the problem in terms of the primitives, however, care should always be taken not to include unnecessary primitives since they will increase the size of the search space. To satisfy Koza's closure property, the function set must accept all possible input from other function and terminal primitives[69].

4.5 The terminal set

The terminal set generally represents input data to the system, constants, functions with an arity of zero, or state variables (variables that are internal to and managed by the genetic program, such as variables for looping).

4.6 The function set

Elements of the function set take as input their arguments (the child nodes of the function node in the tree). The number of arguments that a particular function takes is known as the functions arity (for example: addition would have an arity of 2).

Examples of common functions include:

(1) Mathematical functions such as addition, subtraction, multiplication and division

(2) Loop statements such as ”REPEAT...UNTIL”, ”FOR..DO”

(3) Binary operations such as AND, OR, NOT, XOR

4.7 Fitness cases

In genetic systems, the training set usually comprises a number of input variables and desired resulting values achieved by combining the input variables in a logical or mathematical manner (such as decision making processes or mathematical equation creation). The input values and desired resulting values for a genetic system are collectively referred to as fitness cases. The desired resulting values are often observed values that the genetic system attempts to find a way of emulating in order to solve a particular problem.

4.8 The evaluation of individuals

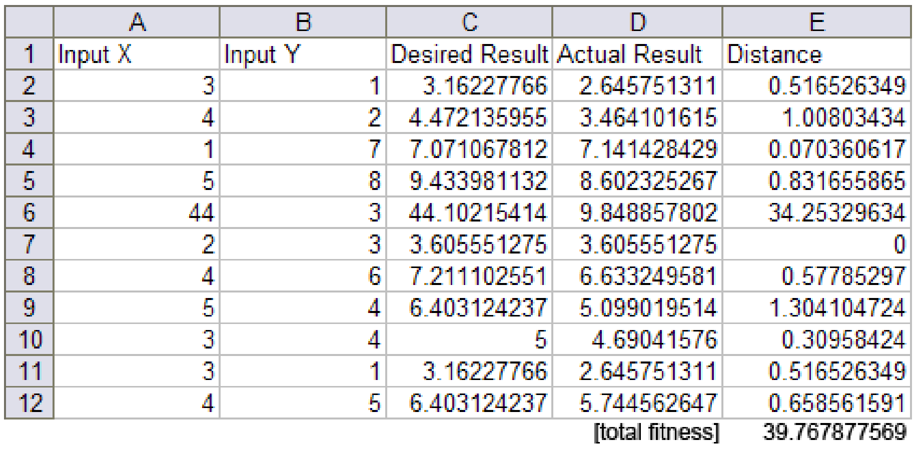
The fitness function calculates the fitness of the individuals in the population by interpreting the individual’s parse tree with all of the available fitness cases[64]. This is done by substituting the values in the terminal inputs in the individual with the corresponding values from the test case being examined, and evaluating the individual by applying the function nodes to the terminal nodes in a traversal (since the terminal nodes now have values from the test case) and then determining the resulting value of the individual. The resulting value is then compared to a target value (in the fitness case) as a part of determining the fitness of the individual.

4.9 The fitness function

When determining fitness, the output of the genetic individual for each set of input data is compared to the corresponding desired result for that input data (see fig. 4.1). The raw fitness can then be calculated in a variety of ways, depending on what is most suitable to the problem. In some cases the distance from the correct answer is used and summed up to represent the fitness (commonly used in mathematical problems, such as in fig. 4.1), whereas in other cases the number of correct or incorrect answers is summed up (commonly used in decision making processes). The following are some of the fitness measures that are used to compare genetic individuals[69]:

- R represents raw fitness.
- MR represents the maximum raw fitness attainable.
- S represents standardised fitness.
- A represents adjusted fitness (see equation 4.1).
- If lower values of R represent more fit individuals then $S(i,t) = R(i,t)$.
- If higher values of R represent more fit individuals then $S(i,t) = MR - R(i,t)$.

$$A(i,t) = \frac{1}{1 + S(i,t)} \tag{4.1}$$



	A	B	C	D	E
1	Input X	Input Y	Desired Result	Actual Result	Distance
2	3	1	3.16227766	2.645751311	0.516526349
3	4	2	4.472135955	3.464101615	1.00803434
4	1	7	7.071067812	7.141428429	0.070360617
5	5	8	9.433981132	8.602325267	0.831655865
6	44	3	44.10215414	9.848857802	34.25329634
7	2	3	3.605551275	3.605551275	0
8	4	6	7.211102551	6.633249581	0.57785297
9	5	4	6.403124237	5.099019514	1.304104724
10	3	4	5	4.69041576	0.30958424
11	3	1	3.16227766	2.645751311	0.516526349
12	4	5	6.403124237	5.744562647	0.658561591
				[total fitness]	39.767877569

Figure 4.1: *An example of fitness being calculated from input values, the desired result and the actual result. The sum of the distances between the desired and actual result over the test cases is used to determine the overall fitness of an individual.*

4.10 Selection and inverse selection methods

There are a variety of methods by which selection and inverse selection can be performed. The method of selection or inverse selection is best chosen with the particular application in mind.

(1) **Tournament selection**

N individuals from the population are selected at random. N is referred to as the tournament size, greater tournament sizes cause greater selection pressure. The fitness's of the individuals in the tournament are calculated and the individual with the best fitness is selected. The rest of the individuals in the tournament are replaced with individuals created via the use of the genetic operators[2].

(2) **Fitness proportionate selection**

For each individual in the population, the probability that the individual will be copied into the next generation can be calculated by equation 4.2 where $f(s_i(t))$ is the adjusted standardised fitness of the individual (see equation 4.1). Individuals that are not copied into the next generation are replaced by new individuals created via the genetic operators[69].

$$\frac{f(s_i(t))}{\sum_{M}^{j=1} f(s_j(t))} \tag{4.2}$$

(3) **Linear ranking**

In the case of linear ranking, there is no fixed number of individuals that are copied forward each iteration. An individual, *Ind*, has a probability, *Pr(Ind)* of being removed that is calculated as follows (where individual fitness rank is expressed as *IFR(Ind)* and number of individuals in the population is denoted *Nbrp*):

$$\Pr(Ind) = \frac{IFR(Ind)}{\sum_{i=1}^{(Nbrp)} IFR(i)} \tag{4.3}$$

For example:

In the case of individuals with the following fitnesses (where lower fitnesses are better): 120,50,66,79,84,150

They would have fitness ranks 6,1,3,2,4,5,7

$(\Sigma_{i=1}^{Nbrp} IFR(i)) = 28$

So their probabilities of being chosen for replacement would be $\frac{6}{28}, \frac{1}{28}, \frac{3}{28}, \frac{2}{28}, \frac{4}{28}, \frac{5}{28}, \frac{7}{28}$, respectively.

4.11 Genetic operations

Once the selection process is performed, individuals in the population are copied forward to the next generation (via application of the genetic operators). The genetic operators are described in the following subsections.

4.11.1 Reproduction

Reproduction (see fig. 4.2) is a straight copy of an individual into the following generation.

(1) The input tree is copied to what will be the output tree (tree C)

(2) Tree C is returned.

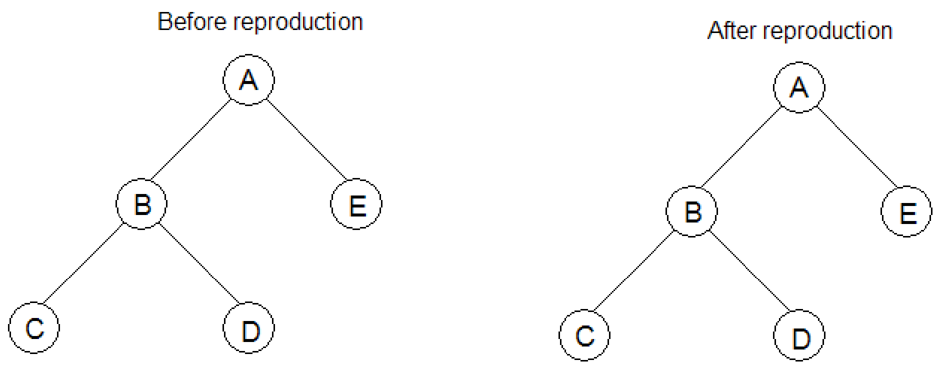


Figure 4.2: *Reproduction takes one individual as input and performs a straight copy into the next generation. It is not necessary to re-calculate the fitness of these individuals.*

4.11.2 Mutation

Mutation is a genetic operator that is used to introduce completely new genetic material into a population. Mutation acts upon a single tree and produces a single tree as output (see fig. 4.3)[2]:

(1) The selected parent tree is copied to what will be the output tree (tree C)

(2) A node from tree C is selected at random.

(3) The selected node and its sub-tree are deleted from tree C.

(4) A new sub-tree is then randomly generated at the position of the deleted node in tree C.

(5) Tree C is returned.

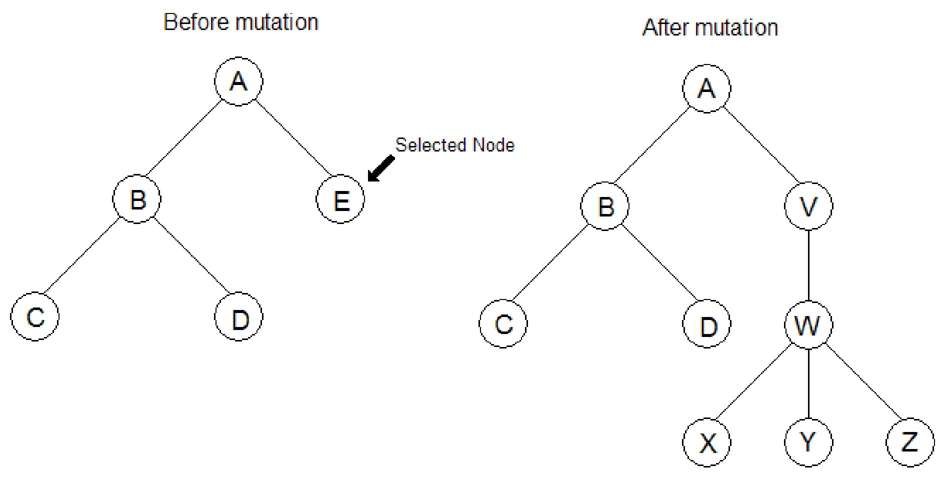


Figure 4.3: *Mutation takes one individual as input and randomly selects a node to undergo mutation. That node and its children are replaced with a random sub-tree.*

4.11.3 Crossover

Crossover generally creates two new individuals from two parent individuals. This operator is the main operator used in genetic programming as it combines individuals that have most likely undergone evolution[2]. Crossover acts upon two parent trees and produces two trees as output (see fig. 4.4):

(1) A node is selected at random from tree A (the first input tree) and tree B (the second input tree).

(2) The selected nodes are swapped between tree A and tree B, yielding output tree C and output tree D.

(3) Tree C and tree D are returned.

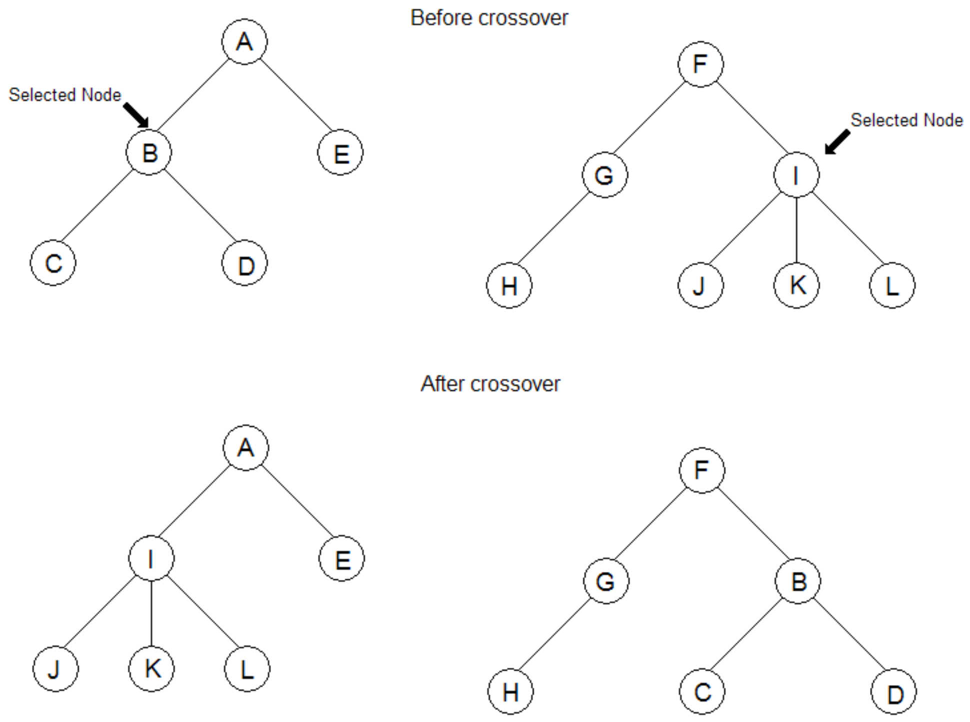


Figure 4.4: *Crossover takes two individuals as input and chooses a random node from each one. The two chosen nodes and their sub-trees are then swapped between the two individuals, giving an output of two new individuals.*

4.12 Termination criteria

The termination criterion of a genetic system is the condition that is required to be met before the evolution of a genetic system legally ends. This condition usually takes the form of a number of rounds, a fitness threshold, or simply if a solution is found (in terms of the Heuristic function).

4.13 Population initialisation

For each population, the specified number of individuals is randomly generated at the start of the run from the terminal and function set (see section 5.1.8 and 5.1.10). The individuals in the population usually have some sort of limit imposed on them in terms of a limit on their number of nodes or depth. The individuals in the populations can be created as follows:

(1) **The full method**

Each individual in the population is randomly generated to its maximum depth. Only function nodes are chosen to form the tree up to the maximum depth. At the maximum depth random terminal nodes are chosen. See figure 4.5 for examples of individuals created using the full method.

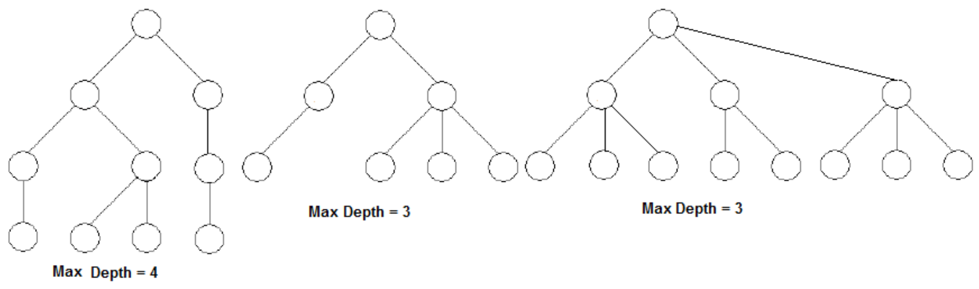


Figure 4.5: *This figure shows individuals created with the full method.*

(2) **The grow method**

Trees generated by the grow method are of random shape and are generated up to the set height limit of the trees (to a maximum depth or number of nodes). Every node in the tree is randomly chosen out of the primitives so any tree shape can occur (within the limit of the set maximum and minimum depths). Nodes at the maximum depth are always chosen as terminals. It is common to use a growth factor of some sort when growing or mutating trees. The growth factor indicates the likelihood that a function node will be chosen or a terminal node. In cases where there are far more possible terminal nodes than possible function nodes, the trees can be expected to be very small if the tree was generated entirely at random out of the primitives, or very large in the other extreme case. Using a growth factor to control the probability of terminal and function nodes being chosen solves this problem. See figure 4.6 for examples of individuals created using the grow method.

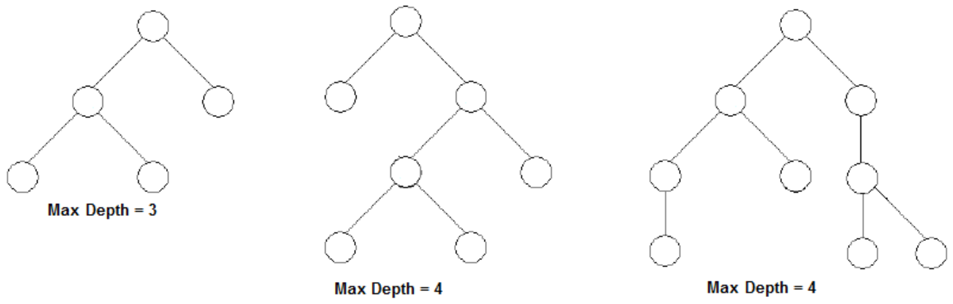


Figure 4.6: *This figure shows individuals created with the grow method.*

(3) **Ramped half-and-half**

The ramped half-and-half method of population generation entails the generation of 50% of the individuals by the grow method, generated with maximum depths split evenly between two and the set maximum depth for the system, and 50% of the individuals generated by the full method, with maximum depths split evenly from two and to the maximum depth for the system. Genetic programming systems using the ramped half-and-half method have been found to, on average, produce better results than genetic programming systems using the grow or full methods of population generation[69].

4.14 Parallel populations

When using parallel populations, the population space is subdivided into multiple subpopulations (or demes). Using multiple parallel populations has been shown to increase the odds of finding a good solution (over using a single population) since it limits the probability of the genetic system getting stuck at a local optima. This is because genetic diversity is maintained by population migrations. In terms of the simulation of actual evolution, this is a slightly more realistic approach. Figure 4.7 shows the fitness over time of four populations running in parallel.

There are a number of ways in which the parallel populations could be generated. The individuals in the populations could be randomly generated with or without certain criteria for each population (for example, certain populations could get a certain subset of primitives).

Population migrations (see fig 4.8) significantly help to decrease the chance that the system will prematurely converge as new individuals are being introduced into the populations that have already undergone evolution (and are thus generally fitter than randomly mutated individuals). The decrease in population stagnation by replacement with good individuals from other populations results in better genetic runs. Population migration can take place once in every few generations, or a set number of times in a run[2]. See algorithm 5 for details.

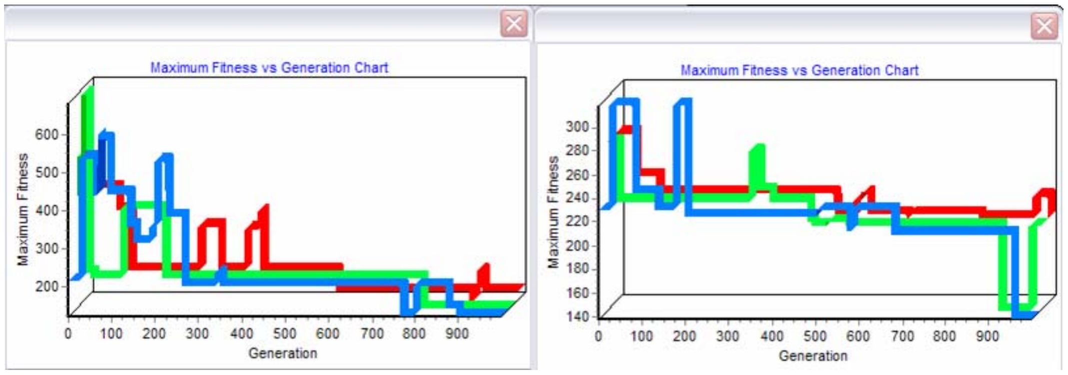


Figure 4.7: *This image shows the max fitness versus generation of three genetic populations running in parallel. The left image shows the top fitness of each population during the genetic run for creating individuals to extract the image segment. The right image shows the top fitness of each populations during the genetic run for creating individuals to extract the text segment.*

Algorithm 5 Population migration can be performed as follows:

```
(1) if ((roundNumber+1) MOD migrationFrequency == 0)

    for (i = 0 to migrationSize)

      .1 Choose two different populations at random (populations A & B)

      .2 Swap a random individual from population A with a random individual from
         population B
```

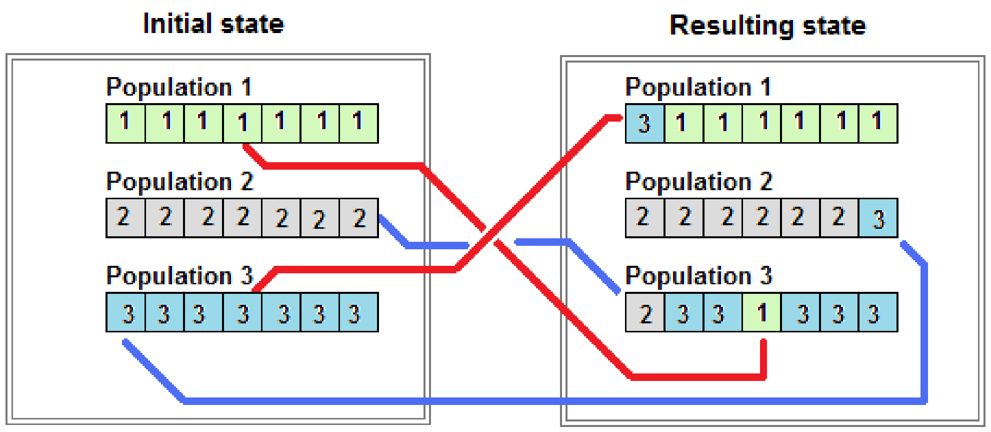


Figure 4.8: *This image demonstrates how individuals could be exchanged between populations via population migration.*

Chapter 5

Systems implemented

5.1 Genetic system

There are several problems with segmenting a document image by clustering the Haralick feature space with the K-Means algorithm. These problems can be improved upon or removed entirely by combining the K-Means segmentations with genetic programs:

(1) Due to the random nature of some of the seeding methods employed by the K-Means algorithm and some other clustering algorithms, the clustering does not always return consistent results. The genetic programs make use of fixed centroids that they have been trained with for clustering data, causing the genetic programs to return more consistent results.

(2) It is difficult to determine which cluster is which in many cases (ie. which cluster is text, which is image and which is background). In some cases it is possible to hard code into the application which cluster is which (for instance high contrast areas in document image segmentation generally represent text). However, for features like the mean, it is a lot more difficult to tell which cluster is which. Keeping the K-Means clustering centroids constant and training genetic programs for recognising particular segments (ie. training one genetic program to segment graphics and another to segment text) solves this problem.

(3) Genetic programming could be used to find the more useful features to use for clustering, thereby eliminating the need to calculate unnecessary Haralick features.

(4) Genetic programming could be used to improve upon K-Means outputs by attempting to find an optimal combination of K-Means segmentations, improving the segmentation accuracy (since noisy features will be eliminated).

(5) In cases where there are fewer clusters than expected, clustering methods that expect a set number of clusters (such as K-Means) can be very unsuitable since such

methods often end up attempting to split one of the clusters. In document image processing, this can occur when the K-Means algorithm expects an image to have three naturally clustered segments (text, image and background) and instead receives an image with only two segments (background and text). In such cases, large parts of the background and/or text regions are often misclassified as belonging to the image region. This issue can be overcome by using trained genetic programs with set clustering centroids. In the case of set clustering centroids, if no data is sufficiently near a centroid, that cluster will simply not exist.

5.1.1 The concept

Some Haralick features are not relevant for segmenting text and/or images. Clustering using irrelevant features is likely to cause inaccuracy. On the other hand, there are features that are good for extracting images (such as mean), while others are better for extracting text (such as variance). In this dissertation, a method is proposed by which the segmentation of image and text are separated to increase segmentation accuracy and the most useful features for segmentation are determined for each segment.

The system combines Haralick features by a number of operators in order to get better results. For instance; one Haralick feature could successfully segment the image component, but at the same time extracting some of the text. This could then be combined with a Haralick feature that extracts text alone. This will then allow us to extract the image segment correctly by removing the text component. The idea is that by combining different features for each component, an extraction of one of the components (for instance text) would yield a far better segmentation.

Algorithm 6 Applying the genetic segmentation algorithm (see section 5.1.3 and figure 5.1 for additional details on input and output):

```
(1) Read the genetic program created by the genetic system to segment the document
    image from the input file.
(2) Perform quick-clustering of each (Haralick feature, angle, distance) triple that is
    used in the genetic program (This is discussed in section 5.1.2).
(3) Input the clusterings used by the genetic program to the genetic system.
      – The genetic program (from step 1) is applied, using the input clusterings as
        input for the relevant terminals. The genetic program returns the resulting seg-
        mentation (which is the segmentation of one component).
(4) Perform step 3 for the second component (the first component is the image compo-
    nent, the second component is text).
(5) Combine the two segmentations. In the case of a region being considered to be both
    a text region and an image region, the region is considered to be 'fuzzy' and is dealt
    with by the post-processor.
(6) Post-process the segmentation.
(7) Return the segmentation.
```

5.1.2 Quick clustering

This clustering is performed to generate input for the genetic image-segmentation algorithm (as shown in step 2 of algorithm 6). The algorithm simply clusters data around the clustering centroids which are stored along with the genetic programs (the same centroids used to segment images for training). The centroids do not change position as in the C-Means or K-Means algorithms. The application algorithm does not use the K-Means algorithm at all since it has set clustering centres. The K-Means algorithm is only used to generate training data for the genetic system.

5.1.3 The training module

The training module (see fig. 5.2) takes as input the training set. The training set consists of human input (the ideal segmentation) and the segmentations of the document image via the K-Means algorithm on single Haralick features (see section A.5.2 for details). As output, the training module produces two individuals; one individual is an algorithm for extracting the image component of a document image, the other component is the algorithm for extracting the text component of the document image (see A.5.3 for details). In order to increase the accuracy of the process and to allow the system more freedom in terms of the operations it may use, the training module performs two separate genetic runs so each run may focus on the extraction of either the text component or the image component.

5.1.4 The segmentation module

For the segmentation of an input image, the algorithm takes as input a genetic program for the extraction of the text component, a genetic program for image extraction, and the Haralick feature space of the image. The algorithm then segments the image accordingly, producing the segmented text component and image component as output. These outputs are then post-processed, which yields the final output of the segmentation (see figure 5.1).

5.1.5 The genetic parameters

Table 5.1 lists the genetic parameters for the system. The system uses the steady state control model, with multiple populations. The populations are generated via the grow method, with ”Mutation population generation growth” specifying the likelihood of a child node

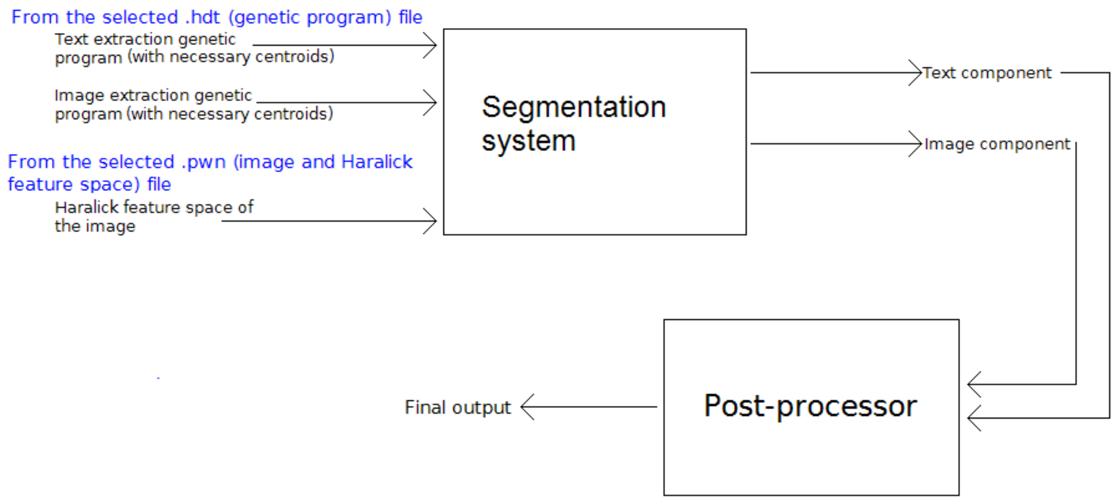

From the selected .hdt (genetic program) file
Text extraction genetic program (with necessary centroids)
Image extraction genetic program (with necessary centroids)
From the selected .pwn (image and Haralick feature space) file
Haralick feature space of the image
Segmentation system
Text component
Image component
Post-processor
Final output

Figure 5.1: *The input and output of the segmentation module.*

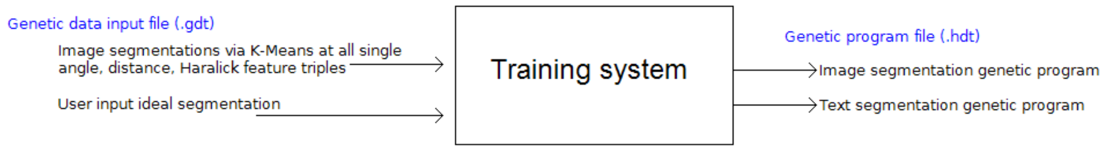

Genetic data input file (.gdt)
Image segmentations via K-Means at all single angle, distance, Haralick feature triples
User input ideal segmentation
Training system
Genetic program file (.hdt)
Image segmentation genetic program
Text segmentation genetic program

Figure 5.2: *The input and output of the training module.*

generated by mutation being a functional node and having children. ”Population migration size and frequency” specifies how often individuals cross over between populations. The termination criteria for the genetic system can be given as a desired fitness (for instance a 1% misclassification rate) as well as a maximum number of iterations. The system stops and returns its results once either of the termination criteria is met.

The default parameters have been chosen with a balance of speed and accuracy in mind. It was found through experimentation with the system that a maximum individual height of 3 or 4 was most efficient in terms of providing good results and lower execution (and evaluation) times. Maximum individual heights of greater than 4 tend to increase the size of the search space drastically, causing great increases in training time. The training system also takes a longer time to evaluate solutions with very large amounts of terminals, due to the number of calculations involved when combining the training segmentations. Populations with maximum heights of less than 3 force the individuals to be too simple, resulting in fewer good solutions in the search space.

A maximum round count of 4000 was found to be long enough for a population of 50 to converge. 50% growth was found to provide a good variety of parse tree shapes. 95% crossover and 5% mutation have been chosen as defaults since the combination tends to provide good results from the system since much more than 5% mutation tends to make the search needlessly random, while less than 5% lacks sufficient introduction of new genetic material.

Table 5.1: ***The default genetic parameters for the system. These values are used as defaults since they provide a good balance between speed and genetic diversity. These default parameters were determined by experimenting with various combinations of parameters. The experimentation was performed with the aim of obtaining good fitnesses in a short timeframe.***

Parameter	**Value**
Crossover %	95%
Mutation	5%
Maximum height	3
Minimum height	0
Termination criteria numrounds	4000 generations
Termination criteria fitness (greater than)	100%
Control model	Steady state
Population initialization method	Grow
Genetic system seed	RANDOM
Maximum clones	3
Number of populations	3 (in parallel)
Individuals per population	50
Population migration size and frequency	1 every 3 rounds
Mutation & population generation growth %	50%

5.1.6 The training set

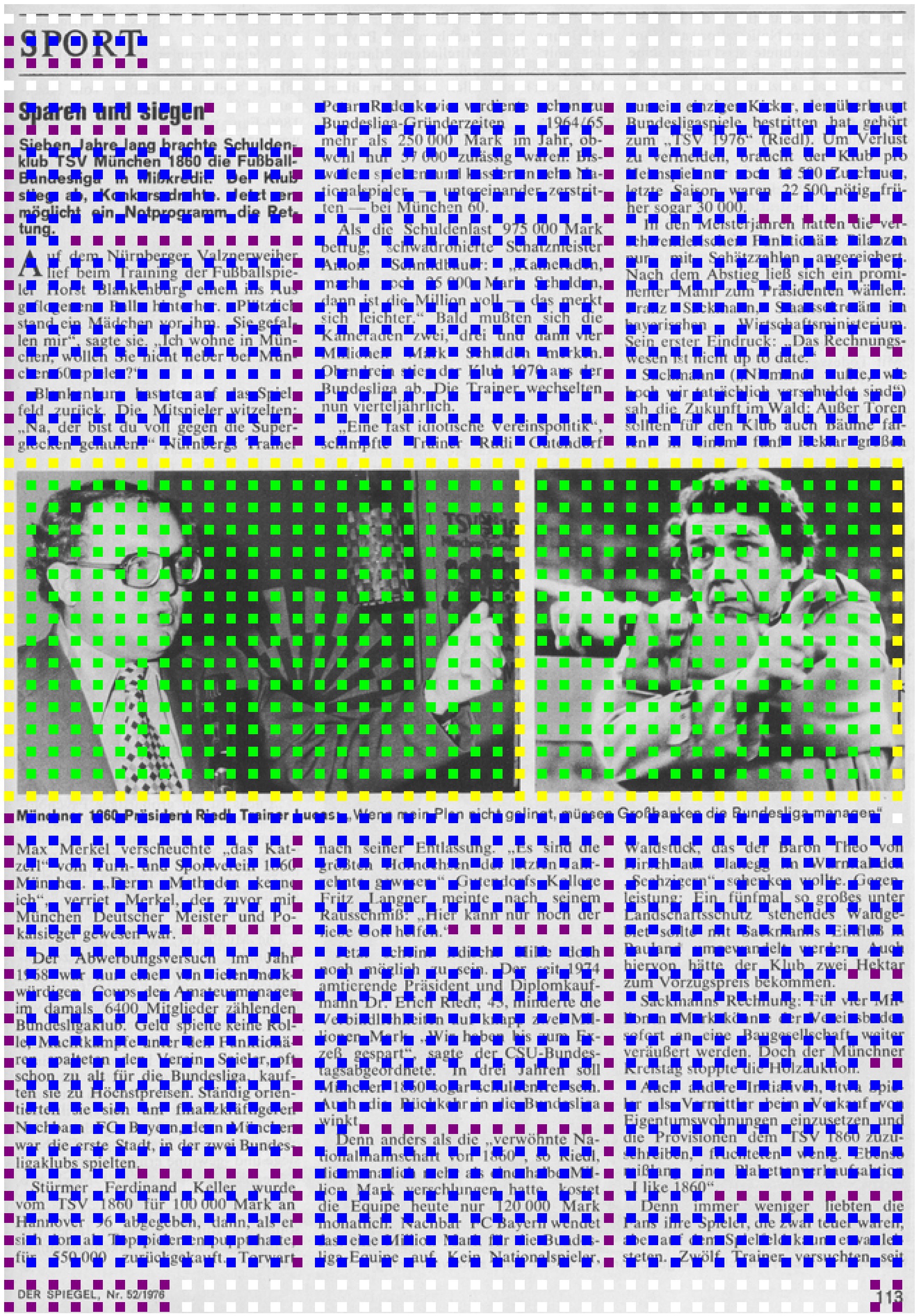
In the case of the implemented genetic system, there are a large number of input values observed. These input values are the results of the segmentation of a training document image into three segments (text, image or background). For the desired result input, the desired membership of each region to a particular segment is used (see figure 5.2). The training set usually consists of a few thousand region classifications, depending on the size of the image used for the training.

The implemented system uses segmentations acquired by clustering combinations of Haralick features, angle and distance using the K-Means algorithm for the training sets' observed values. The desired output from the system for the sets of input is given by the user as described in section 5.1.7. The desired segmentation output values for the training document image are used to determine the fitness of individuals.

5.1.7 Human input

User input(see figure 5.3 for an example of the training) is chosen to evaluate the individuals in the genetic population since text and image regions are human constructed concepts, and the qualitative evaluation of segmentations is performed by the user. As a result, humans would be able to provide better segmentations than document image segmentation software.

The alternative method for creating training data is to use an existing segmentation algorithm. However, there is no perfect solution to the texture based segmentation, and it would be likely that the system would be trained with flawed data, resulting in the system simply attempting to emulate another algorithm along with its flaws. When trained using human expert-input, the system attempts to solve the problem in terms of the way humans segment document images. With human training input, the system is also given a lot more flexibility when attempting to train for specific texture types, such as ones not in the field of document image segmentation.



SPORT

Sparen und siegen

Sieben Jahre lang brachte Schuldenklub TSV München 1860 die Fußball-Bundesliga in Mißkredit. Der Klub stieg ab, Konkurs drohte. Jetzt ermöglicht ein Notprogramm die Rettung.

Auf dem Nürnberger Valznerweiher lief beim Training der Fußballspieler Horst Blankenburg einem ins Aus geflogenen Ball hinterher. Plötzlich stand ein Mädchen vor ihm. „Sie gefallen mir“, sagte sie. „Ich wohne in München, wollen Sie nicht lieber bei München 60 spielen?“

Blankenburg hastete auf das Spielfeld zurück. Die Mitspieler witzelten: „Na, der bist du voll gegen die Superglocken gelaufen.“ Nürnbergs Trainer Max Merkel verscheuchte „das Katzerl“ vom Turn- und Sportverein 1860 München. „Deren Methoden kenne ich“, verriet Merkel, der zuvor mit München Deutscher Meister und Pokalsieger gewesen war.

Der Abwerbungsversuch im Jahr 1968 war nur einer von vielen merkwürdigen Coups der Amateurmanager im damals 6400 Mitglieder zählenden Bundesligaklub. Geld spielte keine Rolle. Machtkämpfe unter den Funktionären spalteten den Verein. Spieler, oft schon zu alt für die Bundesliga, kauften sie zu Höchstpreisen. Ständig orientierten sie sich am finanzkräftigeren Nachbarn FC Bayern, denn München war die erste Stadt, in der zwei Bundesligaklubs spielten.

Stürmer Ferdinand Keller wurde vom TSV 1860 für 100 000 Mark an Hannover 96 abgegeben, dann, als er sich dort als Torjäger entpuppt hatte, für 550 000 zurückgekauft. Torwart Peter Radenkovic verdiente schon zu Bundesliga-Gründerzeiten 1964/65 mehr als 250 000 Mark im Jahr, obwohl nur 57 000 zulässig waren. Bisweilen spielten und kassierten acht Nationalspieler — untereinander zerstritten — bei München 60.

Als die Schuldenlast 975 000 Mark betrug, schwadronierte Schatzmeister Anton Schmidbauer: „Kameraden, macht noch 25 000 Mark Schulden, dann ist die Million voll — das merkt sich leichter.“ Bald mußten sich die Kameraden zwei, drei und dann vier Millionen Mark Schulden merken. Obendrein stieg der Klub 1970 aus der Bundesliga ab. Die Trainer wechselten nun vierteljährlich.

„Eine fast idiotische Vereinspolitik“, schimpfte Trainer Rudi Gutendorf nach seiner Entlassung. „Es sind die größten Hohlköpfe der letzten Jahrzehnte gewesen.“ Gutendorfs Kollege Fritz Langner meinte nach seinem Rausschmiß: „Hier kann nur noch der liebe Gott helfen.“

Jetzt scheint irdische Hilfe doch noch möglich zu sein. Der seit 1974 amtierende Präsident und Diplomkaufmann Dr. Erich Riedl, 43, minderte die Verbindlichkeiten auf knapp zwei Millionen Mark. „Wir haben bis zum Exzeß gespart“, sagte der CSU-Bundestagsabgeordnete. In drei Jahren soll München 1860 sogar schuldenfrei sein. Auch die Rückkehr in die Bundesliga winkt.

Denn anders als die „verwöhnte Nationalmannschaft von 1860“, so Riedl, die monatlich mehr als eine halbe Million Mark verschlungen hatte, kostet die Equipe heute nur 120 000 Mark monatlich. Nachbar FC Bayern wendet fast eine Million Mark für die Bundesliga-Equipe auf. Kein Nationalspieler, nur ein einziger Kicker, der überhaupt Bundesligaspiele bestritten hat, gehört zum „TSV 1976“ (Riedl). Um Verlust zu vermeiden, braucht der Klub pro Heimspiel nur noch 12 500 Zuschauer, letzte Saison waren 22 500 nötig, früher sogar 30 000.

In den Meisterjahren hatten die verschwenderischen Funktionäre Bilanzen nur mit Schätzzahlen angereichert. Nach dem Abstieg ließ sich ein prominenter Mann zum Präsidenten wählen: Franz Sackmann, Staatssekretär im bayerischen Wirtschaftsministerium. Sein erster Eindruck: „Das Rechnungswesen ist nicht up to date.“

Sackmann („Niemand wußte, wie hoch wir tatsächlich verschuldet sind“) sah die Zukunft im Wald: Außer Toren sollten für den Klub auch Bäume fallen. In einem fünf Hektar großen Waldstück, das der Baron Theo von Hirsch auf Hallegg im Wurmtal den „Sechzigern“ schenken wollte. Gegenleistung: Ein fünfmal so großes unter Landschaftsschutz stehendes Waldgebiet sollte mit Sackmanns Einfluß in Bauland umgewandelt werden. Auch hiervon hätte der Klub zwei Hektar zum Vorzugspreis bekommen.

Sackmanns Rechnung: Für vier Millionen Mark könne der Vereinsboden sofort an eine Baugesellschaft weiter veräußert werden. Doch der Münchner Kreistag stoppte die Holzauktion.

Auch andere Initiativen, etwa Spieler als Vermittler beim Verkauf von Eigentumswohnungen einzusetzen und die Provisionen dem TSV 1860 zuzuschreiben, fruchteten wenig. Ebenso mißlang eine Plakettenverkaufsaktion „I like 1860“.

Denn immer weniger liebten die Fans ihre Spieler, die zwar teuer waren, aber auf dem Spielfeld kaum etwas leisteten. Zwölf Trainer versuchten seit

Münchner 1860-Präsident Riedl, Trainer Lucas: „Wenn mein Plan nicht gelingt, müssen Großbanken die Bundesliga managen“

DER SPIEGEL, Nr. 52/1976 113

Figure 5.3: *The creation of a benchmark segmentation using the GUI tool. The image is displayed in grey scale and areas are highlighted by the user according to which segment the area belongs to. White blocks indicate background, green blocks indicate image, blue blocks indicate text, yellow blocks indicate uncertainty between background and image, and purple blocks indicate uncertainty between background and text. This information is then saved to a file along with the Haralick feature segmentations and the original image.*

5.1.8 The terminal set

For the terminals of the genetic system, indexes to different possible Haralick feature segmentations are used. When the genetic program is run, the quick clustering algorithm is used to perform the clusterings as described in algorithm 6. The terminals comprise of all the simple combinations of Haralick features (the eight Haralick features discussed in 3.3.4), angles $(0^o, 45^o, 90^o, 135^o)$ and distances (1 or 2). Our terminals are triple: *(feature name, angle, distance)*. For example: *(Contrast, 45, 1)*, *(Energy, 90, 2)*, *(Mean, 0, 1)*.

In terms of the fitness function evaluation, the terminals represent the resulting values of a segmentation of the test image via a single Haralick feature (for example this could be contrast at angle 45 degrees at distance 1). The observed values of each segmentation after the K-Means clustering are 0, 1 or 2 for the image, text and background segments. The K-Means algorithm is only used for training input, not in the application of the genetic program.

In order to allow our system more flexibility, the use of faster computations, and input data that is more relevant to decision making[71, 83, 84]; binary observed values are used (instead of observed values of 0,1 and 2). To do this, each resulting segmentation is split into two segmentations with possible values of 0 or 1 (the process of splitting is further described in section 5.1.9).

As far as the representation of the terminals is concerned, each of the two binary segmentations maps of the initial segmentation from the Haralick/K-Means algorithm [eg. *(IDM, 45, 1)*] are known as 'splits' (with segmentations being split for training, as described in the previous paragraph). In the case of the terminals, the split is represented by the split parameter. The split parameter may have a value of 0 or 1 to denote the two different splits of the observed values [eg. *(IDM, 45, 1,0)* and *(IDM, 45, 1,1)*]. The final representation of the terminal values is as follows: (*feature name, angle, distance, split parameter*).

As far as the semantics of the terminals are concerned, when running a genetic program that has undergone training via the genetic system on other document images, the program would interpret the terminal value *(Contrast, 45, 1, 0)* as the quick clustering segmentation of the new document image using the Haralick feature contrast at 45^o and distance one. It would then, since the split parameter is 0, interpret all the values of 1 as 1 and all other values (0 or 2) as 0.

5.1.9 The splitting of training and input data

The input training data is split into two binary arrays (for use by the binary functions) as illustrated in figure 5.4. This new representation encodes the mapping of the image, text and background segments into a binary format. This is done as follows:

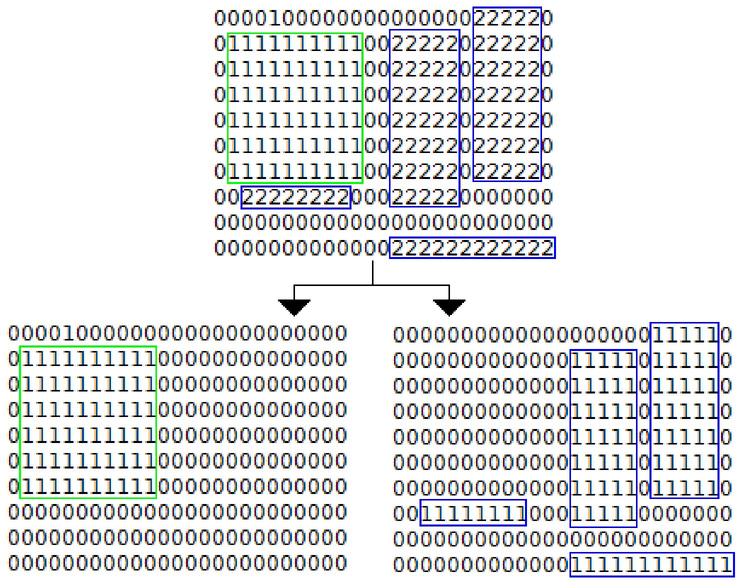
$$f : \{0,1,2\}^{N\times M} \longrightarrow \{0,1\}^{N\times M} \times \{0,1\}^{N\times M}$$

$$I \longmapsto (I_1, I_2)$$

$$f(I)(x,y) = (0,0) \text{ if } I(x,y) = 0$$

$$f(I)(x,y) = (1,0) \text{ if } I(x,y) = 1$$

$$f(I)(x,y) = (0,1) \text{ if } I(x,y) = 2$$



```
0000100000000000000222220
0111111111100222220222220
0111111111100222220222220
0111111111100222220222220
0111111111100222220222220
0111111111100222220222220
0111111111100222220222220
0022222222000222220000000
0000000000000000000000000
0000000000000222222222222
```

```
0000100000000000000000000    0000000000000000000111110
0111111111100000000000000    0000000000000111110111110
0111111111100000000000000    0000000000000111110111110
0111111111100000000000000    0000000000000111110111110
0111111111100000000000000    0000000000000111110111110
0111111111100000000000000    0000000000000111110111110
0111111111100000000000000    0000000000000111110111110
0000000000000000000000000    0011111111000111110000000
0000000000000000000000000    0000000000000000000000000
0000000000000000000000000    0000000000000111111111111
```

Figure 5.4: *Translation of segmentation data from a $\{0,1,2\}^{N\times M}$ segmentation to two $\{0,1\}^{N\times M}$ segmentations. Fuzzy training values of 3 (image or background) and 4 (background or text) are not taken into account when genetic programs are evaluated, and only apply when a genetic program is being trained.*

5.1.10 The function set

Boolean operations have been chosen as the function set as they lead to fast computations. This is a valuable property in the context of genetic systems due to the very large number of computations that are carried out when determining the fitness of individuals. Boolean operations are also a logical choice due to the decision making nature of the problem (since a true or false result is required when determining whether or not a block belongs to a segment).

The functions act upon between 1 and 3 terminals as described below and as graphically illustrated in figure 5.5

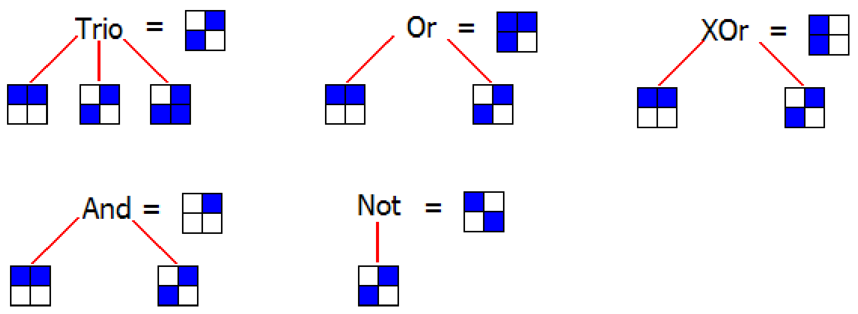


Figure 5.5: *Graphical representation of the boolean function set.*

AND

AND acts upon two input values, producing an output value equal to the binary AND of the two input values. If both values are equal to 1, then the output has a value of 1, otherwise the output has a value of 0. This function has the effect of combining two possibly over-sensitive (in terms of giving false positives) terminals together into a more useful result.

OR

OR acts upon two input values, producing an output value equal to the binary OR of the two input values. If both of the input values are equal to 0, then the output has a value of

0, otherwise the output has a value of 1. This function has the effect of combining two possibly under-sensitive (in terms of giving false negatives) terminals together into a more useful result.

XOR

XOR acts upon two input values, producing an output value equal to the binary XOR of the two input values. If exactly one of the input values are equal to 1, then the output has a value of 1, otherwise the output has a value of 0.

NOT

NOT acts upon one input value, producing an output value equal to the binary NOT of the input value. If the input value is equal to 1, then the output has a value of 0, otherwise the output has a value of 1. This operation has been chosen for inverting input values to a possibly correct value.

TRIO

TRIO acts upon three input values, producing the output value as the value that is in the majority for the three input values.

The evaluation of individuals

The fitness function will combine the K-Means Haralick segmentations from the input according to the genetic program and determine the results. The results are then compared to the ideal result for the test case and then the distance from the correct result is returned (see example 7).

Example 7 Parse tree evaluation

For example a parse tree of the form:

(Mean, 0, 1, 1) XOR ((Contrast, 45, 1, 0) OR NOT (TRIO ((Energy, 90, 2, 1), (Correlation, 0, 1, 1), (Energy, 135, 1, 0))))

where (Mean, 0, 1, 1), (Contrast, 45, 1, 0), (Energy, 90, 2, 1), (Correlation, 0, 1, 1), (Energy, 135, 1, 0) in the test case have values 1, 0, 1, 1, 0 respectively will evaluate to

1 XOR (0 OR NOT (TRIO (1, 1, 0)))

1 XOR (0 OR NOT (1))

1 XOR (0 OR 0)

1 XOR 0

RESULT = 1

this value is then compared to the fitness case to determine distance

DISTANCE = ABS (DESIRED VALUE - RESULT)

if our DESIRED VALUE = 0

DISTANCE = ABS (0 - 1)

DISTANCE = 1

This distance is then added to the sum of the distance of the individuals results from the correct answer (their fitness).

5.1.11 The representation of individuals

The content of each node is represented as a string in this implementation (for example: AND, OR, NOT, or some number indexing a segmentation). Each node also contains a count of the number of child nodes it has and references to its child nodes. This is illustrated in figure 5.6. See figure 5.7 for an example of some individuals in the population. Representations 8 and 9 describe the composition of individuals and populations in the system respectively.

Representation 8 Individuals in the population are represented as follows:

TREE class:

(1) A pointer to a root node.

(2) Fitness calculation capabilities.

(3) Self copying capabilities.

(4) Self output capabilities.

(5) Capabilities to return a pointer to a random node in the tree.

NODE class:

.1 A string representing the contents of the node.

.2 An integer (N) representing the number of children the node has.

.3 An array of [0..N] pointers to child nodes.

5.1.12 The fitness function

The fitness function uses the desired segmentation specified in the input file for the training image (the desired result for the test cases) and then compares it to the result achieved by the evaluation of each individual with the test cases. The function then returns a fitness, in this implementation the percentage of correct classifications is used when determining fitness.

When determining fitness, the program runs through all sets of input values and calculates the results from the genetic program (see fig. 4.1). The result is then compared to the corresponding desired value for the inputs (see fig. 5.8).

Representation 9 Populations are represented as follows:

POPULATION class:

(1) An integer (N) representing the number of individuals in a population

(2) An array of [0..N] pointers to individuals in the population.

(3) An integer representing the maximum fitness of the population.

(4) A function which performs an evolutionary step for the population.

(5) Functions to perform the genetic operations (mutate, reproduce and crossover)

(6) A function for displaying the entire population

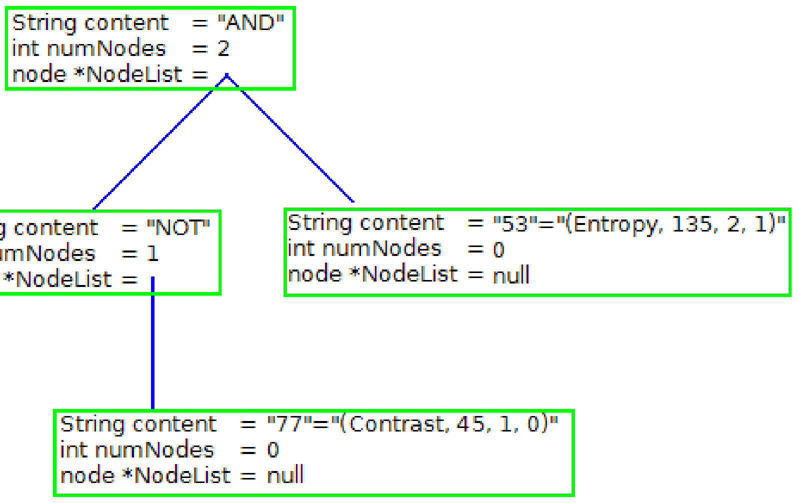
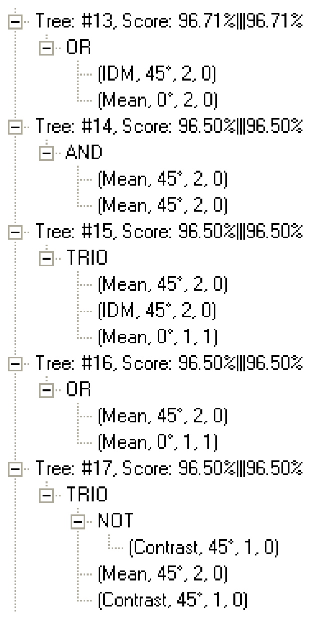


Figure 5.6: *Illustrates the representation of the genetic trees to the program.*

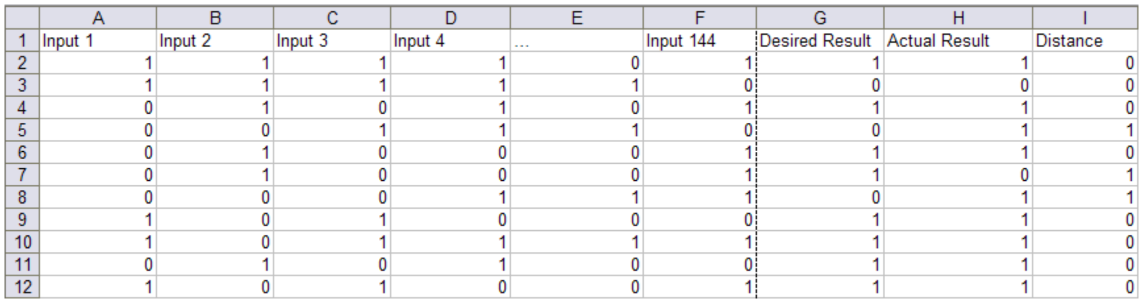


```
⊟ Tree: #13, Score: 96.71%|||96.71%
   ⊟ OR
      (IDM, 45°, 2, 0)
      (Mean, 0°, 2, 0)
⊟ Tree: #14, Score: 96.50%|||96.50%
   ⊟ AND
      (Mean, 45°, 2, 0)
      (Mean, 45°, 2, 0)
⊟ Tree: #15, Score: 96.50%|||96.50%
   ⊟ TRIO
      (Mean, 45°, 2, 0)
      (IDM, 45°, 2, 0)
      (Mean, 0°, 1, 1)
⊟ Tree: #16, Score: 96.50%|||96.50%
   ⊟ OR
      (Mean, 45°, 2, 0)
      (Mean, 0°, 1, 1)
⊟ Tree: #17, Score: 96.50%|||96.50%
   ⊟ TRIO
      ⊟ NOT
         (Contrast, 45°, 1, 0)
      (Mean, 45°, 2, 0)
      (Contrast, 45°, 1, 0)
```

Figure 5.7: *Final output of some of the genetic programs created.*

Formally, the fitness of individuals is calculated as follows (where F_i is the fitness of individual i, N_c is the number of fitness cases, R_a is the achieved result and R_d is the desired result):

$$F_i = 1 - \frac{\sum_{j=1}^{N_c} |R_a - R_d|}{N_c} \tag{5.1}$$



	A	B	C	D	E	F	G	H	I
1	Input 1	Input 2	Input 3	Input 4	...	Input 144	Desired Result	Actual Result	Distance
2	1	1	1	1	0	1	1	1	0
3	1	1	1	1	1	0	0	0	0
4	0	1	0	1	0	1	1	1	0
5	0	0	1	1	1	0	0	1	1
6	0	1	0	0	0	1	1	1	0
7	0	1	0	0	0	1	1	0	1
8	0	0	0	1	1	1	0	1	1
9	1	0	1	0	0	0	1	1	0
10	1	0	1	1	1	1	1	1	0
11	0	1	0	1	0	0	1	1	0
12	1	0	1	0	0	1	1	1	0

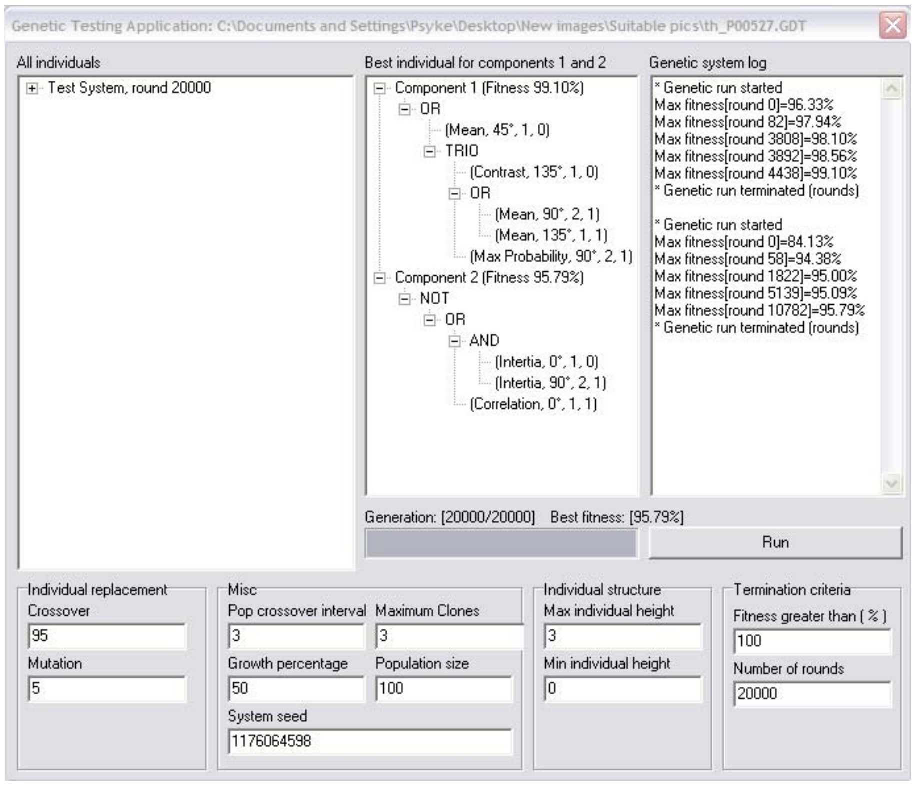
Figure 5.8: *This is an example of fitness cases for this application. In this table the values going down the columns are for the observed values (each square segmented in the image moving across the page from left to right then to the following row). The 3rd and 2nd to last columns are the output and desired values for that square of the segmentation. Input 1 to 144 are observed values of the segmentation by (featurename,angle,distance,split).*

5.1.13 Finishing off

Once the genetic system has found a suitable individual for segmenting the image and a suitable individual for segmenting the text, the genetic programs will be saved to a file for recall for segmenting other document images (as described in algorithm 6).

5.1.14 Example training and segmentation

The following figures (5.9,5.10,5.11 and 5.12) show the training of genetic program G527 and some example segmentations. The input data for the training is shown in figure 5.3. The original image 527 used for training can be found in appendix C.



Genetic Testing Application: C:\Documents and Settings\Psyke\Desktop\New images\Suitable pics\th_P00527.GDT

All individuals

```
⊞ Test System, round 20000
```

Best individual for components 1 and 2

```
⊟ Component 1 (Fitness 99.10%)
    ⊟ OR
        (Mean, 45°, 1, 0)
        ⊟ TRIO
            (Contrast, 135°, 1, 0)
            ⊟ OR
                (Mean, 90°, 2, 1)
                (Mean, 135°, 1, 1)
            (Max Probability, 90°, 2, 1)
⊟ Component 2 (Fitness 95.79%)
    ⊟ NOT
        ⊟ OR
            ⊟ AND
                (Intertia, 0°, 1, 0)
                (Intertia, 90°, 2, 1)
            (Correlation, 0°, 1, 1)
```

Genetic system log

```
* Genetic run started
Max fitness[round 0]=96.33%
Max fitness[round 82]=97.94%
Max fitness[round 3808]=98.10%
Max fitness[round 3892]=98.56%
Max fitness[round 4438]=99.10%
* Genetic run terminated (rounds)

* Genetic run started
Max fitness[round 0]=84.13%
Max fitness[round 58]=94.38%
Max fitness[round 1822]=95.00%
Max fitness[round 5139]=95.09%
Max fitness[round 10782]=95.79%
* Genetic run terminated (rounds)
```

Generation: [20000/20000] Best fitness: [95.79%]

Run

Individual replacement
- Crossover: 95
- Mutation: 5

Misc
- Pop crossover interval: 3
- Maximum Clones: 3
- Growth percentage: 50
- Population size: 100
- System seed: 1176064598

Individual structure
- Max individual height: 3
- Min individual height: 0

Termination criteria
- Fitness greater than (%): 100
- Number of rounds: 20000

Figure 5.9: *This image shows the genetic parameters for the creation of the genetic program used in the following images. The genetic program itself (which shall be referred to as G527) is shown in the centre output and the generations on which the best genetic program was improved over are shown in the right output.*

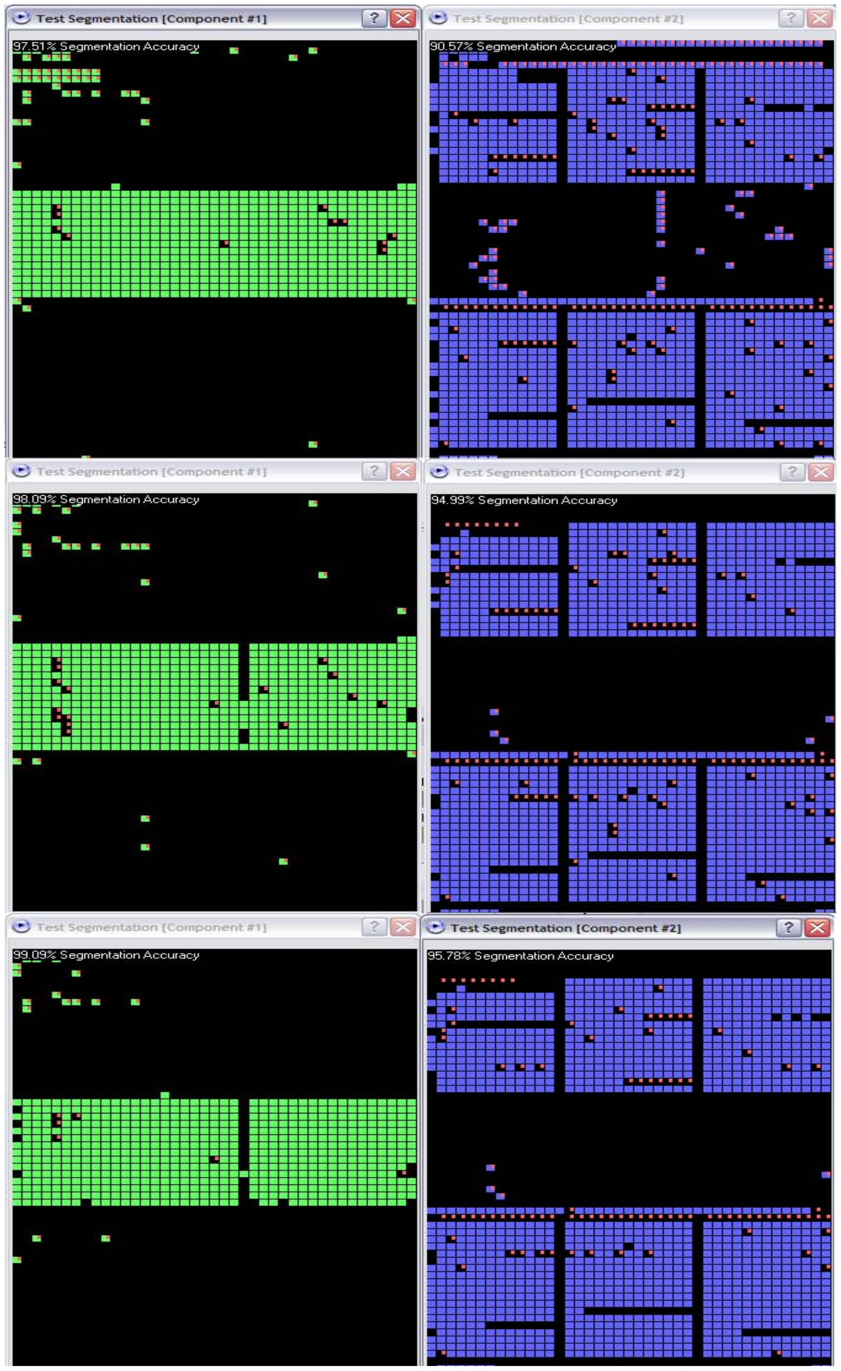

Test Segmentation [Component #1]
97.51% Segmentation Accuracy
Test Segmentation [Component #2]
90.57% Segmentation Accuracy
Test Segmentation [Component #1]
98.09% Segmentation Accuracy
Test Segmentation [Component #2]
94.99% Segmentation Accuracy
Test Segmentation [Component #1]
99.09% Segmentation Accuracy
Test Segmentation [Component #2]
95.78% Segmentation Accuracy

Figure 5.10: *These images demonstrate the training process as it progresses. The top pair of images are the image and text components segmented by the best genetic program at generation 1. The middle pair of images are the image and text components segmented by the best genetic program at generation 3808. The bottom pair of images are the image and text components segmented by the best genetic program at generation 10785. Red blocks indicate incorrect classification, whereas blue regions indicate the region being classified as part of the text component and green regions indicate the region being classified as part of the image component. Black regions do not belong to the component being extracted.*

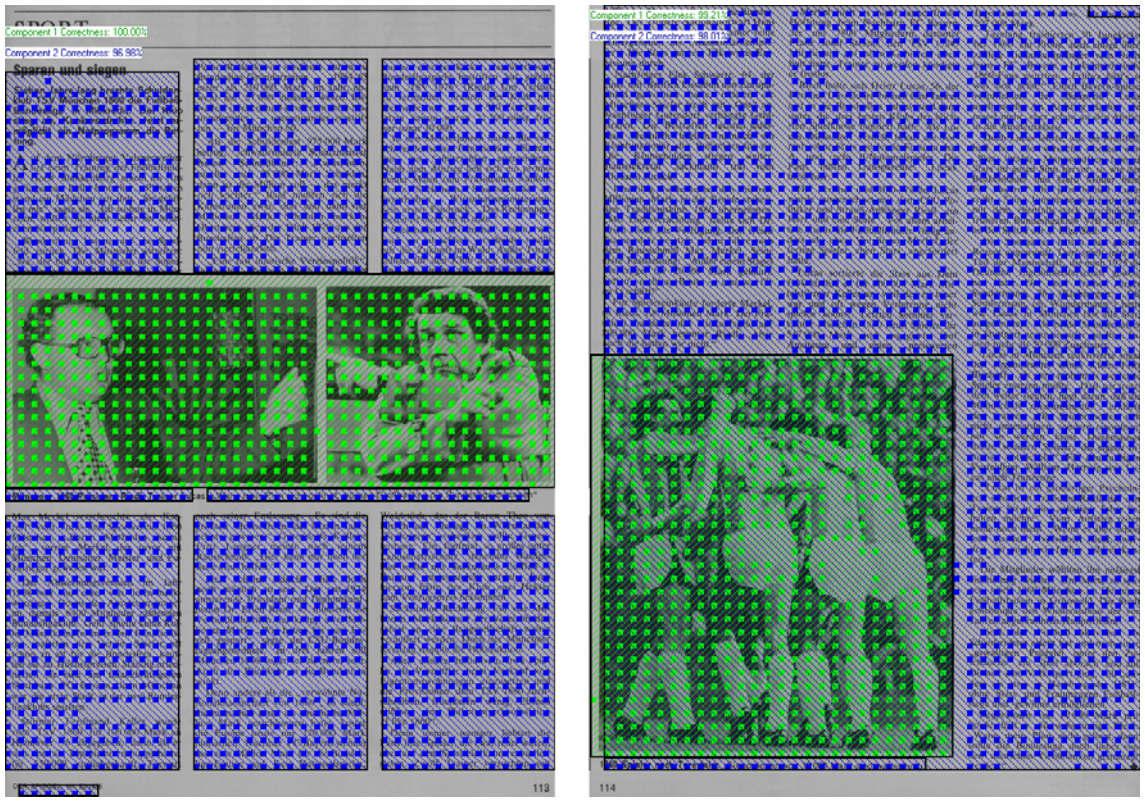

Component 1 Correctness: 100.00%
Component 2 Correctness: 96.98%
SPORT
Sparen und siegen
113
Component 1 Correctness: 99.21%
Component 2 Correctness: 98.01%
114

Figure 5.11: *The left image is image 527 segmented by the genetic programming based system (program G527). Image 527 is the image that was used to train the genetic program being used. The right image is image 528 segmented by the genetic programming based system (program G527). A full sized version of this image can be found in appendix E. The original unsegmented images can be found in C.*

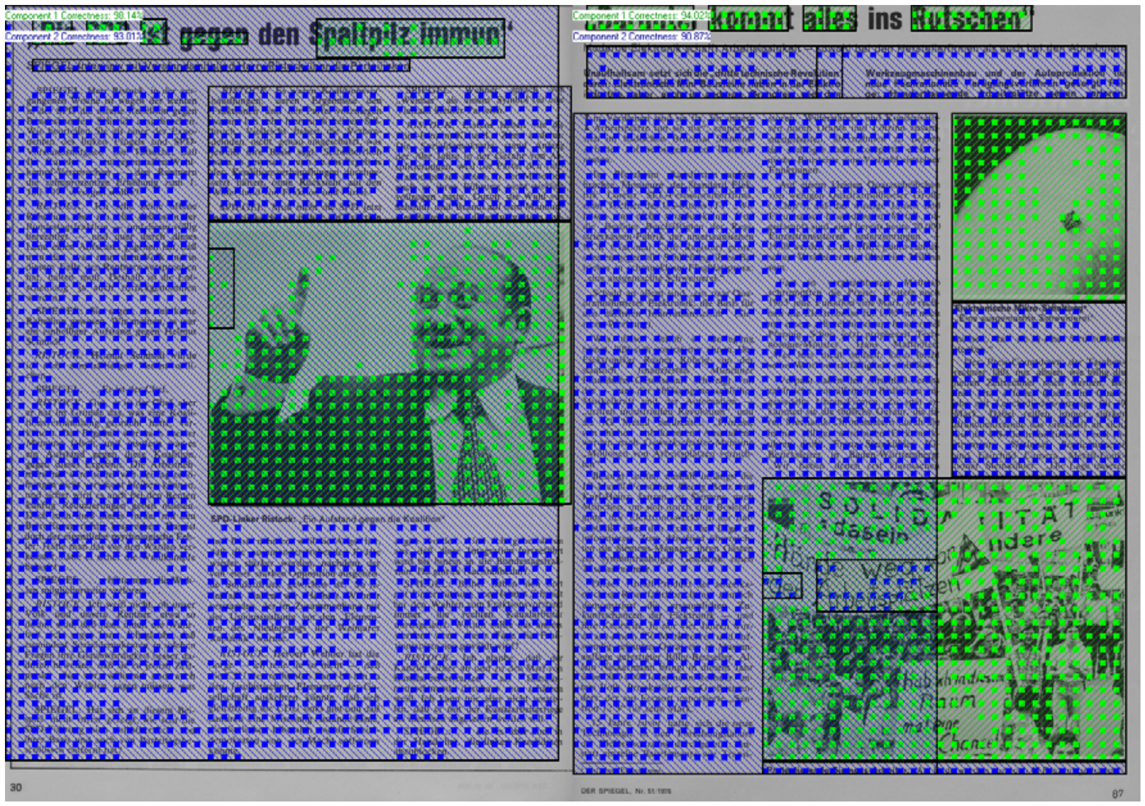


Figure 5.12: *The left image is image 531 segmented by the genetic programming based system (program G527). The right image is image 533 segmented by the genetic programming based system (program G527). A full sized version of this image can be found in appendix E. The original unsegmented images can be found in C.*

5.2 Comparison system

In order to test the proposed genetic programming based system, it was necessary to implement a system to test it against that uses a standard approach to document image segmentation. The system performs segmentations by making use of the K-Means algorithm to cluster the Haralick feature space of a document image. The results of the K-Means segmentation are then post-processed (as described in section 5.3). The post-processing used on the GLCM/K-Means based algorithms' results is the same as the post-processing used upon the genetic programming based systems' results. Example segmented images can be found in appendix D.

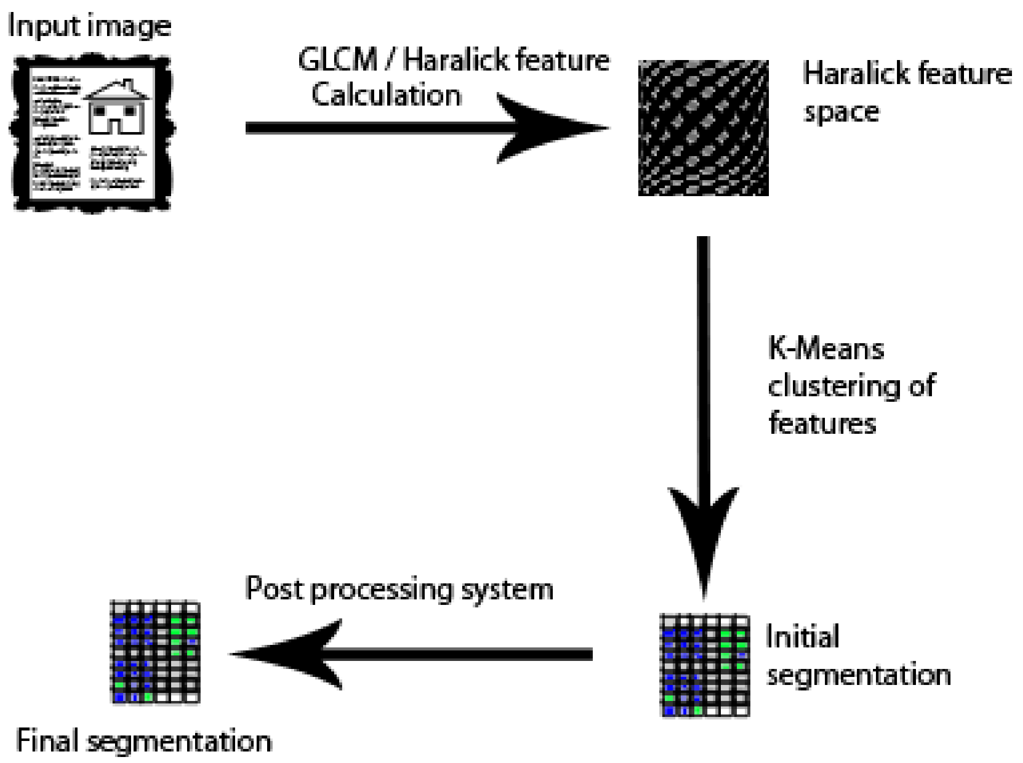


Figure 5.13: *The process by which document images are segmented by the system used for comparison.*

The following Haralick features (at angles 0^o, 45^o, 90^o, 135^o and distances of 1 and 2) are clustered when performing the segmentation:

- Contrast
- Energy
- Entropy
- Maximum probability
- Inverse difference moment
- Correlation
- Mean
- Standard deviation

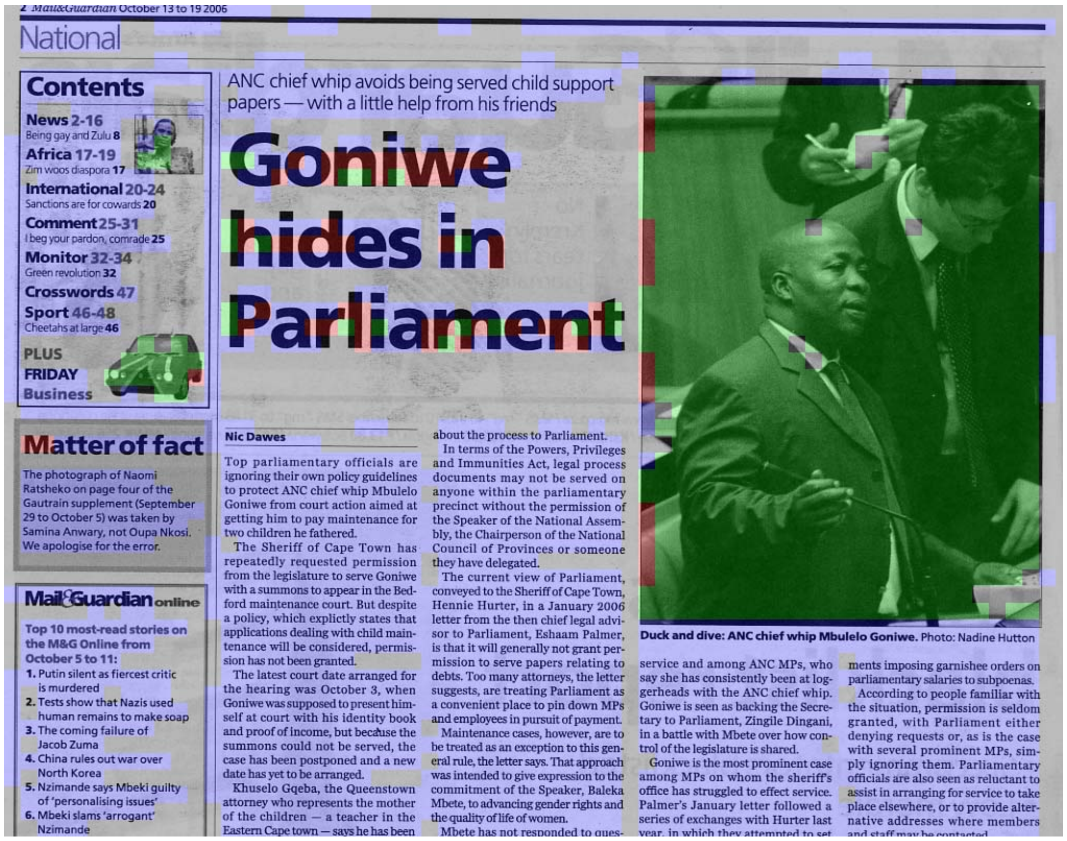
5.3 Post-processing

Texture classifiers themselves generally do not make use of surrounding regions for decision making, they simply classify the texture being examined by itself. The task then falls upon post-processing to apply domain specific knowledge and perform clean-up of the segmentation. It is, however very important to note that the input information to the post-processing needs to be fairly accurate, otherwise the post-processing is worthless. Post-processing is an important step for document image segmentation systems since there is a large amount of domain specific knowledge that can be applied. For example:

(1) Text regions generally take on a rectangular shape and need to be more than one or two units wide (with 16x16 texture windows).

(2) Image regions generally take on a rectangular shape and need to be more than one or two units wide and high (with 16x16 texture windows).

(3) Text and image generally do not appear in very small amounts (noise), or surrounded by other non-background components (text or high contrast regions in images, or image surrounded by text as a result of noise).



2 Mail&Guardian October 13 to 19 2006

National

Contents

News 2-16
Being gay and Zulu **8**
Africa 17-19
Zim woos diaspora **17**
International 20-24
Sanctions are for cowards **20**
Comment 25-31
I beg your pardon, comrade **25**
Monitor 32-34
Green revolution **32**
Crosswords 47
Sport 46-48
Cheetahs at large **46**

PLUS
FRIDAY
Business

Matter of fact

The photograph of Naomi Ratsheko on page four of the Gautrain supplement (September 29 to October 5) was taken by Samina Anwary, not Oupa Nkosi. We apologise for the error.

Mail&Guardian online

Top 10 most-read stories on the M&G Online from October 5 to 11:

1. Putin silent as fiercest critic is murdered
2. Tests show that Nazis used human remains to make soap
3. The coming failure of Jacob Zuma
4. China rules out war over North Korea
5. Nzimande says Mbeki guilty of 'personalising issues'
6. Mbeki slams 'arrogant' Nzimande

ANC chief whip avoids being served child support papers — with a little help from his friends

Goniwe hides in Parliament

Nic Dawes

Top parliamentary officials are ignoring their own policy guidelines to protect ANC chief whip Mbulelo Goniwe from court action aimed at getting him to pay maintenance for two children he fathered.

The Sheriff of Cape Town has repeatedly requested permission from the legislature to serve Goniwe with a summons to appear in the Bedford maintenance court. But despite a policy, which explictly states that applications dealing with child maintenance will be considered, permission has not been granted.

The latest court date arranged for the hearing was October 3, when Goniwe was supposed to present himself at court with his identity book and proof of income, but because the summons could not be served, the case has been postponed and a new date has yet to be arranged.

Khuselo Gqeba, the Queenstown attorney who represents the mother of the children — a teacher in the Eastern Cape town — says he has been about the process to Parliament.

In terms of the Powers, Privileges and Immunities Act, legal process documents may not be served on anyone within the parliamentary precinct without the permission of the Speaker of the National Assembly, the Chairperson of the National Council of Provinces or someone they have delegated.

The current view of Parliament, conveyed to the Sheriff of Cape Town, Hennie Hurter, in a January 2006 letter from the then chief legal advisor to Parliament, Eshaam Palmer, is that it will generally not grant permission to serve papers relating to debts. Too many attorneys, the letter suggests, are treating Parliament as a convenient place to pin down MPs and employees in pursuit of payment.

Maintenance cases, however, are to be treated as an exception to this general rule, the letter says. That approach was intended to give expression to the commitment of the Speaker, Baleka Mbete, to advancing gender rights and the quality of life of women.

Mbete has not responded to ques-

Duck and dive: ANC chief whip Mbulelo Goniwe. Photo: Nadine Hutton

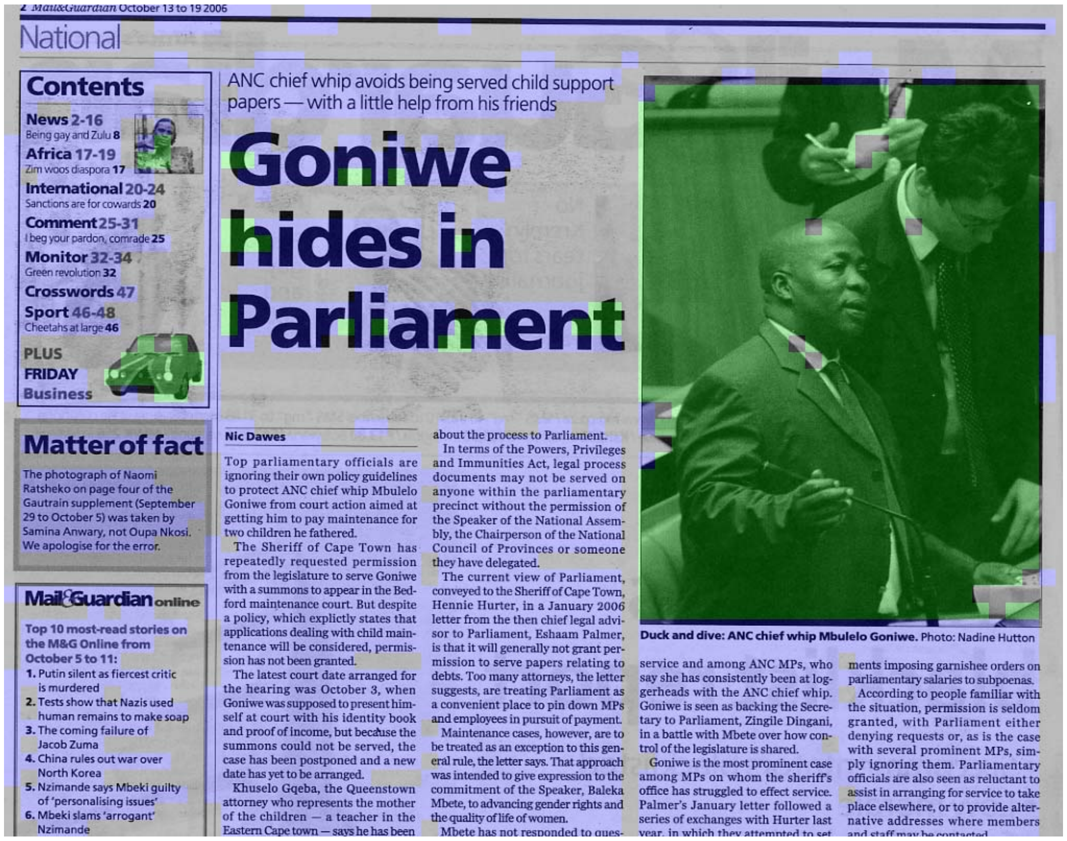
service and among ANC MPs, who say she has consistently been at loggerheads with the ANC chief whip. Goniwe is seen as backing the Secretary to Parliament, Zingile Dingani, in a battle with Mbete over how control of the legislature is shared.

Goniwe is the most prominent case among MPs on whom the sheriff's office has struggled to effect service. Palmer's January letter followed a series of exchanges with Hurter last year, in which they attempted to set

ments imposing garnishee orders on parliamentary salaries to subpoenas.

According to people familiar with the situation, permission is seldom granted, with Parliament either denying requests or, as is the case with several prominent MPs, simply ignoring them. Parliamentary officials are also seen as reluctant to assist in arranging for service to take place elsewhere, or to provide alternative addresses where members and staff may be contacted.

Figure 5.14: *This image is the initial segmentation of fig. C.1 by the genetic programming based system. No post-processing has been performed. Red regions indicate indecision between the text and image components, blue indicates that the region belongs to the text component and green indicates that the region belongs to the image component.*



2 *Mail&Guardian* October 13 to 19 2006

National

Contents

News 2-16
Being gay and Zulu **8**
Africa 17-19
Zim woos diaspora **17**
International 20-24
Sanctions are for cowards **20**
Comment 25-31
I beg your pardon, comrade **25**
Monitor 32-34
Green revolution **32**
Crosswords 47
Sport 46-48
Cheetahs at large **46**
PLUS
FRIDAY
Business

Matter of fact

The photograph of Naomi Ratsheko on page four of the Gautrain supplement (September 29 to October 5) was taken by Samina Anwary, not Oupa Nkosi. We apologise for the error.

Mail&Guardian online

Top 10 most-read stories on the M&G Online from October 5 to 11:

1. Putin silent as fiercest critic is murdered
2. Tests show that Nazis used human remains to make soap
3. The coming failure of Jacob Zuma
4. China rules out war over North Korea
5. Nzimande says Mbeki guilty of 'personalising issues'
6. Mbeki slams 'arrogant' Nzimande

ANC chief whip avoids being served child support papers — with a little help from his friends

Goniwe hides in Parliament

Nic Dawes

Top parliamentary officials are ignoring their own policy guidelines to protect ANC chief whip Mbulelo Goniwe from court action aimed at getting him to pay maintenance for two children he fathered.

The Sheriff of Cape Town has repeatedly requested permission from the legislature to serve Goniwe with a summons to appear in the Bedford maintenance court. But despite a policy, which explictly states that applications dealing with child maintenance will be considered, permission has not been granted.

The latest court date arranged for the hearing was October 3, when Goniwe was supposed to present himself at court with his identity book and proof of income, but because the summons could not be served, the case has been postponed and a new date has yet to be arranged.

Khuselo Gqeba, the Queenstown attorney who represents the mother of the children — a teacher in the Eastern Cape town — says he has been about the process to Parliament.

In terms of the Powers, Privileges and Immunities Act, legal process documents may not be served on anyone within the parliamentary precinct without the permission of the Speaker of the National Assembly, the Chairperson of the National Council of Provinces or someone they have delegated.

The current view of Parliament, conveyed to the Sheriff of Cape Town, Hennie Hurter, in a January 2006 letter from the then chief legal advisor to Parliament, Eshaam Palmer, is that it will generally not grant permission to serve papers relating to debts. Too many attorneys, the letter suggests, are treating Parliament as a convenient place to pin down MPs and employees in pursuit of payment.

Maintenance cases, however, are to be treated as an exception to this general rule, the letter says. That approach was intended to give expression to the commitment of the Speaker, Baleka Mbete, to advancing gender rights and the quality of life of women.

Mbete has not responded to ques-

Duck and dive: ANC chief whip Mbulelo Goniwe. Photo: Nadine Hutton

service and among ANC MPs, who say she has consistently been at loggerheads with the ANC chief whip. Goniwe is seen as backing the Secretary to Parliament, Zingile Dingani, in a battle with Mbete over how control of the legislature is shared.

Goniwe is the most prominent case among MPs on whom the sheriff's office has struggled to effect service. Palmer's January letter followed a series of exchanges with Hurter last year, in which they attempted to set

ments imposing garnishee orders on parliamentary salaries to subpoenas.

According to people familiar with the situation, permission is seldom granted, with Parliament either denying requests or, as is the case with several prominent MPs, simply ignoring them. Parliamentary officials are also seen as reluctant to assist in arranging for service to take place elsewhere, or to provide alternative addresses where members and staff may be contacted.

Figure 5.15: *This image is the segmentation of fig. C.1 by the genetic programming based system. Pass 1 has been performed. Blue indicates that the region belongs to the text component and green indicates that the region belongs to the image component.*

5.3.1 Pass 1: fuzzy class removal

This process removes the fuzzy class from the output by replacing regions classified as being uncertain with their most likely class. This is done by connected component analysis of the fuzzy class and then making the class decision based on the neighbourhood of that component. See fig. 5.15 for the resulting segmentation and fig. 5.14 for the image in its raw segmented form. Algorithm 10 describes the steps taken for the first post-processing pass.

Algorithm 10 Pass 1:

```
(1) Perform connected component analysis on all fuzzy regions.
(2) Label each fuzzy region.
(3) For each connected fuzzy region.
    (.1) Determine number of squares surrounding the fuzzy region that belong to the
         image and text components.
    (.2) Set the entire fuzzy region to a text or image region, depending on what sur-
         rounding component is in the majority.
```

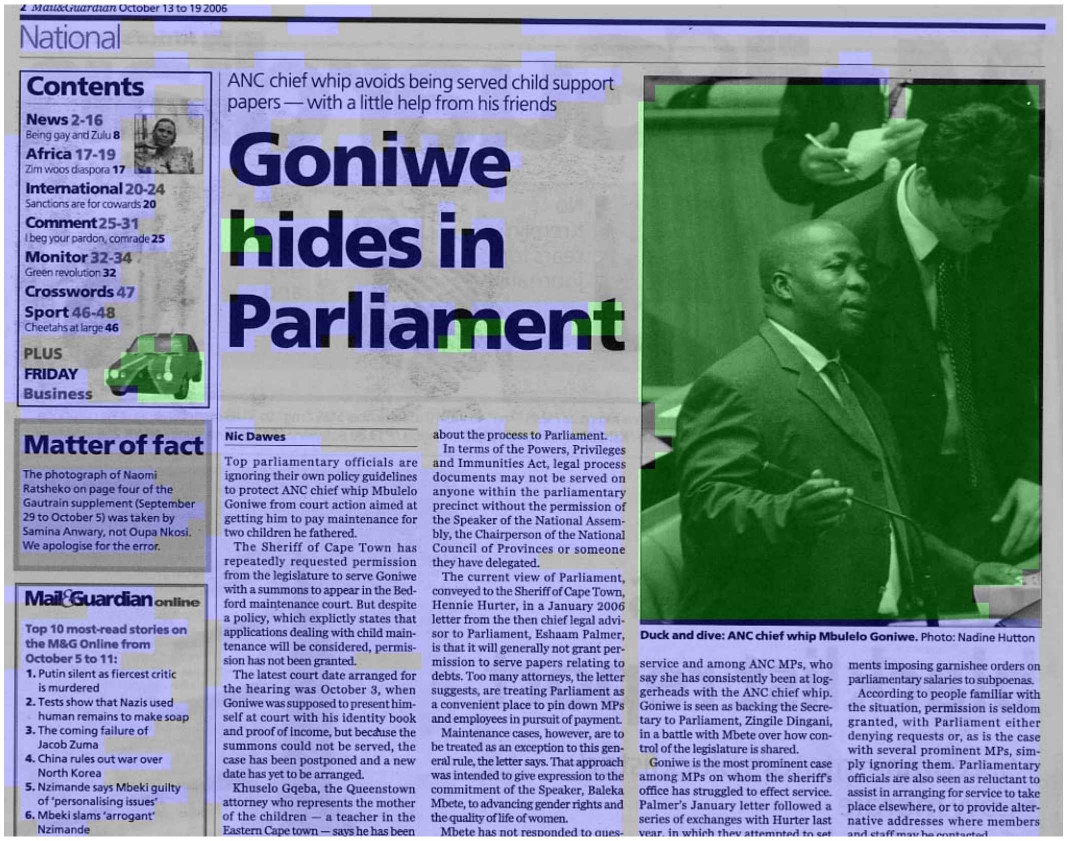
5.3.2 Pass 2: text component clean-up

This process removes weak connections (often caused by noise) between text components. This is done to separate columns of text. The method for this step entails using vertical line scans to find weak connections within a connected component. This process is only applied to the text class. Also, if a text connected component is one column wide, it can be considered to be noise and is replaced by the component which is in the majority of its neighbourhood. This is because text components do not appear in vertical lines, and because border lines in document images are often confused with text components since they share certain characteristics. In the case of the example image's segmentation there were no weak connections. Algorithm 11 describes the steps taken for the second post-processing pass.

Algorithm 11 Pass 2:

```
(1) Perform connected component analysis on all text regions.
(2) Label each connected text region.
(3) For each connected text region.
    (.1) Perform line scans from the left to the right of the connected text region, summing the number of squares that belong to the connected text region along each column.
    (.2) If the column with the smallest number of squares has less than 50% of the number of squares that the column with the second smallest number of squares has, then the smallest column can be considered to be noise. In the case of a noise column being found, the component is split into a left region and a right region.
    (.3) If the text connected component is one column wide it is replaced with the component that is in the majority in its immediate neighbourhood.
```

5.3.3 Pass 3: noise culler



2 Mail&Guardian October 13 to 19 2006

National

Contents

News 2-16
Being gay and Zulu 8
Africa 17-19
Zim woos diaspora 17
International 20-24
Sanctions are for cowards 20
Comment 25-31
I beg your pardon, comrade 25
Monitor 32-34
Green revolution 32
Crosswords 47
Sport 46-48
Cheetahs at large 46
PLUS
FRIDAY
Business

Matter of fact

The photograph of Naomi Ratsheko on page four of the Gautrain supplement (September 29 to October 5) was taken by Samina Anwary, not Oupa Nkosi. We apologise for the error.

Mail&Guardian online

Top 10 most-read stories on the M&G Online from October 5 to 11:

1. Putin silent as fiercest critic is murdered
2. Tests show that Nazis used human remains to make soap
3. The coming failure of Jacob Zuma
4. China rules out war over North Korea
5. Nzimande says Mbeki guilty of 'personalising issues'
6. Mbeki slams 'arrogant' Nzimande

ANC chief whip avoids being served child support papers — with a little help from his friends

Goniwe hides in Parliament

Nic Dawes

Top parliamentary officials are ignoring their own policy guidelines to protect ANC chief whip Mbulelo Goniwe from court action aimed at getting him to pay maintenance for two children he fathered.

The Sheriff of Cape Town has repeatedly requested permission from the legislature to serve Goniwe with a summons to appear in the Bedford maintenance court. But despite a policy, which explictly states that applications dealing with child maintenance will be considered, permission has not been granted.

The latest court date arranged for the hearing was October 3, when Goniwe was supposed to present himself at court with his identity book and proof of income, but because the summons could not be served, the case has been postponed and a new date has yet to be arranged.

Khuselo Gqeba, the Queenstown attorney who represents the mother of the children — a teacher in the Eastern Cape town — says he has been about the process to Parliament.

In terms of the Powers, Privileges and Immunities Act, legal process documents may not be served on anyone within the parliamentary precinct without the permission of the Speaker of the National Assembly, the Chairperson of the National Council of Provinces or someone they have delegated.

The current view of Parliament, conveyed to the Sheriff of Cape Town, Hennie Hurter, in a January 2006 letter from the then chief legal advisor to Parliament, Eshaam Palmer, is that it will generally not grant permission to serve papers relating to debts. Too many attorneys, the letter suggests, are treating Parliament as a convenient place to pin down MPs and employees in pursuit of payment.

Maintenance cases, however, are to be treated as an exception to this general rule, the letter says. That approach was intended to give expression to the commitment of the Speaker, Baleka Mbete, to advancing gender rights and the quality of life of women.

Mbete has not responded to ques-

Duck and dive: ANC chief whip Mbulelo Goniwe. Photo: Nadine Hutton

service and among ANC MPs, who say she has consistently been at loggerheads with the ANC chief whip. Goniwe is seen as backing the Secretary to Parliament, Zingile Dingani, in a battle with Mbete over how control of the legislature is shared.

Goniwe is the most prominent case among MPs on whom the sheriff's office has struggled to effect service. Palmer's January letter followed a series of exchanges with Hurter last year, in which they attempted to set

ments imposing garnishee orders on parliamentary salaries to subpoenas.

According to people familiar with the situation, permission is seldom granted, with Parliament either denying requests or, as is the case with several prominent MPs, simply ignoring them. Parliamentary officials are also seen as reluctant to assist in arranging for service to take place elsewhere, or to provide alternative addresses where members and staff may be contacted.

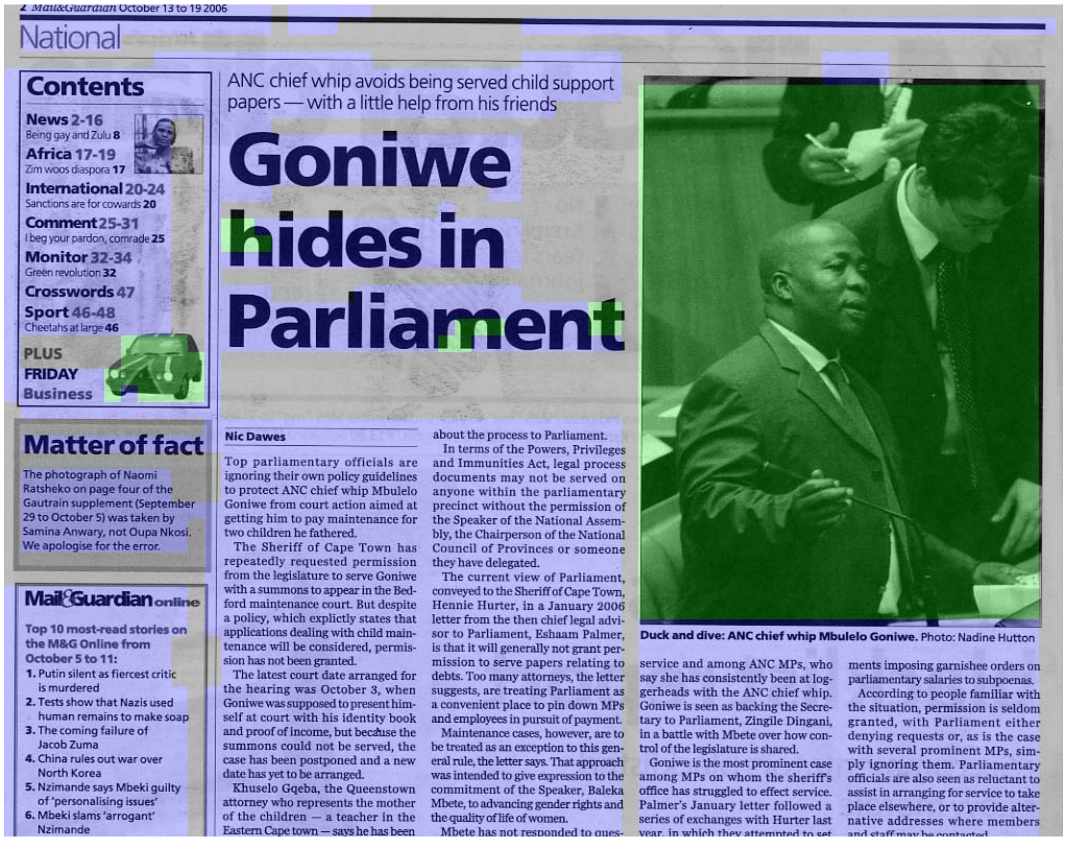
Figure 5.16: *This image is the initial segmentation of fig. C.1 by the genetic programming based system. Passes 1, 2 and 3 have been performed. Blue indicates that the region belongs to the text component and green indicates that the region belongs to the image component.*

This process culls noise components. This is done by finding all connected components and then replacing the components of a size smaller than a certain threshold with the class they most likely are. This is determined by the amount of each class in the immediate neighbourhood of the connected component. See fig. 5.16 for the resulting segmentation. See algorithm 12 for the steps taken during the third post-processing pass.

Algorithm 12 Pass 3:

```
(1) Perform connected component analysis on all regions.
(2) Label each connected region.
(3) For each connected region, if the region is smaller than a certain threshold:
    (.1) Determine number of squares surrounding the region that belong to each component.
    (.2) Set the connected region to a text, background or image region, depending on what surrounding component is in the majority.
```

5.3.4 Pass 4: block assignment


2 Mail&Guardian October 13 to 19 2006

National

Contents

News 2-16
Being gay and Zulu **8**
Africa 17-19
Zim woos diaspora **17**
International 20-24
Sanctions are for cowards **20**
Comment 25-31
I beg your pardon, comrade **25**
Monitor 32-34
Green revolution **32**
Crosswords 47
Sport 46-48
Cheetahs at large **46**

PLUS
FRIDAY
Business

Matter of fact

The photograph of Naomi Ratsheko on page four of the Gautrain supplement (September 29 to October 5) was taken by Samina Anwary, not Oupa Nkosi. We apologise for the error.

Mail&Guardian online

Top 10 most-read stories on the M&G Online from October 5 to 11:

1. Putin silent as fiercest critic is murdered
2. Tests show that Nazis used human remains to make soap
3. The coming failure of Jacob Zuma
4. China rules out war over North Korea
5. Nzimande says Mbeki guilty of 'personalising issues'
6. Mbeki slams 'arrogant' Nzimande

ANC chief whip avoids being served child support papers — with a little help from his friends

Goniwe hides in Parliament

Nic Dawes

Top parliamentary officials are ignoring their own policy guidelines to protect ANC chief whip Mbulelo Goniwe from court action aimed at getting him to pay maintenance for two children he fathered.

The Sheriff of Cape Town has repeatedly requested permission from the legislature to serve Goniwe with a summons to appear in the Bedford maintenance court. But despite a policy, which explictly states that applications dealing with child maintenance will be considered, permission has not been granted.

The latest court date arranged for the hearing was October 3, when Goniwe was supposed to present himself at court with his identity book and proof of income, but because the summons could not be served, the case has been postponed and a new date has yet to be arranged.

Khuselo Gqeba, the Queenstown attorney who represents the mother of the children — a teacher in the Eastern Cape town — says he has been about the process to Parliament.

In terms of the Powers, Privileges and Immunities Act, legal process documents may not be served on anyone within the parliamentary precinct without the permission of the Speaker of the National Assembly, the Chairperson of the National Council of Provinces or someone they have delegated.

The current view of Parliament, conveyed to the Sheriff of Cape Town, Hennie Hurter, in a January 2006 letter from the then chief legal advisor to Parliament, Eshaam Palmer, is that it will generally not grant permission to serve papers relating to debts. Too many attorneys, the letter suggests, are treating Parliament as a convenient place to pin down MPs and employees in pursuit of payment.

Maintenance cases, however, are to be treated as an exception to this general rule, the letter says. That approach was intended to give expression to the commitment of the Speaker, Baleka Mbete, to advancing gender rights and the quality of life of women.

Mbete has not responded to ques-

Duck and dive: ANC chief whip Mbulelo Goniwe. Photo: Nadine Hutton

service and among ANC MPs, who say she has consistently been at loggerheads with the ANC chief whip. Goniwe is seen as backing the Secretary to Parliament, Zingile Dingani, in a battle with Mbete over how control of the legislature is shared.

Goniwe is the most prominent case among MPs on whom the sheriff's office has struggled to effect service. Palmer's January letter followed a series of exchanges with Hurter last year, in which they attempted to set

ments imposing garnishee orders on parliamentary salaries to subpoenas.

According to people familiar with the situation, permission is seldom granted, with Parliament either denying requests or, as is the case with several prominent MPs, simply ignoring them. Parliamentary officials are also seen as reluctant to assist in arranging for service to take place elsewhere, or to provide alternative addresses where members and staff may be contacted.

Figure 5.17: *This image is the segmentation of fig. C.1 by the genetic programming based system. Pass 1,2,3 and 4 have been performed. Blue indicates that the region belongs to the text component and green indicates that the region belongs to the image component.*

This process expresses the segmentation map in terms of square areas. This is done because image and text blocks usually appear in rectangles. See fig. 5.17 for the resulting segmentation. This form of post-processing is not performed by default. Algorithm 13 describes the steps taken during this pass.

Algorithm 13 Pass 4:

```
(1) Perform connected component analysis on all regions (including background).
(2) Label each connected region and determine each connected region's bounding rectangle (the set of bounding rectangles is known as BR). The bounding rectangles are used to determine which regions overlap and which regions are contained within each other.
(3) For each connected region, if there is another connected region inside the bounding rectangle:
    (.1) Split the two clashing bounding rectangles into however many pieces are necessary, such that the 'clash' region is no longer part of the bounding rectangle. This entails deleting the two clashing rectangles from BR and adding all of the new bounding rectangles resulting from the split to BR.
    (.2) Copy the region of the image where there is a clash between the two connected components bounding rectangles into a smaller array SX.
    (.3) Perform connected component analysis on SX.
    (.4) Label each connected region and determine each connected region's bounding rectangle. Add the bounding rectangles to BR.
    (.5) For each connected region, if there is another connected region inside the bounding rectangle:
        (.1) Determine which of the two regions in the overlapping region are in the majority.
        (.2) The region that is in the minority in the overlapping region has its bounding rectangle split into however many necessary smaller rectangles such that the region that is overlapping may be removed. This entails deleting the minority rectangle and adding all of the rectangles resulting from the minority rectangle being split so as to no longer clash with the majority rectangle.
(4) Draw all bounding rectangles in BR.
```

Chapter 6

Experimental results and discussion

6.1 Data set

391465 sub-images from thirty-two large document images (taken from the MediaTeam document image database [1]) were classified by six different algorithms (five genetic programs and the K-Means/GLCM based algorithm) in order to test the accuracies of the genetic programs and the post-processed K-Means algorithm. The images were all chosen to contain text, background and image components (since the K-Means based algorithm attempts to cluster the data into three segments). The thirty-two document images have each been segmented with the five separately created genetic programs, and each document image has been segmented five separate times with the post-processed K-Means algorithm. Five separate runs of the post-processed K-Means based algorithm are performed on each image in order to determine the post-processed K-Means based algorithm's average accuracy, since separate runs of the post-processed K-Means based algorithm can produce different results.

The document images are taken from a large variety of sources, resulting in a variety of Haralick features distributions (this is done in order to make the job of the genetic programming based segmentation algorithm more challenging). The bulk of the input images are scanned newspaper images (see appendix C) since:

– Newspaper document-images have a fair amount of background, text and images.

– Scanned images are generally very noisy and newspaper quality printing and paper makes the job of the classifiers somewhat more difficult.

– The processing of scanned newspaper images for the purpose of archiving is one of the major applications of such classifiers.

6.2 Genetic program creation

Five different sets of training data were used when training the genetic programs. For each set of training data, multiple genetic programs were created. Out of those genetic programs, one was selected for each set of training data. This resulted in five different genetic programs (for comparison with five runs of the K-Means based algorithm) created from five different sets of training data. Each genetic program is named after the document image used for its training. The five genetic programs have been tested extensively in order to show the different results from the different segmentation algorithms.

When training the genetic programs in table 6.1, care was taken in choosing images which provide a good general example of how text (both title text and regular text), background and images look like on average. Well trained genetic programs are very rarely outperformed by the post-processed K-Means algorithm on the image that the genetic program was trained on. The genetic programs' performance on other images depends on how similar the training image is (in terms of the representation of text, background and image) to the image to be segmented.

Table 6.1: *This table shows the parameters used for the creation of the genetic programs shown in the results tables. The parameters listed are the parameters that differ from the default parameters of the genetic training system (see table 5.1). The parameters used were chosen for a medium length run time. Each genetic program name is the same as the name of the file used to train the genetic program.*

Genetic program	Seed	Max individual height	Population size	Number of generations
th_00527	1176064598	3	100	20000
th_00531	1176043820	4	100	20000
th_00545	1176116394	3	100	20000
th_00564	1176112366	3	100	20000
th_00839	1176071382	3	50	20000

6.3 Methods used for comparison of results

In the discussion of the results of the system Receiver Operator Characteristic (ROC) space graphs (discussed in section 6.4.1) and some numerical measures (discussed in section

6.4) are used. The ROC space graphs and the numerical measures are determined by the confusion matrices of the results (discussed in section 6.3.2).

6.3.1 Basic terminology

True Positives (TP)

The number of correct classifications in which a region is classified as belonging to the extracted class.

False Positives (TP)

The number of classifications in which a region is incorrectly classified as belonging to the extracted class.

False Negatives (FN)

The number of classifications in which a region is incorrectly classified as not belonging to the extracted class.

True Negatives (TP)

The number of classifications in which a region is correctly classified as not belonging to the extracted class.

6.3.2 Confusion matrices

Confusion matrices are commonly used in the field of artificial intelligence when it is necessary to determine the accuracy of classification systems, as well as other disciplines where it is necessary to determine the accuracy of tests (such as medicine, psychology and economics)[34, 94]. Since our system classifies regions as belonging to, or not belonging to a class, the system has four possible results (in terms of confusion matrices) for each classification:

- a true positive result (a correct assignment of a region to a class).
- a true negative result (a correct assignment of a region as belonging to a class).

– a false positive result (an incorrect assignment of a region as belonging to a class, when the region does not actually belong to the class).

– a false negative result (an incorrect assignment of a region as not belonging to a class, when the region does actually belong to the class).

In our case, the confusion matrices will look as follows (table 6.2):

Table 6.2: *The format of the confusion matrices used in this dissertation. p is an actual positive value, n is an actual negative value, p' is a predicted positive value, n' is a predicted negative value, P' is the total number of predicted positive values, N' is the total number of predicted negative values, P is the total number of actual positive values, and N is the total number of actual negative values. TP,TN,FP,FN are described in section 6.3.1*

		Actual values		
		p	n	
Predicted values	p'	TP	FP	P'
	n'	FN	TN	N'
		P	N	

6.4 Measures derived from a confusion matrix

For this, and more information on ROC analysis, consult [94, 34].

True Positive Rate (TPR)

The rate of correct classifications.

$$TPR = TP/P \tag{6.1}$$

False Positive Rate (FPR)

The rate of incorrect positive classifications.

$$FPR = FP/N \tag{6.2}$$

Accuracy (ACC)

The rate of correct classifications out of all classifications.

$$ACC = (TP + TN)/(P + N) \tag{6.3}$$

Precision (PPV)

The rate of true positive classifications out of all positive classifications.

$$PPV = TP/(TP + FP) \tag{6.4}$$

6.4.1 Receiver Operator Characteristic (ROC) graphs

A receiver operating characteristic graph is a technique used to visualize the performance of classifiers. ROC graphs are commonly used to depict hit rates and false alarm rates of classifiers as well as for visualizing and analyzing the performance of diagnostic systems. ROC analysis has been used in the fields of medicine, radiology, psychology, machine learning, data mining, amongst others[34, 94]. In the case of the systems implemented, the results can be visualized as points on ROC space. A ROC space plot is generally of TPR against FPR, as shown in figure 6.1.

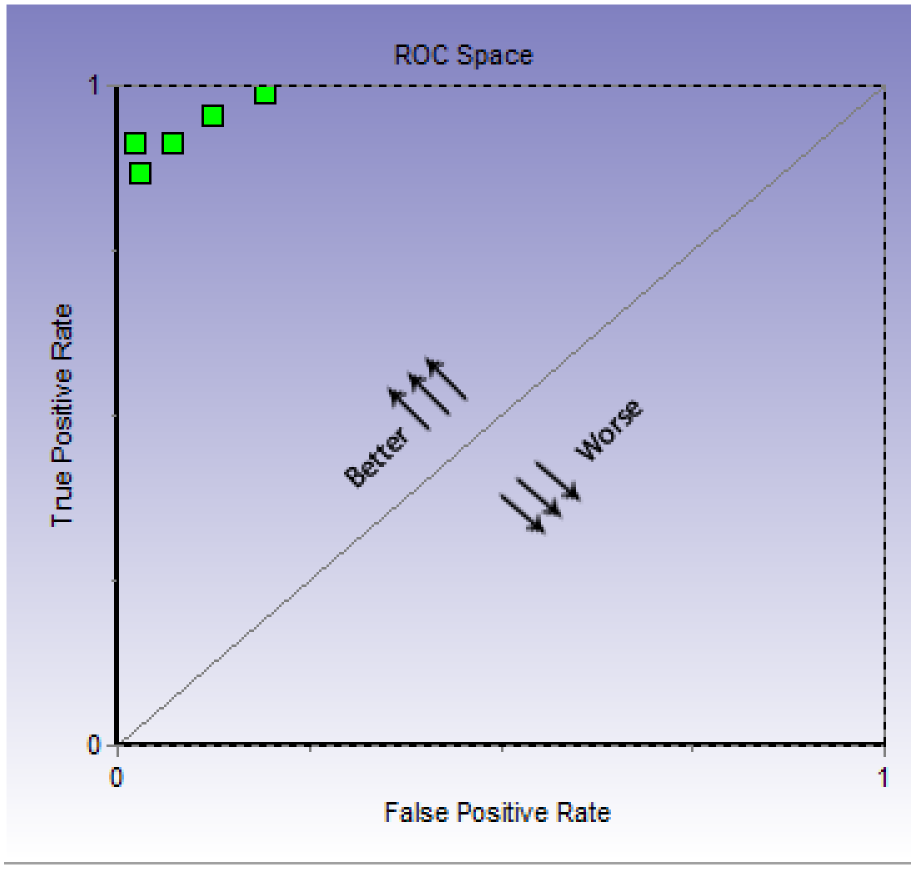


Figure 6.1: *Typical ROC graph output from the system implemented.*

6.5 Individual results

Some individual results from the K-Means algorithm and the genetic algorithms can be viewed in appendices D and E. The original images used to produce the results can be viewed in appendix C. Due to space limitations, not all of the results (32 large images, each segmented ten times) can be shown in this dissertation. However, all results discussed can be generated with the application on the attached compact disc (the disc also contains the image files that were segmented as well as the genetic programs used to segment them).

6.5.1 Result comparison

Image extraction

In terms of accuracy, the post-processed K-Means algorithm performed as well as two of the genetic programs (”th_00531” and ”th_00545”), while being out performed by two others. Genetic program ”th_00527” clearly outperforms the post-processed K-Means algorithm in terms of accuracy (see table 6.3 and figure 6.2). Appendix B illustrates the results of each algorithm in detail via ROC graphs. Table 6.3 gives a full list of results obtained when extracting the image segment of the document images in the data set.

Text extraction

The post-processed K-Means algorithm was out-performed by all five of the genetic programs. Appendix B illustrates the results of each algorithm in detail via ROC graphs. Table 6.4 gives a full list of results obtained when extracting the image segment of the document images in the data set.

6.5.2 Run-time comparison

The major difference between the two methods (in terms of run-time) lies in the amount of pre-calculation necessary to obtain good results. The number of features used by the genetic program is far lower than the number of features used by the GLCM/K-Means based algorithm (since the genetic program is a type of feature selection and merging algorithm). The post-processed K-Means segmentations shown have all used sixty four features. The

Table 6.3: *Accuracy values obtained when extracting the image component of large document images via various methods. The post-processed K-Means based algorithm was run five times on each data set and the average result was used. The genetic programs (denoted by a 'G' prefix) are numbered according to their source of training data (eg. G527 is a genetic algorithm trained from image th_P00527).*

Data set	K-Means	G531	G839	G564	G545	G527
th_P00531	0.91	0.95	0.95	0.92	0.92	0.93
th_P00539	0.91	0.95	0.93	0.91	0.94	0.97
th_P00527	0.96	0.96	0.95	0.91	0.95	0.97
th_P00839	0.83	0.91	0.93	0.8	0.86	0.87
th_P00837	0.82	0.96	0.94	0.97	0.98	0.97
th_P00825	0.98	0.97	0.97	0.96	0.98	0.97
th_P00836	0.92	0.94	0.95	0.97	0.97	0.93
th_P00826	0.98	0.95	0.96	0.96	0.97	0.98
th_P00805	0.82	0.47	0.83	0.37	0.47	0.99
th_P00804	0.82	0.75	0.83	0.8	0.77	0.8
th_P00617	0.97	0.98	0.97	0.95	0.97	0.98
th_P00616	0.96	0.97	0.98	0.95	0.96	0.96
th_P00611	0.97	0.97	0.98	0.9	0.94	0.98
th_P00610	0.97	0.96	0.97	0.93	0.95	0.96
th_P00607	0.95	0.97	0.97	0.96	0.98	0.98
th_P00606	0.99	0.98	0.98	0.94	0.97	0.97
th_P00605	0.99	0.99	0.99	0.97	0.98	0.97
th_P00566	0.9	0.88	0.74	0.95	0.9	0.9
th_P00565	0.92	0.84	0.65	0.97	0.89	0.84
th_P00564	0.9	0.79	0.63	0.98	0.89	0.82
th_P00563	0.95	0.95	0.89	0.98	0.94	0.94
th_P00562	0.95	0.96	0.9	0.99	0.95	0.95
th_P00555	0.91	0.95	0.96	0.88	0.92	0.94
th_P00545	0.92	0.92	0.92	0.94	0.98	0.93
th_P00542	0.87	0.94	0.93	0.89	0.92	0.94
th_P00541	0.83	0.91	0.9	0.85	0.89	0.88
th_P00540	0.95	0.91	0.97	0.83	0.89	1
th_P00535	0.97	0.97	0.96	0.85	0.92	0.99
th_P00534	0.97	0.96	0.96	0.9	0.92	0.98
th_P00533	0.8	0.87	0.9	0.85	0.88	0.91
th_P00528	0.95	0.97	0.97	0.98	0.98	0.98
th_P00827	0.89	0.94	0.95	0.96	0.97	0.96
AVERAGE	0.92	0.92	0.92	0.91	0.92	0.94

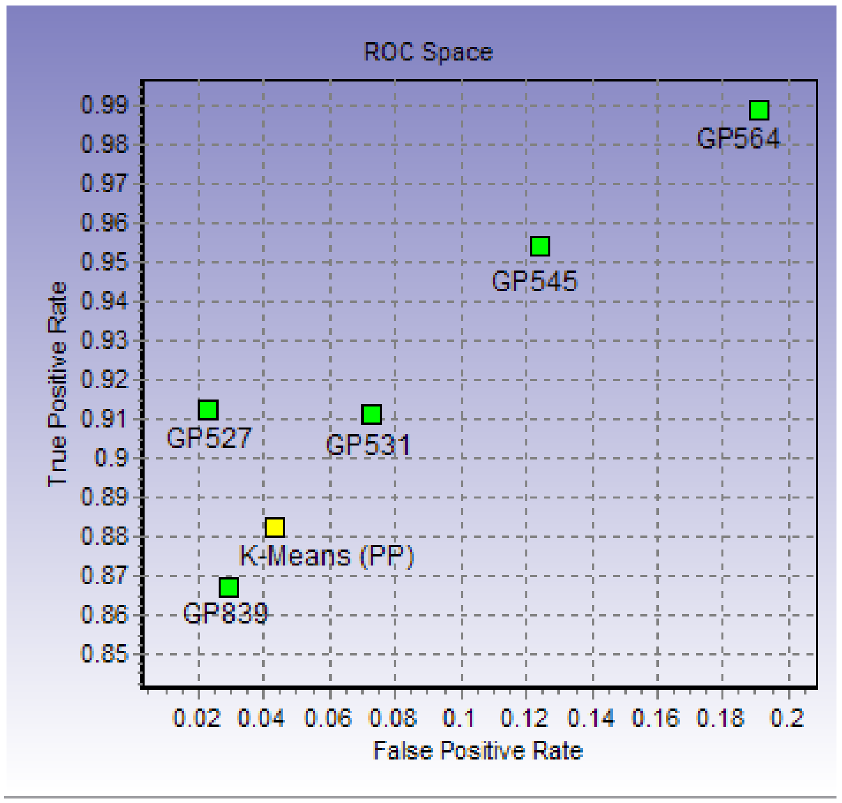


Figure 6.2: *This figure illustrates the True Positive Rate (TPR) and False Positive Rate (FPR) of the image extraction algorithms via a ROC graph.*

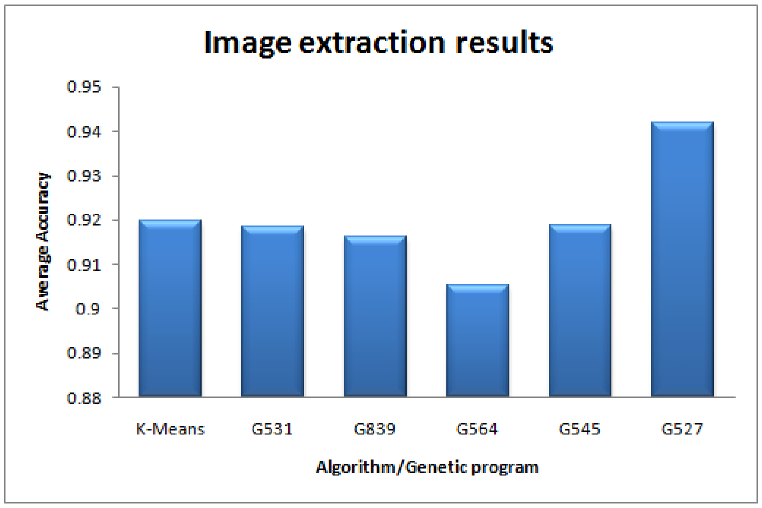


Figure 6.3: *A comparison of the average accuracies of the image extraction methods via a bar graph.*

Table 6.4: *Accuracy values obtained when extracting the text component of large document images via various methods. The post-processed K-Means based algorithm was run five times on each data set and the average result was used. The genetic programs (denoted by a 'G' prefix) are numbered according to their source of training data (eg. G527 is a genetic algorithm trained from image th_P00527).*

Data set	K-Means	G531	G839	G564	G545	G527
th_P00531	0.91	0.98	0.96	0.98	0.95	0.98
th_P00539	0.88	0.9	0.95	0.87	0.95	0.94
th_P00527	1	1	0.99	0.96	0.98	1
th_P00839	0.84	0.9	0.93	0.84	0.87	0.88
th_P00837	0.84	0.99	0.99	0.98	0.99	1
th_P00825	0.99	0.98	0.98	0.96	0.98	0.99
th_P00836	0.93	0.94	0.97	0.96	0.96	0.93
th_P00826	0.95	0.94	0.94	0.92	0.95	0.95
th_P00805	0.7	0.41	0.75	0.41	0.49	0.91
th_P00804	0.84	0.84	0.82	0.85	0.84	0.87
th_P00617	0.94	0.93	0.94	0.9	0.93	0.97
th_P00616	0.94	0.96	0.97	0.94	0.98	0.98
th_P00611	0.91	0.89	0.93	0.83	0.91	0.97
th_P00610	0.96	0.94	0.94	0.9	0.89	0.93
th_P00607	0.96	0.97	0.99	0.95	0.97	0.98
th_P00606	0.9	0.9	0.9	0.88	0.87	0.9
th_P00605	0.95	0.95	0.96	0.93	0.96	0.96
th_P00566	0.91	0.94	0.66	0.97	0.84	0.91
th_P00565	0.92	0.93	0.6	0.98	0.85	0.91
th_P00564	0.9	0.95	0.64	0.99	0.87	0.93
th_P00563	0.92	0.97	0.86	0.97	0.91	0.93
th_P00562	0.97	0.97	0.88	0.99	0.92	0.93
th_P00555	0.8	0.9	0.94	0.91	0.91	0.9
th_P00545	0.83	0.89	0.92	0.98	1	0.92
th_P00542	0.84	0.93	0.95	0.92	0.92	0.92
th_P00541	0.8	0.98	0.91	0.97	0.96	0.96
th_P00540	0.87	0.87	0.95	0.8	0.87	0.97
th_P00535	0.96	0.96	0.96	0.87	0.91	0.98
th_P00534	0.98	0.98	1	0.9	0.91	1
th_P00533	0.82	0.9	0.94	0.84	0.88	0.94
th_P00528	0.99	0.99	1	0.98	0.98	0.99
th_P00827	0.78	0.96	0.97	0.95	0.97	0.96
AVERAGE	0.90	0.92	0.91	0.91	0.91	0.95

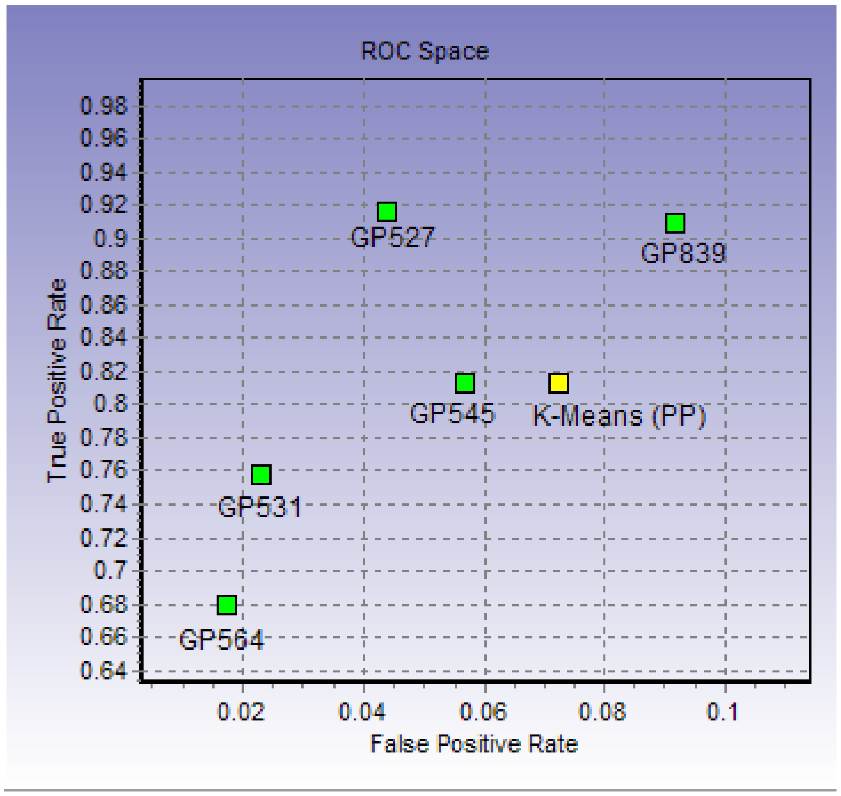


Figure 6.4: *The True Positive Rate (TPR) and False Positive Rate (FPR) of the text extraction algorithms illustrated via a ROC graph.*

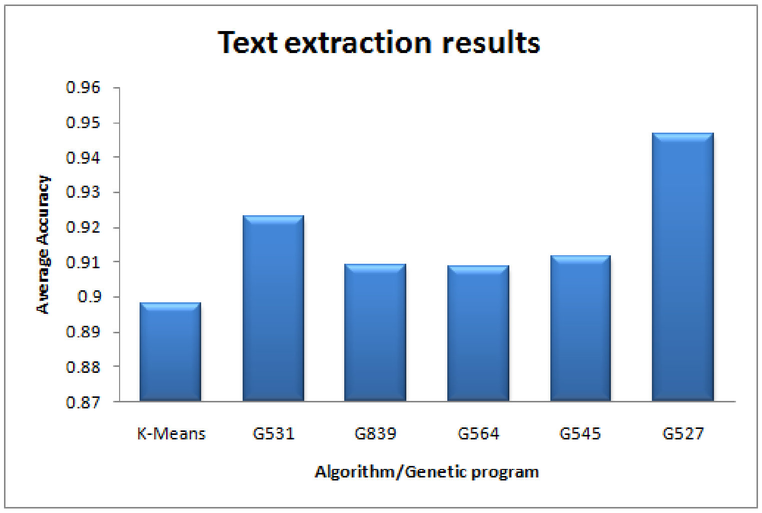


Figure 6.5: *A comparison of the average accuracies of the text extraction methods via a bar graph.*

genetic programming based segmentations generally use fewer than ten features to perform segmentations (depending on the genetic parameters). While it is possible to lower the number of features the post-processed K-Means based segmentation algorithm uses, the quality of the segmentation is likely to decrease without feature selection being applied to K-Means based algorithm.

On an $Intel^{(R)}$ $Centrino^{(R)}$ Duo T2400 running at 1.83GHz, each segmentation algorithm (the K-Means based algorithm or a genetic program) takes around one second to run on an large sized image (around 800×1000 pixels) without the Haralick feature calculations. The sixty four used Haralick features of an image are calculated at a rate of one million pixels per minute (with a block size of 16×16). It is unnecessary to calculate so many Haralick features, and when dealing with a batched process of a large number of document images to be segmented, such unnecessary calculations could turn out to be very time consuming. The genetic programs perform the segmentations using only a small subset of features, all of which are relevant to the task, whereas the post-processed K-Means algorithm performs no feature selection and thus uses an unnecessarily large group of features to perform segmentations.

6.5.3 Qualitative comparison

In order to compare the quality of the segmentation systems implemented, the system was used to segment an image that was segmented by *Dong et al.*[31]. The method used by *Dong et al.* is discussed in section 2.4. See figure 6.6 for a visual comparison of the segmentation results.

Dong et al.'s method seems to be rather accurate. Judging by the visual output of their system, they appear to prioritise the avoidance of false negative image and text classifications, resulting in somewhat more false positive classifications of image and text regions. The genetic programming based system's heuristic attempts to optimise overall accuracy, treating false negative and false positive classifications equally.

The genetic programming based method is also designed to work with somewhat noisier images than image 6.6 (such as newspaper and magazine scans), and thus will not perform as accurately as a segmentation method which could be aimed at segmenting web page images. For example, some of the images in the document image have a light grey background. This could be interpreted as a noisy background, or as part of an image region. If it is known that a web page is being segmented, one can assume that the background will be made up of a smooth single colour, leading to extremely simple background classification at the expense of noise tolerance.

Despite the genetic program being trained to segment images with a lot of noise, it still performs very well when segmenting the web page in figure 6.6.

6.5.4 Statistical significance

In order to prove the statistical significance of the results, the Z-test is used (see equations 6.5 and 6.6). A confidence level of 95% will be used (an alpha value of 0.05). The Z-score for an alpha value of 0.05 is 1.65.

$$SE = \frac{\sigma}{\sqrt{n}} \tag{6.5}$$

$$z = \frac{x - \mu}{SE} \tag{6.6}$$

Using the results (table 6.4 for text classification and table 6.3 for image classification) from the segmentation of the document images via the K-Means algorithm and the genetic programming based algorithm (genetic program G527):

Text classification statistical significance

$$\text{sample size}(n) = 32$$

$$\text{sample mean}(m) = 0.9466$$

$$\text{k-means mean}(\mu) = 0.8978$$

$$\text{k-means standard deviation}(\sigma) = 0.0718$$

$$\text{Standard Error } SE = 0.0718/\sqrt{32}$$

$$SE = 0.0127$$

$$z = \frac{0.9466 - 0.8978}{SE}$$

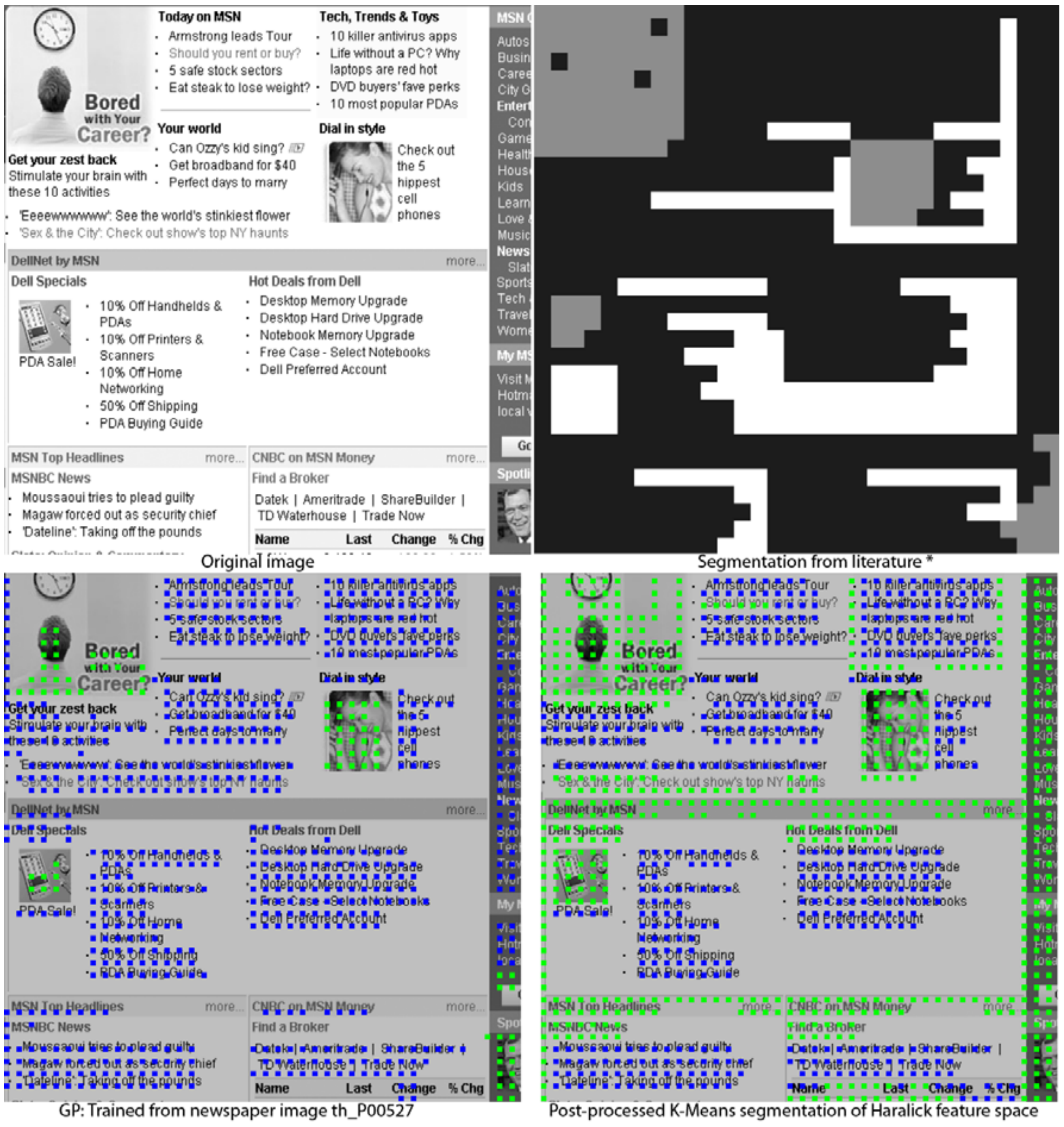

Original image
Segmentation from literature *
GP: Trained from newspaper image th_P00527
Post-processed K-Means segmentation of Haralick feature space

Figure 6.6: *The top left image is the original image taken from [31]. The top right image is the image segmented via* Dong et al.'s *system (taken from [31]). In the top right image, the black areas represent the text class, grey areas represent the image class and white areas represent the background class. The bottom left image is the original image segmented by a genetic program trained from the newspaper training set. The bottom right image is the original image segmented by the post-processed K-Means/Haralick feature based system used for comparison in this dissertation.*

$$z = \frac{0.0488}{0.0127}$$

$$z = 3.8425$$

The value from the Z-table for $z = 3.8425$ is greater than the Z value of 1.65 required for a 95% confidence level. It can be said that the performance of the GP based document image segmentation algorithm is better than that of the GLCM/K-Means based algorithm for text segment classification with a 95% confidence level.

Image classification statistical significance

$$\text{sample size}(n) = 32$$

$$\text{sample mean}(m) = 0.9419$$

$$\text{k-means mean}(\mu) = 0.9197$$

$$\text{k-means standard deviation}(\sigma) = 0.0574$$

$$SE = 0.0574/\sqrt{32}$$

$$SE = 0.0101$$

$$z = \frac{0.9419 - 0.9197}{SE}$$

$$z = \frac{0.0222}{0.0101}$$

$$z = 2.1980$$

The value from the Z-table for $z = 2.1980$ is greater than the Z value of 1.65 required for a 95% confidence level. It can be said that the performance of the GP based document image segmentation algorithm is better than that of the GLCM/K-Means based algorithm for image segment classification with a 95% confidence level.

6.6 Interpretation of trained genetic programs:

A sample size of twenty genetic program pairs have been examined in order to find trends in the feature, angle, distance triples that they use.

6.6.1 Image

The most common Haralick feature used in the image segmentation genetic programs is mean with 42 out of 93 triples, followed by contrast with 21 out of the 93 triples. Mean is regarded as a useful Haralick feature when extracting the image region of a document image, the large amount of contrast triples is due to contrasts text extraction ability, which would be used for text removal in the genetic programs. The distance parameter of 2 is the most frequently occurring with 68 out of 93 triples. No particular angle seems to be dominant when segmenting images. The individuals examined had on average 4.65 triples each.

6.6.2 Text

The most common Haralick feature present in the text segmentation genetic programs discussed in this chapter is contrast/inertia with 51 out of 67 triples. This is an expected result since contrast/inertia measures local variance and is known to be a useful feature for segmenting text. The distance parameter of 1 is the most frequently occurring with 45 out of 67 triples. Angles of 0 and 90 degrees are equally dominant when segmenting text with 60 out of 67 triples having angles of 0 or 90 degrees, this is an expected result due to the highly directional nature of text. The individuals examined had on average 3.4 triples each.

6.7 Random individual generation vs. genetic programming based individual generation

Due to the manner in which the terminals are constructed it is not unusual for randomly created individuals to achieve high fitness values.

– Since each terminal represents a segmentation of a component of the target image by a Haralick feature, angle, distance triple, these segmentations sometimes achieve good results by themselves. The aim of the system is to create better segmentations by merging these simpler segmentations together, yielding an improved fitness over an already potentially good fitness.

– Since there are a lot of terminals that may individually have a high fitness, there are potentially many combinations that yield good results present in the search space.

This increases the odds of a good random solution being found.

– Improvements of fitness over the initially generated members appear small, since an individual with a good fitness is relatively common; however, an individual with a fitness of a few % higher can be extremely rare (see figure 6.7) and unlikely to be found with a random search.

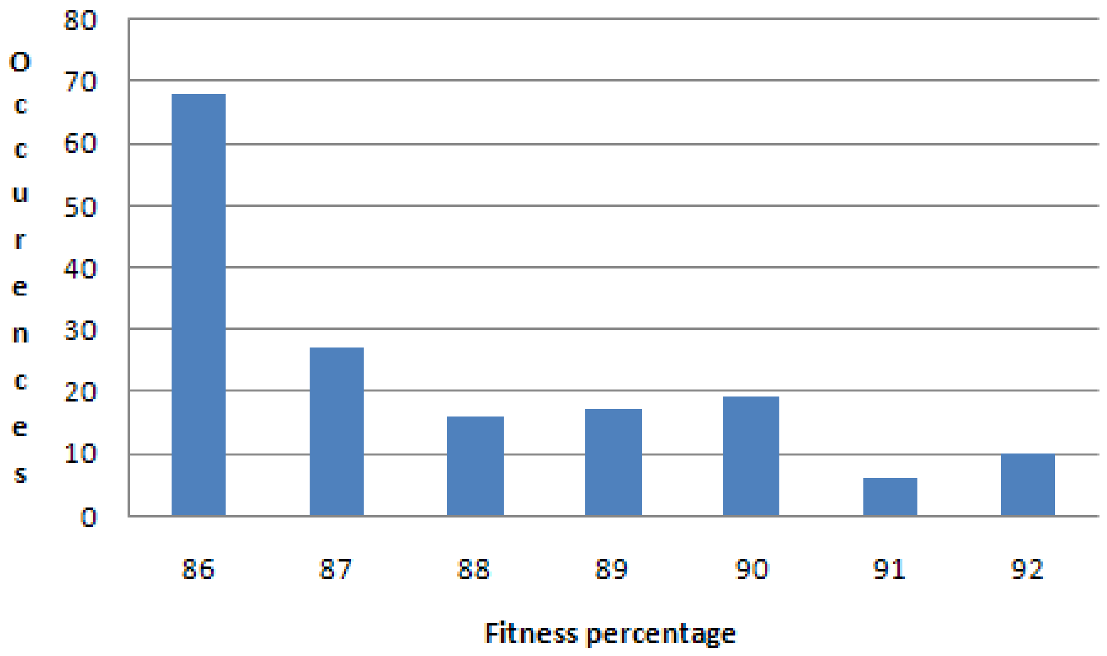


Figure 6.7: *This figure shows the occurrences of the top fitness values out of a sample of 30000 randomly created individuals. The maximum fitness value is 92% (the top 0.03% of the population). The minimum fitness value on the graph is 86%, representing 0.27% of the population. Run parameters are: two separate runs of 1 generation length, 3 populations of 5000 individuals with the applications default parameter values and seeds of 1205095725 and 1205099401. The training data set used was ”th_P00531”.*

Chapter 7

Conclusion and future work

In this dissertation, the problems examined include feature extraction, classification, and more specifically feature extraction and classification applied to the problem of document image segmentation.

The problem description was formulated by performing a survey of literature related to document image segmentation in the fields of genetic programming, classification and more general image processing. Document image segmentation using genetic programming appears to be a fairly new field as there is a relatively small amount of information available on the subject compared to other document image segmentation techniques.

In this dissertation, some common methods of feature extraction have been discussed, including Markov random fields, discrete wavelet transforms, Gabor filtering and grey level co-occurrence matrices (GLCM) with Haralick features. This research has focused on the use of Haralick features extracted from grey level co-occurrence matrices for performing feature extraction on document images and has used it as part of the genetic programming approach as well as a part of the K-Means based method for comparison against the genetic programming based system.

In terms of discussion and what has been implemented, classification and clustering techniques have been approached with the goal of segmenting the Haralick feature space. Such techniques include the K-Means and C-Means algorithms. However, there are several problems with the K-Means and C-Means algorithms in terms of robustness, accuracy and efficiency (as discussed in sections 3.4.1 and 3.4.2).

In this dissertation, a genetic programming based solution that performs segmentations and combines them according to the particular genetic program in order to solve the document image segmentation problem has been presented. The segmentations that the genetic programs use can be from the K-Means algorithm or even a combination of any segmentation algorithms that express their segmentation results in the same format. However, for the purpose of this project, the generated genetic programs only use K-Means segmentations.

From experiments performed (see table 6.4) using the post-processed K-Means based method and the genetic programming based method, it appears that the genetic programming based method is capable of producing better and more consistent results (when provided with training data that sufficiently represents the text, image and background classes) than the post-processed K-Means based algorithm. The genetic programs, once trained, are also able to make more efficient use of the Haralick features, possibly resulting in a lower total segmentation time since, on average, fewer Haralick features need to be calculated for a segmentation of comparable quality. This is due to the feature selection performed by the GP based method when applied to K-Means segmentations of Haralick features.

In this dissertation, it has been shown that genetic programming can be used to combine various document image segmentation inputs, potentially from a number of different sources. It has also been demonstrated that genetic programming has the potential to improve over existing techniques in the field of document image segmentation

7.1 Limitations

Although improvements have been presented (as mentioned in section 7.2), the system has some limitations:

(1) Very large differences between the distribution of Haralick features in the training image and the image to be segmented could result in lower accuracy. Such cases are uncommon.

(2) Thought needs to be given to the choice of training image. The training image needs to be as representative as possible when compared to the rest of the images to be segmented (for example: the training image needs to have a reasonable amount of the text, image and background components).

7.2 Future work

Possible future work could include:

(1) The analysis of a larger variety of images for different types of texture segmentation.

(2) The introduction of heuristics that take execution and feature calculation speed into account. This would be very useful for making segmentation algorithms that are very fast, but quite accurate.

(3) Due to the manner in which the genetic system combines features, it is possible to apply parallel processing in order to increase the speed of the segmentations.

(4) The genetic training system is able to take input from sources other than K-Means segmentations of Haralick features (such as images segmented using Gabor filters or discrete wavelet transforms). The combination of various segmentation techniques could yield interesting results.

(5) The majority of the data for the system is stored in separate files (such as the genetic programs, test images, testing data, etc), while results are stored in the results database. The results database could be enhanced to store all of the data pertaining to the program and possibly converted to a SQL database (as opposed to its current text based incarnation). More statistical functionality could then be added to the result viewing sub-program.

(6) Expansion of the genetic system to allow a genetic program to train on several training images simultaneously.

(7) Expansion of the system to allow it to work with multiple block sizes simultaneously.

Bibliography

[1] The mediateam document database. (available at mediateam.oulu.fi/downloads/mtdb/download.html).

[2] Ajith Abraham, N. Nedjah, and L. Mourelle. Evolutionary computation: from genetic algorithms to genetic programming. *Genetic Systems Programming*, 2006.

[3] M.R. Anderberg. *Cluster analysis for applications*. Academic Press Inc., 1973.

[4] D. Anguelov, B. Taskar, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng. Discriminative learning of markov random fields for segmentation of 3d scan data. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 169–176, Washington, DC, USA, 2005. IEEE Computer Society.

[5] C. A. Ankenbrandt, B. P. Buckles, and E. E. Petry. Scene recognition using genetic algorithms with semantic nets. *Pattern Recognition Letters*, 11:285–293, April 1990.

[6] D. Arthur and S. Vassilvitskii. How slow is the k-means method? In *In Proc. 22nd SoCG*, pages 144–153, 2006.

[7] D. Arthur and S.i Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford University, 2006.

[8] D. Ashlock. Data crawlers for simple optical character recognition. In *Proc. of the 2000 Congress on Evolutionary Computation*, pages 706–713, Piscataway, NJ, 2000. IEEE Service Center.

[9] A. P. Asirvatham. Script segmentation of multi-script documents, bachelors thesis, international institute of information technology, 2002.

[10] A. K. Bachoo and J. R. Tapamo. Texture analysis and unsupervised clustering for segmenting iris images. *PRASA 2005*, pages 157–163, 2005.

[11] T. Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford Univ. Press., 1996.

[12] P. Baldi and G.W. Hatfield. *DNA Microarrays and Gene Expression*. Cambridge Univ. Press, 2002.

[13] S. Bandyopadhyay and U. Maulik. An evolutionary technique based on k-means algorithm for optimal clustering in $\mathbb{R}^N$. *Information SciencesApplications: An International Journal*, 146:221–237, 2002.

[14] A. Barta and I. Vajk. Document image analysis by probabilistic network and circuit diagram extraction. *Informatica*, 29:291301, 2005.

[15] B. Bhanu, S. Lee, and J. Ming. Adaptive image segmentation using a genetic algorithm. *Image Understanding Workshop*, pages 1043–1055, 1989.

[16] S.K. Bhatia and J.S. Deogun. Conceptual clustering in information retrieval. *IEEE Trans. Systems, Man, and Cybernetics Part B*, 28:427–436, 1998.

[17] S. Bhattacharjee, Y. Chen, and A. Jain. On texture in document images. *In Proceedings of Computer Vision and Pattern Recognition*, 31:677–680, 1992.

[18] C. A. Bouman and H. Cheng. Document compression using rate-distortion optimized segmentation. *Journal of Electronic Imaging*, 10(2):460–474, April April 2001.

[19] A. C. Bovik and M. Clark. Experiments in segmenting texton patterns using localized spatial filters. *Pattern recognition*, 22:707–717, 1989.

[20] J. Burpee. Genetic programming. In *CSI 5388*, 2005.

[21] J. D. Carroll and A Chaturvedi. A feature-based approach to market segmentation via overlapping k-centroids clustering. *J. Marketing Research*, 34(3):247–254, 1997.

[22] W. Chan and J. Sivaswamy. Local energy analysis for text script classification. In *Image and Vision Computing New Zealand*, 99.

[23] F. Chang, C. H. Chou, E. Lai, and M. C. Su. A reinforcement-learning approach to color quantization. In *Intelligent Systems and Control*, 2004.

[24] S. Chen and D. Zhang. A novel kernelized fuzzy c-means algorithm with application in medical image segmentation. *Artificial Intelligence in Medicine*, 32:37–50, 2004.

[25] D. N. Chun and H. S. Yang. Robust image segmentation using genetic algorithm with a fuzzy measure. *Pattern Recognition*, 29(7):1195–1211., July July.

[26] V. Ciesielski, A. Innes, S. John, and J. Mamutil. Genetic programming for landmark detection in cephalometric radiology images. *International Journal of Knowledge-Based Intelligent Engineering Systems*, 7(3):401–409, 2003.

[27] Wikipedia Contributors. Wikipedia entry on markov random fields. Available at: (http://en.wikipedia.org/wiki/Markov_network), 2006.

[28] G. R. J. Cooper. The textural analysis of gravity data using co-occurence matrices. *Computers and Geosciences*, 30:107–115, 2004.

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

[30] J. Dombi and M. Jelasity. Gas, a concept on modeling species in genetic algorithms. *Artificial Intelligence*, 99:1–19, 1998.

[31] Y. Dong, G. Fan, L. Liu, and X. Song. A simplified quantization rate-distortion model for fast document image segmentation. Technical report, School of Electrical and Computer Engineering, Oklahoma State University, Stillwater.

[32] A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing*. Springer, 2003.

[33] G. Fan and X. G. Xia. A joint multi-context and multiscale approach to bayesian image segmentation. *IEEE Trans. Geoscience and Remote Sensing*, 39(12), December 2001.

[34] T. Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27:861874, 2006.

[35] L. A. Fletcher and R. Kasturi. A robust algorithm for text string separation from mixed text/graphics images. 10(6):910–918, 1988.

[36] K. Franke and M. Koppen. A framework for document pre-processing in forensic handwriting analysis. In *Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition*, pages 73–82, September 2000.

[37] H. Frigui and R. Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5(5):450–465, May 1999.

[38] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. . *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[39] U. Grasemann and R. Miikkulainen. Effective image compression using evolved wavelets. GECCO, June 2005.

[40] A. Haar. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69:331–371, 1910.

[41] R.M. Haralick. Statistical and structural approaches to texture. *Proceeding of the IEEE*, 7(5):786804, 1979.

[42] S. J. Hartley. Stephen j. hartley's web page. Available at: (http://www.mcs.drexel.edu/ shartley/), July 2006.

[43] A. Hill and C. J. Taylor. Model-based image interpretation using genetic algorithm. *Image Fis. Comput.*, 10:295–300, June 1992.

[44] J.H. Holland and J.S. Reitman. Cognitive systems based on adaptive algorithms. In D. A. Waterman and F. Hayes-Roth, editors, *Pattern-Directed Inference Systems*. Academic Press, 1978.

[45] M. Iwayama and T. Tokunaga. Cluster-based text categorization: A comparison of category search strategies. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(3):252–264, March 1991.

[46] A. Jain, N. Ratha, and S. Lakshmanan. Object detection using gabor filters. *1997*, 30:295–309, 1997.

[47] A. K. Jain and F. Korrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1991.

[48] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. The analysis of a simple k -means clustering algorithm. In *Symposium on Computational Geometry*, pages 100–109, 2000.

[49] R. Kasturi, L. OGorman, and V. Govindaraju. Document image analysis: A primer. *special issue on Document Processing*, 2002.

[50] Z. Kato and T. Pong. A markov random field image segmentation model using combined color and texture features.

[51] L. Kaufman and P. Rousseeuw. *Finding Groups in Data. An Introduction to Cluster Analysis*. Wiley, 1990.

[52] O. Kia. *Document Image Compression and Analysis*. PhD thesis, University of Maryland at College Park, 1997.

[53] John R. Koza. Genetic programming: On the programming of computers by means of natural selection (complex adaptive systems). The MIT Press, December 1992.

[54] M. Lett and M. Zhang. New fitness functions in genetic programming for object detection. Technical report, Akaroa, New Zealand, November 2004.

[55] C.T. Lia and R. Chiao. Multiresolution genetic clustering algorithm for texture segmentation. *Image and Vision Computing*, 21:955966, 2003.

[56] M.W. Lin, J.R. Tapamo, and B. Ndovie. A texture based segmentation for document segmentation and classification. *South African Computer Journal*, (36):49–56, 2006.

[57] M. Luo, Y. Ma, and H. Zhang. A Spatial Constrained K-Means Approach to Image Segmentation. *Fourth IEEE Pacific-Rim Conference On Multimedia*, pages 738 – 742, December 2003.

[58] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[59] H.C. Martin and A.T. Mario. Simultaneous feature selection and clustering using mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26, 2004.

[60] M.F. McNitt-Gray, N. Wyckoff, J.W. Sayre, J.G. Goldin, and D.R. Aberle. The effects of co-occurrence matrix based texture parameters on the classification of solitary pulmonary nodules imaged on computed tomography. *Computerized Medical Imaging and Graphics*, 23:339, 1999.

[61] P. Moscato. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. Technical report, Caltech Concurrent Computation Program, 1989.

[62] R.F. Murphy, M. Velliste, and G. Porreca. Robust classification of subcellular location patterns in fluorescence microscope image. In *IEEE Intl Workshop Neural Networks Signal Processing*, volume 12, pages 67–76., 2002.

[63] G. Nagy, S. Seth, and S. Stoddard. Document analysis with an expert system. In *In Proceedings of Pattern Recognition in Practicem, Amsterdam*, volume 2, June 1985.

[64] M. Obitko. Introduction to genetic algorithms. Available on: http://cs.felk.cvut.cz/ xobitko/ga/, 1998.

[65] L. O'Gorman and R. Kasturi. *Document Image Analysis*. IEEE Computer Society Press, April 1995.

[66] W. Pedrycz. *Knowledge-Based Clustering*. John Wiley & Sons, Inc., 2005.

[67] J.M. Pena, J.A. Lozano, and P. Larranaga. An empirical comparison of four initialization methods for the k-means algorithm. *Pattern Recognition Letters*, 20:10271040, 1999.

[68] J.M. Pena, J.A. Lozano, and P. Larranaga. An empirical comparison of four initialization methods for the k-means algorithm. *Pattern Recognition Letters*, 20:1027–1040, 1999.

[69] N. Pillay. An introduction to genetic programming (lecture notes). School of Computer Science, University of Kwa-Zulu Natal, 2007.

[70] R. Poli. Discovery of symbolic, neuro symbolic and neural networks with parallel distributed genetic programming. Technical Report CSRP-96-14, August 1996.

[71] R. Poli. Genetic programming for feature detection and image segmentation. In T. C. Fogarty, editor, *Evolutionary Computing*, number 1143, pages 110–125, University of Sussex, UK, 1-2 1996. Springer-Verlag.

[72] R. Poli. Genetic programming for image analysis. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 363–368, Stanford University, CA, USA, FebruaryAugust–MarchJanuary 1996. MIT Press.

[73] R. Polikar. The engineers ultimate guide to wavelet analysis: The wavelet tutorial. (Available at: http://users.rowan.edu/ polikar/WAVELETS/WTtutorial.html).

[74] V. Ramos and F. Muge. Image colour segmentation by genetic algorithms. Technical report, CVRM - Centro de Geosistemas, Instituto Superior Tcnico, Av. Rovisco Pais, Lisboa, Portugal.

[75] S.J. Raudys and A.K. Jain. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(3):252–264, March 1991.

[76] J. Rennard. Introduction to genetic algorithms. Available on: http://www.rennard.org/alife/english/gavintrgb.html, July 2006.

[77] S. V. Rice, J. Kanai, and T.A. Nartker. Report on the accuracy of ocr devices. Technical report, Information Science Research Institute, 1992.

[78] G. Roth and M.D. Levine. A genetic algorithm for primitive extraction. *Proc. 4th Intl Conf. Genetic Algorithm*, pages 487–494, July 1991.

[79] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[80] S.J. Simske. Navigation using hybrid genetic programming: Initial conditions and state transitions. Technical report, Intelligent Enterprise Technologies Lab, HP Laboratories, March 11, 2003.

[81] N. Sinha and A.G. Ramakrishnan. Automation of differential blood count. In *TENCON*, volume 2, pages 547–551, 2003.

[82] W.R. Smart and M. Zhang. Classification strategies for image classification in genetic programming. In *Image and Vision Computing New Zealand*, 2003.

[83] A. Song and V. Ciesielski. Fast texture segmentation using genetic programming. In Ruhul Sarker, Robert Reynolds, Hussein Abbass, Kay Chen Tan, Bob McKay, Daryl Essam, and Tom Gedeon, editors, *Proceedings of the 2003 Congress on Evolutionary Computation CEC2003*, pages 2126–2133, Canberra, 8-12 December 2003. IEEE Press.

[84] A. Song and V. Ciesielski. Texture analysis by genetic programming. In *In Proceedings of the 2004 Congress on Evolutionary Computation (CEC2004)*, 2004.

[85] N. Speer, C. Spieth, and A. Zell. In a memetic clustering algorithm for the functional partition of genes based on the gene ontology. Proceedings of the 2004 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, pp 252-259.

[86] P.K. Spivak. Discovery of optical character recognition algorithms using genetic programming. In John R. Koza, editor, *Genetic Algorithms and Genetic Programming at Stanford 2002*, pages 223–232. Stanford Bookstore, Stanford, California, 94305-3079 USA, June 2002.

[87] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection using evolutionary gabor filter optimization. *IEEE Transations on Intelligent Transportation Systems*, 6(2):125–137, June 2005.

[88] K. Tombre, S. Tabbone, L. Pellissier, B. Lamiroy, and P. Dosch. Text/graphics separation revisited. *Proceedings of the 5th International Workshop on Document Analysis Systems V*, pages 200 – 211, 2002.

[89] K. Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3:1415–1438, 2003.

[90] V. Torra. Fuzzy c-means for fuzzy hierarchical clustering. In *FUZZ-IEEE 2005, Reno, Nevada on May 22-25, 2005,*, pages 646 – 651, 2005.

[91] M. Wall. Introduction to genetic algorithms. Available at: http://lancet.mit.edu/ mbwall/, July 2006.

[92] Q. Wang, T. Xia, C.L. Tan, and L. Li. Directional wavelet approach to remove document image interference. *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, 2:736, 2003.

[93] T.P. Weldon, W.E. Higgins, and Dunn D.F. Gabor filter design for multiple texture segmentation. *Optical Engineering*, 35:2852–2863, 1996.

[94] L.K. Westin. Receiver operating characteristic (roc) analysis. Technical report, Ume University, Sweden.

[95] R. Wilson and M. Spann. *Image Segmentation and Uncertainty*. Research Studies Press, 1988.

[96] K.Y. Wong, R.G. Casey, and F.M. Wahl. Document analysis system. *IBM Journal of Research and Development*, 26(6):647–656, 1982.

[97] F. Wu. A framework for memetic algorithms. Master's thesis, University of Auckland, 2001.

[98] M. Yoshimura and S. Oe. Evolutionary segmentation of texture image using genetic algorithms towards automatic decision of optimum number of segmentation areas. *Pattern Recognition*, 12:2041–2054, 1999.

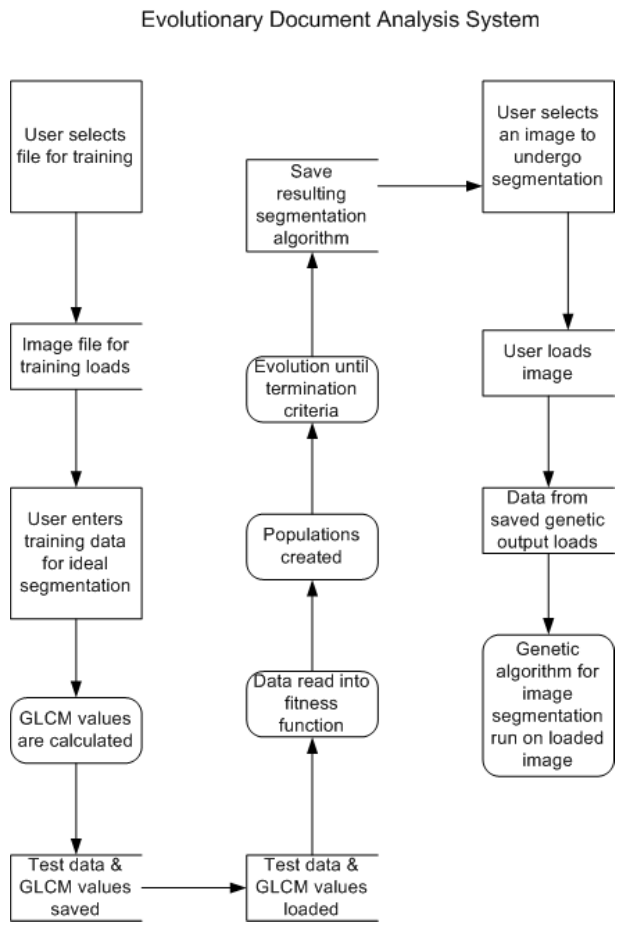
[99] Y. Zheng, H. Li, and D. Doerman. Machine printed text and handwriting identification in noisy document images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):337–353, March 2004.

Appendix A

Implementation details

A.1 Program Design

In this appendix, the three most significant parts of the program shall be discussed: the genetic system, the image processing system and the user interface. The way the different modules of the program interact is best illustrated by the data flow diagram in figure A.1.



Evolutionary Document Analysis System
User selects file for training
Image file for training loads
User enters training data for ideal segmentation
GLCM values are calculated
Test data & GLCM values saved
Test data & GLCM values loaded
Data read into fitness function
Populations created
Evolution until termination criteria
Save resulting segmentation algorithm
User selects an image to undergo segmentation
User loads image
Data from saved genetic output loads
Genetic algorithm for image segmentation run on loaded image

Figure A.1: *The process of learning and interaction between genetic processing and image processing demonstrated via a data flow diagram.*

A.2 The genetic system

The genetic system is illustrated via UML in figure A.2. The whole system has been designed with extendibility and modularity in mind. It is possible by simply inheriting the NodeDescription interface and creating a new class for a new type of problem, to change the type of problem to be solved entirely. For instance, in the UML diagram, the problem definition (in terms of the functions, terminals, individual evaluation and fitness function) for solving Pythagoras's equation is also a subclass of geneticTree, with a different nodeDescription and some other details changed in the geneticTree subclass.

fmGeneticTestingApp

This is the class for the form which controls the input, output and execution for the genetic system. When the run command on the form is used, it sets the genetic parameters and initializes the genetic system. The program then performs a run of the genetic system for each component.

geneticParams

The geneticParams class contains all of the parameters for the genetic system. It contains variables which determine the percentage of crossover and mutation to apply, the termination criteria, the population crossover parameters, the random seed used for the system, a pointer to the progress bar on the user interface, parameters governing the creation and alteration of individuals (such as height and the way they are randomly grown) as well as the number and size of the populations.

geneticNode

This class represents the most basic unit of the genetic system. It is a node on a tree, and it contains functionality to add and remove children, as well as to output all of the nodes in it's sub-tree in text format. The geneticNode class keeps variables to keep track of the number of children a node has, pointers to those children a string representing the value of the node

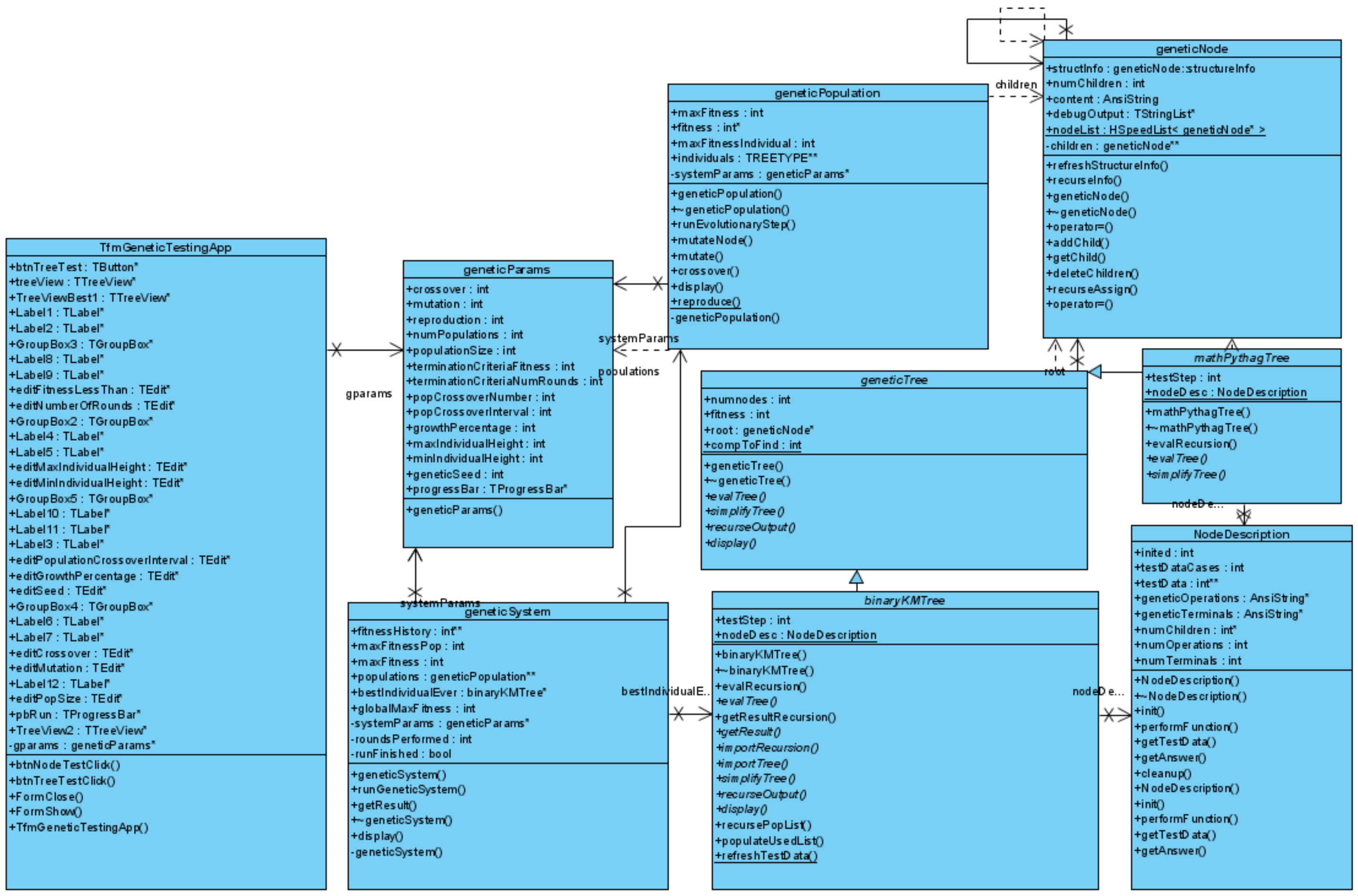


Figure A.2: *The genetic system displayed in UML format.*

(a function or terminal value). The class also has the assignment operator overridden for the purpose of performing assignments from strings directly to the content of the node.

NodeDescription

NodeDescription contains a description of the problem to be solved in terms of the genetic program. It contains variables indicating the number of terminals and functions available. The class also keeps track of the string representation of the function and terminal set. performFunction performs the actions of a specified member of the function set upon however many terminals and returns the result.

geneticTree

The geneticTree class containts pure virtual functions to be overloaded by it's subclasses. The pure virtual functions (evalTree and simplifyTree) are overridden to meet the requirements for each application that the genetic system is designed for. The geneticTree class also has variables to keep track of fitness, the number of nodes in the tree and a pointer to the root node of the tree.

mathPythagTree

mathPythagTree was the initial testing class for the application. A mathematics genetic program was chosen since such an application is a pretty standard way of testing GP systems. The evalTree and simplifyTree functions have been overridden appropriately according to the problem.

binaryKMTree

binaryKMTree is the class used to represent the genetic programs that are generated by the genetic system to solve the problem of segmentation. The class contains an instance of NodeDescription which performs the functions of the function set on the terminals and performs evaluation of the individuals. The evalTree and simplifyTree functions have been overridden appropriately according to the problem.

geneticPopulation

The geneticPopulation class controls a population of genetic programs. The class keeps track of the size of the population, the fittest individual in the population, the individuals themselves, as well as having the means to perform evolutionary steps on the population and genetic crossover, mutation and reproduction.

geneticSystem

The geneticSystem class is used to control the populations of the genetic system, perform population crossovers and control the genetic run in general (this is done in runGeneticSystem). It also keeps track of several statistics of the genetic system, such as the fitness for each evolutionary step in each population and the fittest individual in the system.

A.3 The image processing system

The image processing system is comprised of several modules with tasks ranging from filing, to re-colouring and segmentation. The UML for the bulk of the image processing system is shown in figure A.3. The classes comprising the image processing system are discussed in more detail in the remainder of this appendix.

HImageProcessor

The HImageProcessor class functions as the root of all of the image processing classes. It is composed of several classes, one of which has the function of storing data and providing general image information, the rest of which perform image processing operations (the operations have been grouped into modules based on the type of operation).

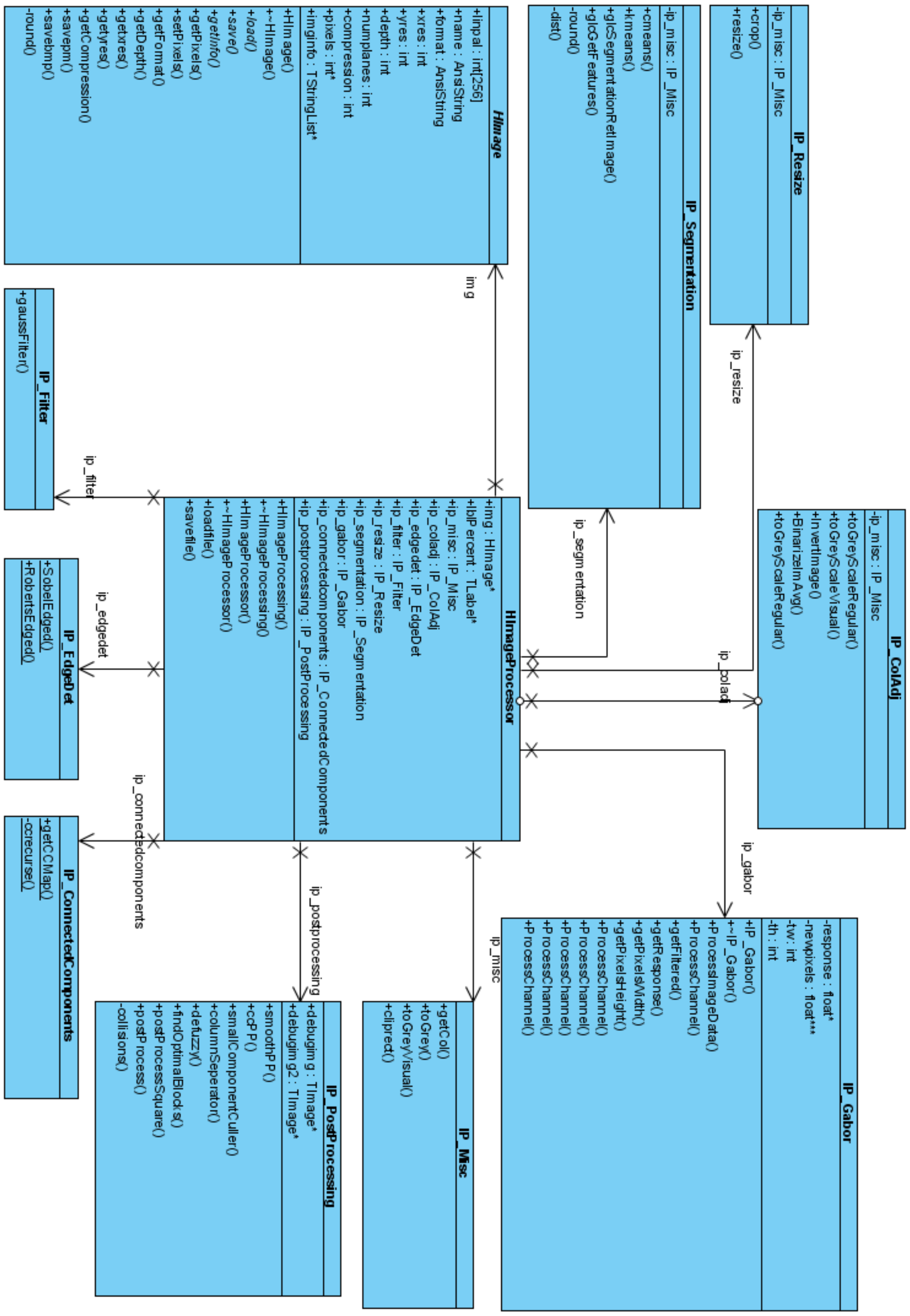


Figure A.3: *The image processing system displayed in UML format.*

HImage

HImage is the a module devoted to storing and managing the image data as well as providing information about the image.

HPm

This class inherits from HImage and is designed to handle the loading of PGM, PBM and PM images.

HBmp

This class inherits from HImage and is designed to handle the loading of BMP images.

IP_Misc

This modules handles some of the more simple image processing function calls, such as clipping and pixel colour adjustments as well as returning components of colours.

IP_Resize

IPResize handles the cropping and scaling of images.

IP_Filter

This module handles the filtering of images.

IP_ColAdj

This module is for adjusting the colouring of images via binarisation, inversions and grey scale conversion.

IP_EdgeDet

This module contains the methods for edge detection.

IP_ConnectedComponents

IP_ConnectedComponents performs searches for connected components. This module is mainly used by the post-processing module.

IP_Gabor

The IP_Gabor class contains all of the functionality to perform Gabor filtering on an image.

IP_Segmentation

This class is responsible for the segmentation of images that are passed to it.

A.4 The human interface classes

The main user interface forms are as follows (some others exist that perform very simple tasks):

TFmMain

This is the main form of the program. It is the first form displayed when the program is run. It is used for controlling most input and output and much of the systems operations.

TFmViewer

This form is used for performing segmentations on an image, creating input data for the genetic system and for viewing Haralick values.

TFmGeneticTestingApp

The TFmGeneticTestingApp is used to run the genetic system and set its parameters.

TFmDataCreator

This class controls the user input and output when creating test data for the genetic system.

A.5 The filing

The application supports the BMP, PGM and PPM file formats for initial input to the system. The application uses three intermediate files to store results obtained, as described in this appendix. Saving segmentation/graphical database query results in BMP format is also supported.

A.5.1 .PWN Cached image format

The purpose of this file is to pre-calculate and cache some common calculations pertaining to a particular image.

The cached file contains the original image, all Haralick features on all angles at 45^o increments at distances of 1 and 2. The file is also capable of containing Gabor filtered images, however, this has proven to be an unnecessary task for the normal use of the program. Due to the large amount of time taken to compute Gabor filtered images, the file is currently written without the pre-calculated Gabor images. These images can now be calculated via the ”tools” menu instead.

The Haralick values saved to the file are normalised and rounded off to integers on a scale of 0 to 1024. The highest and lowest Haralick values that are used to scale the values from 0 to 1024 are also saved as they are later used to determine the original values of the normalised Haralick values.

The formatting is as follows:

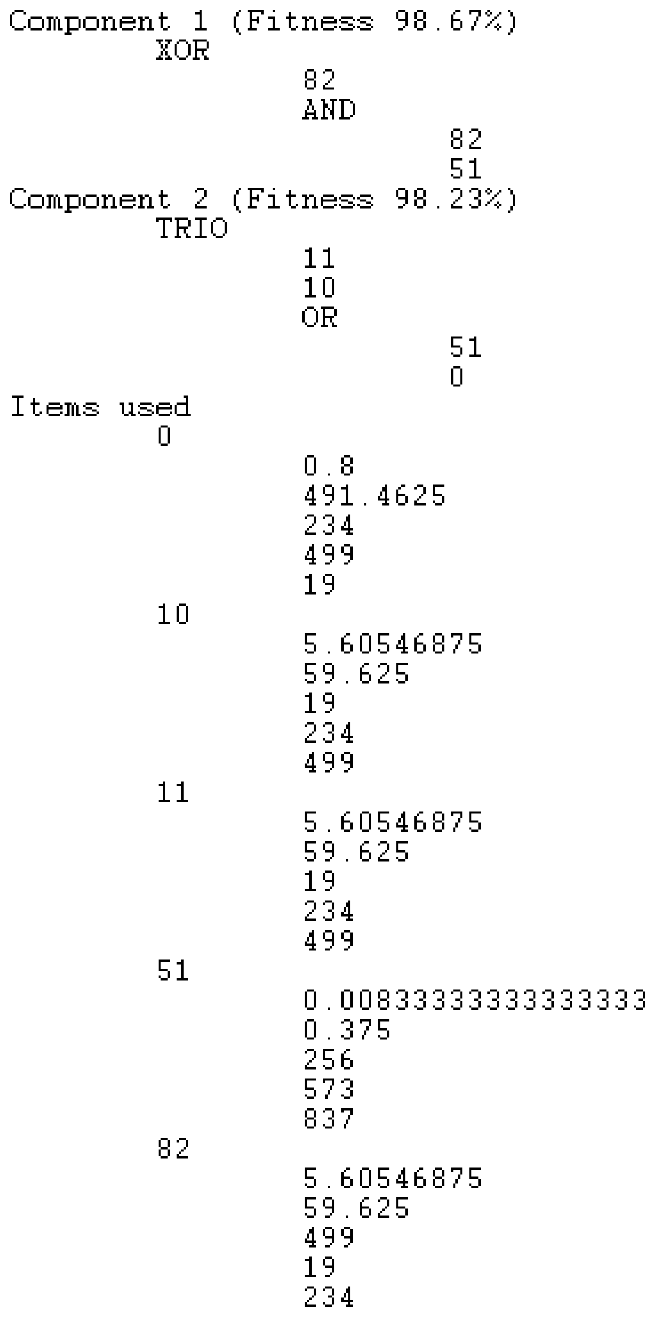
```
(WIDTH)
(HEIGHT)
(GLCO BLOCKSIZE)
(NUMBER OF IMAGES)
(GABOR MARGIN)
#ORIGINAL IMAGE IN DIRECT GREYSCALE
line by line values of image pixels
numlines = (WIDTH)*(HEIGHT)
#GABORS
#GABOR ANGLE (ANGLE)
line by line values of image pixels
numlines = (WIDTH-(GABOR MARGIN))(HEIGHT-(GABOR MARGIN))
#HARALICKS
#HARALICK (NAMEOFHARALICK) (ANGLE) (DIST)
line by line values of image pixels
numlines = (WIDTH/(GLCO BLOCKSIZE))(HEIGHT/(GLCO BLOCKSIZE))
#IDEAL SEGMENTATION
line by line values of image pixels
numlines = (WIDTH/(GLCO BLOCKSIZE))(HEIGHT/(GLCO BLOCKSIZE))
```

A.5.2 .GDT Genetic input format

The purpose of the GDT file is to provide all necessary training input to the genetic system. The GDT file is generated from the segmentation form. When it is created, the application runs through all possible triples of angle, distance and Haralick feature and performs the K-Means segmentation for that triple. The segmentations are then stored to GDT file along with the normalisation values of the Haralick features, centroid values for the segmentations and the user input training data for the genetic system.

A.5.3 .HDT Genetic algorithm format

HDT files store the genetic programs that are trained by the genetic system. Each file records the titles of each component, followed by the genetic program for extracting the component. After the extraction programs, the triples used for the extraction are listed, along with their min and max values used for normalisation (so that the genetic algorithm can apply to other images that have been normalised with different values) and the values of the three centroids for each segmentation (see fig A.4).



```
Component 1 (Fitness 98.67%)
        XOR
                82
                AND
                        82
                        51
Component 2 (Fitness 98.23%)
        TRIO
                11
                10
                OR
                        51
                        0
Items used
        0
                0.8
                491.4625
                234
                499
                19
        10
                5.60546875
                59.625
                19
                234
                499
        11
                5.60546875
                59.625
                19
                234
                499
        51
                0.00833333333333333
                0.375
                256
                573
                837
        82
                5.60546875
                59.625
                499
                19
                234
```

Figure A.4: *This is an example of the HDT file format.*

Appendix B

Comparison of results

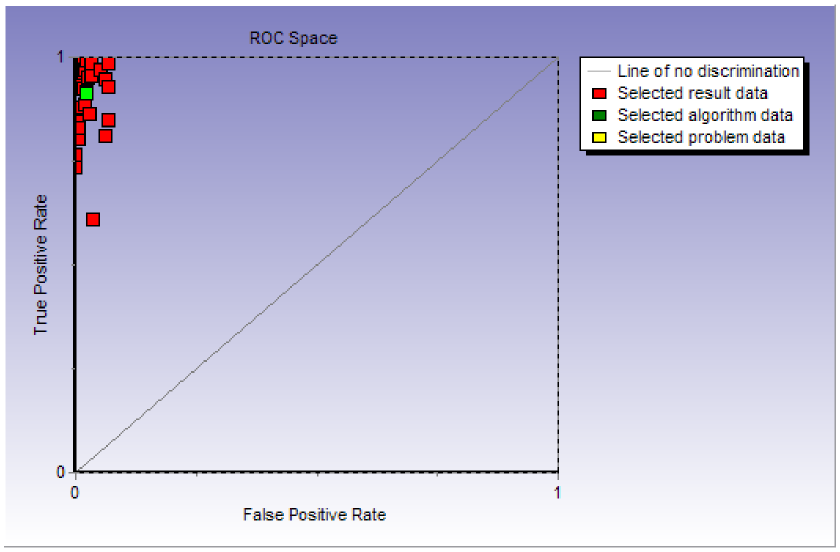


Figure B.1: *The above ROC graph illustrates the results of the genetic algorithm trained from image 527 for extracting the image segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

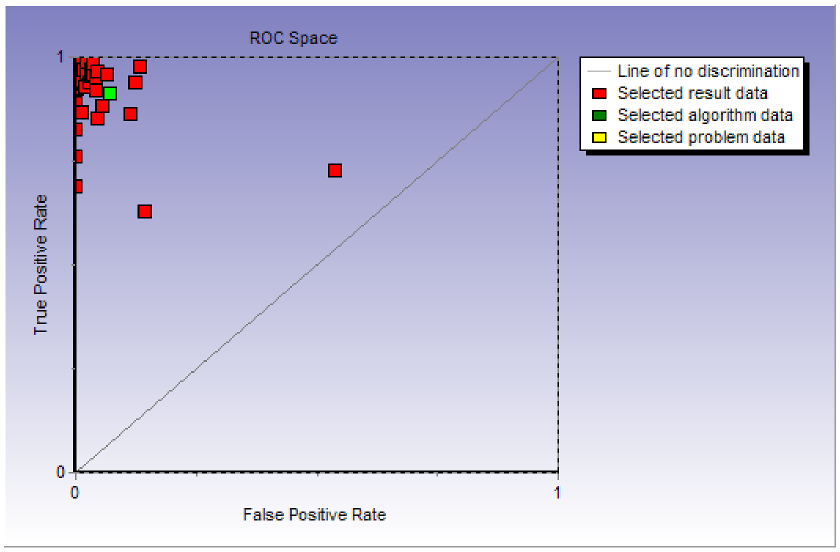


Figure B.2: *The above ROC graph illustrates the results of the genetic algorithm trained from image 531 for extracting the image segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

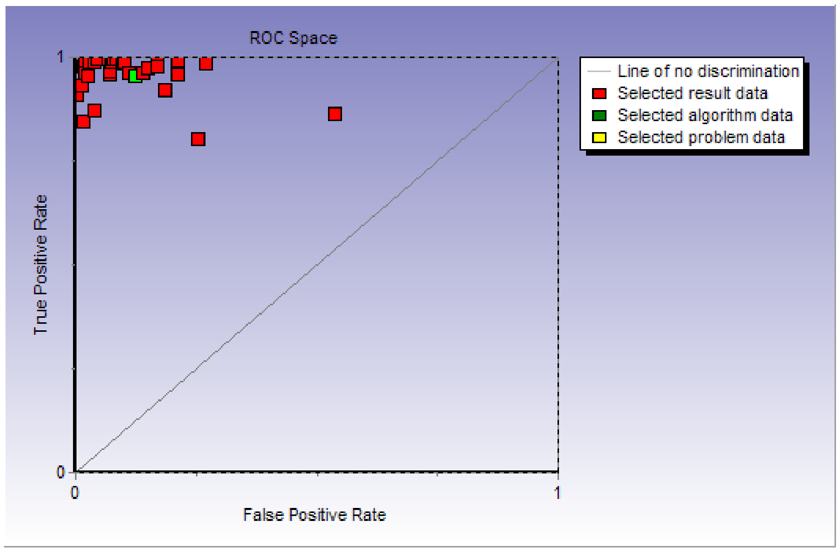


Figure B.3: *The above ROC graph illustrates the results of the genetic algorithm trained from image 545 for extracting the image segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

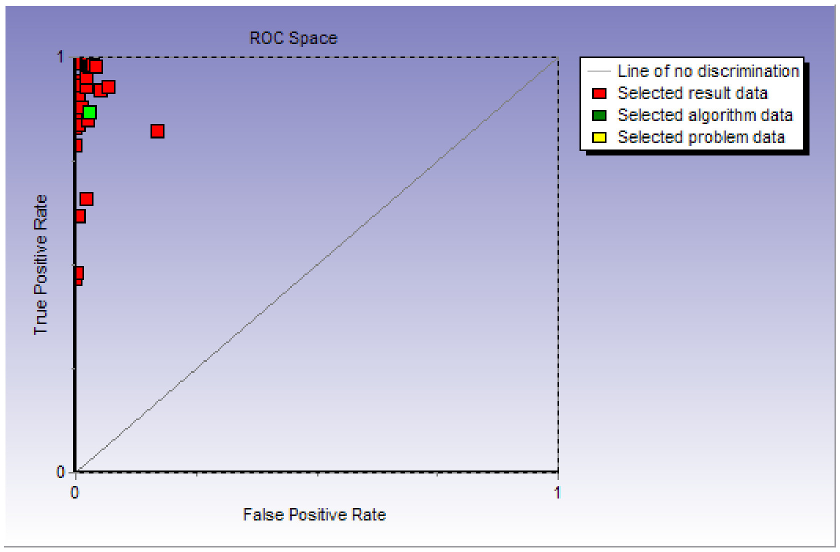


Figure B.4: *The above ROC graph illustrates the results of the genetic algorithm trained from image 839 for extracting the image segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

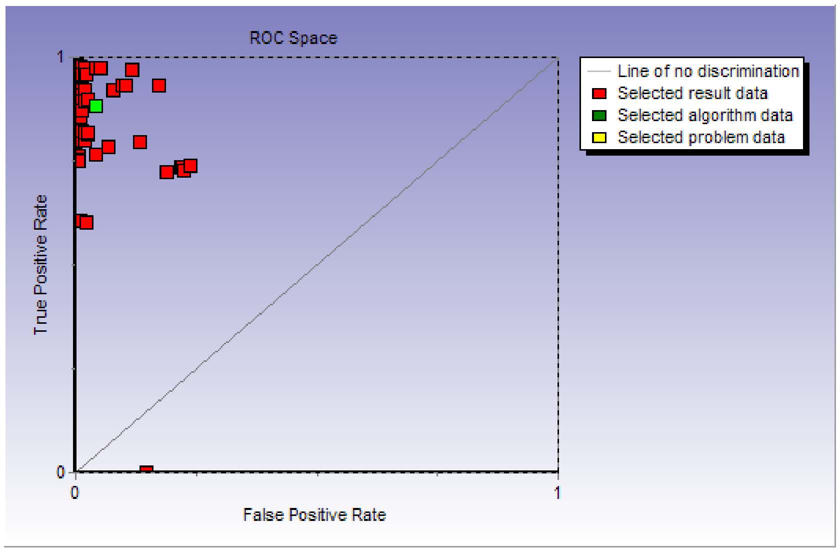


Figure B.5: *The above ROC graph illustrates the results of the post-processed K-Means algorithm for extracting the image segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

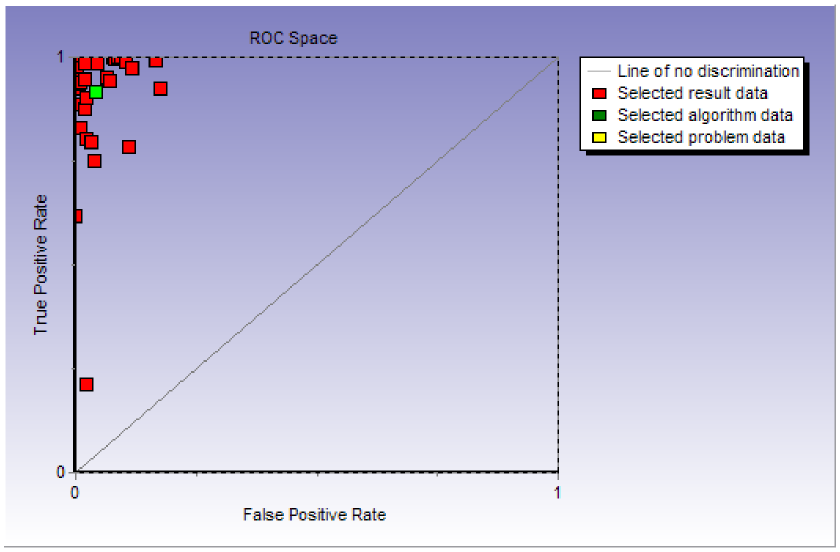


Figure B.6: *The above ROC graph illustrates the results of the genetic algorithm trained from image 527 for extracting the text segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

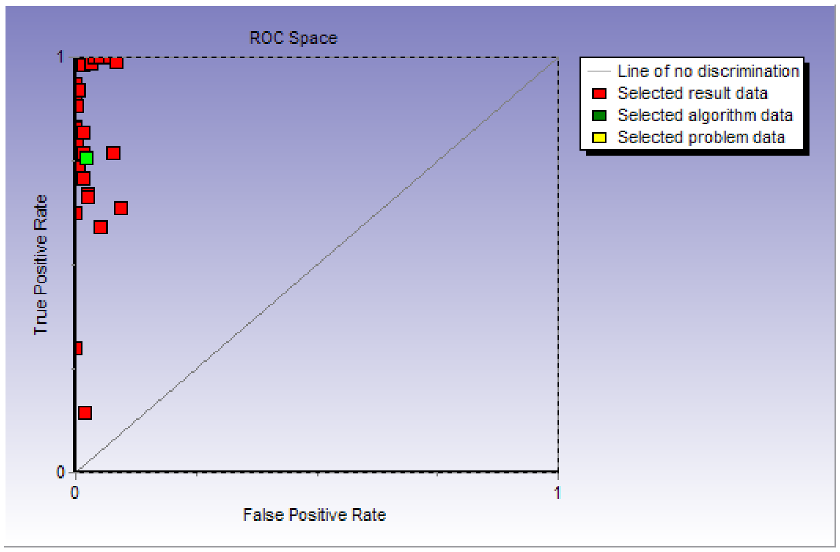


Figure B.7: *The above ROC graph illustrates the results of the genetic algorithm trained from image 531 for extracting the text segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

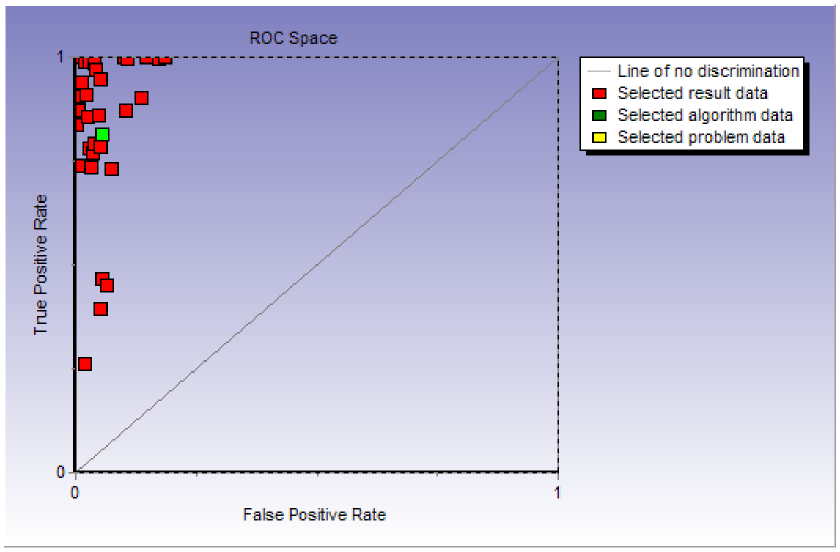


Figure B.8: *The above ROC graph illustrates the results of the genetic algorithm trained from image 545 for extracting the text segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

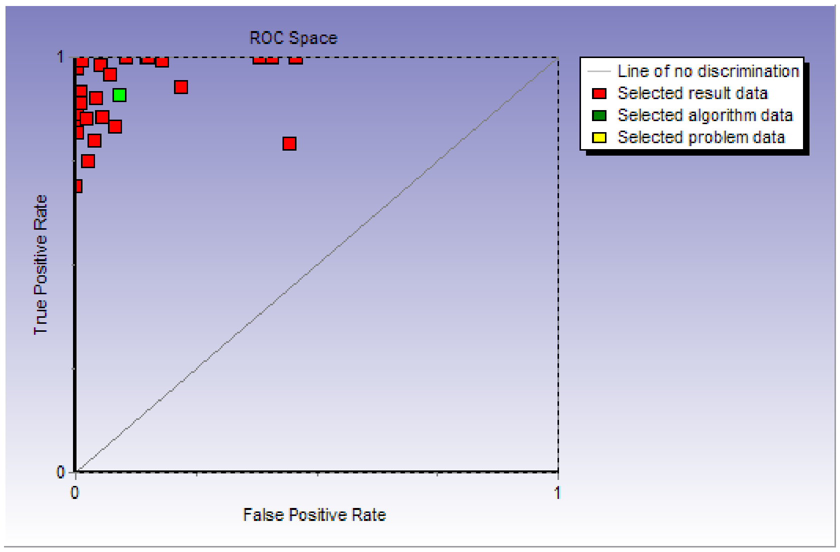


Figure B.9: *The above ROC graph illustrates the results of the genetic algorithm trained from image 839 for extracting the text segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

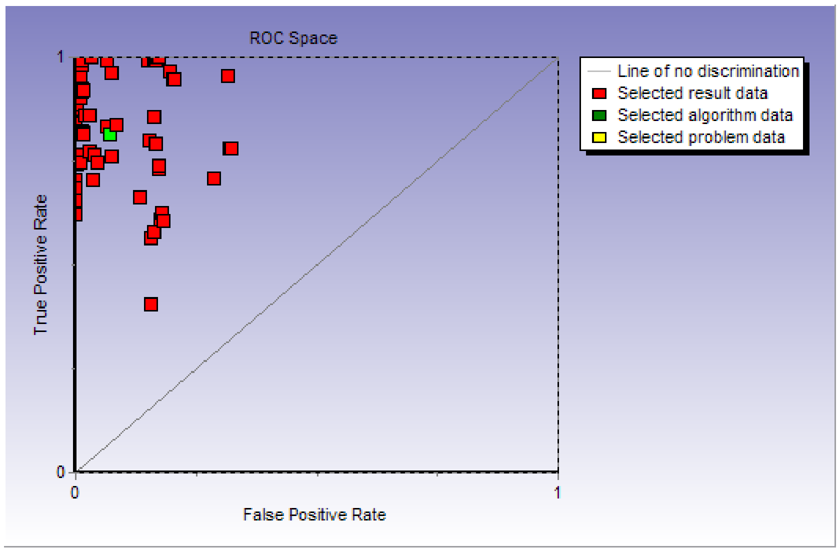
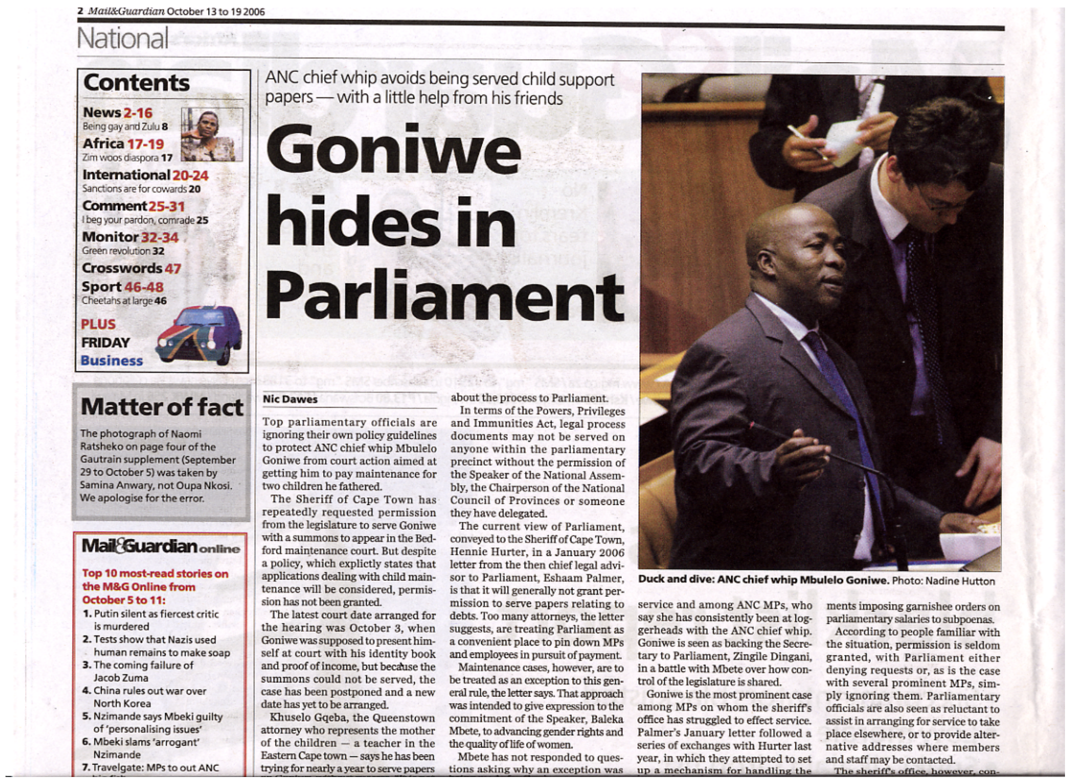


Figure B.10: *The above ROC graph illustrates the results of the post-processed K-Means algorithm for extracting the text segment. The red points represent results from individual document images, the green point represents the average result from the test data for the algorithm.*

Appendix C

Original images

The following pages contain the original images used in the example segmentations in the dissertation.



2 *Mail&Guardian* October 13 to 19 2006

National

Contents

- **News 2-16** Being gay and Zulu **8**
- **Africa 17-19** Zim woos diaspora **17**
- **International 20-24** Sanctions are for cowards **20**
- **Comment 25-31** I beg your pardon, comrade **25**
- **Monitor 32-34** Green revolution **32**
- **Crosswords 47**
- **Sport 46-48** Cheetahs at large **46**

PLUS FRIDAY Business

Matter of fact

The photograph of Naomi Ratsheko on page four of the Gautrain supplement (September 29 to October 5) was taken by Samina Anwary, not Oupa Nkosi. We apologise for the error.

Mail&Guardian online

Top 10 most-read stories on the M&G Online from October 5 to 11:

1. Putin silent as fiercest critic is murdered
2. Tests show that Nazis used human remains to make soap
3. The coming failure of Jacob Zuma
4. China rules out war over North Korea
5. Nzimande says Mbeki guilty of 'personalising issues'
6. Mbeki slams 'arrogant' Nzimande
7. Travelgate: MPs to out ANC

ANC chief whip avoids being served child support papers — with a little help from his friends

Goniwe hides in Parliament

Nic Dawes

Top parliamentary officials are ignoring their own policy guidelines to protect ANC chief whip Mbulelo Goniwe from court action aimed at getting him to pay maintenance for two children he fathered.

The Sheriff of Cape Town has repeatedly requested permission from the legislature to serve Goniwe with a summons to appear in the Bedford maintenance court. But despite a policy, which explictly states that applications dealing with child maintenance will be considered, permission has not been granted.

The latest court date arranged for the hearing was October 3, when Goniwe was supposed to present himself at court with his identity book and proof of income, but because the summons could not be served, the case has been postponed and a new date has yet to be arranged.

Khuselo Gqeba, the Queenstown attorney who represents the mother of the children — a teacher in the Eastern Cape town — says he has been trying for nearly a year to serve papers about the process to Parliament.

In terms of the Powers, Privileges and Immunities Act, legal process documents may not be served on anyone within the parliamentary precinct without the permission of the Speaker of the National Assembly, the Chairperson of the National Council of Provinces or someone they have delegated.

The current view of Parliament, conveyed to the Sheriff of Cape Town, Hennie Hurter, in a January 2006 letter from the then chief legal advisor to Parliament, Eshaam Palmer, is that it will generally not grant permission to serve papers relating to debts. Too many attorneys, the letter suggests, are treating Parliament as a convenient place to pin down MPs and employees in pursuit of payment.

Maintenance cases, however, are to be treated as an exception to this general rule, the letter says. That approach was intended to give expression to the commitment of the Speaker, Baleka Mbete, to advancing gender rights and the quality of life of women.

Mbete has not responded to questions asking why an exception was

Duck and dive: ANC chief whip Mbulelo Goniwe. Photo: Nadine Hutton

service and among ANC MPs, who say she has consistently been at loggerheads with the ANC chief whip. Goniwe is seen as backing the Secretary to Parliament, Zingile Dingani, in a battle with Mbete over how control of the legislature is shared.

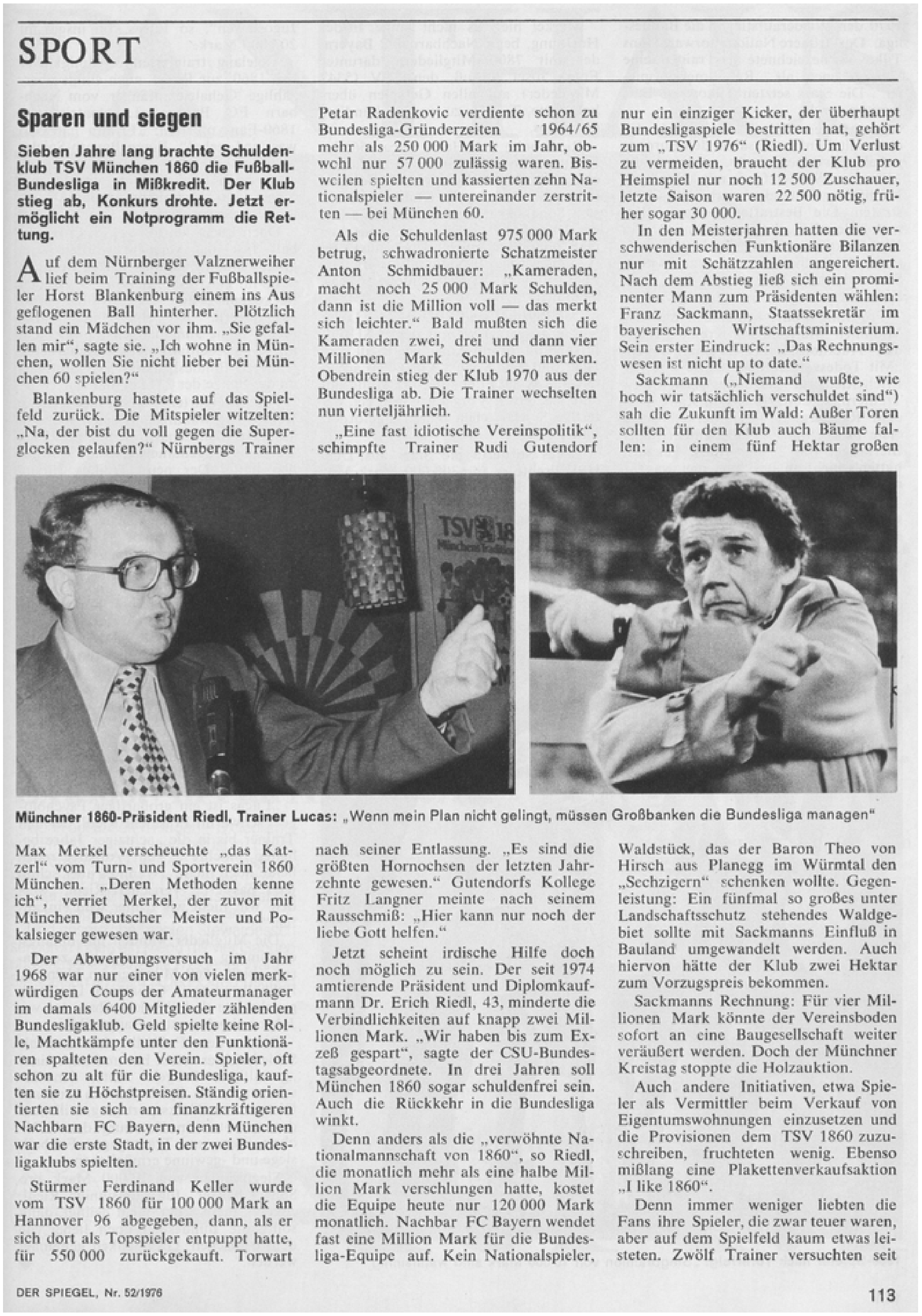
Goniwe is the most prominent case among MPs on whom the sheriff's office has struggled to effect service. Palmer's January letter followed a series of exchanges with Hurter last year, in which they attempted to set up a mechanism for handling the

ments imposing garnishee orders on parliamentary salaries to subpoenas.

According to people familiar with the situation, permission is seldom granted, with Parliament either denying requests or, as is the case with several prominent MPs, simply ignoring them. Parliamentary officials are also seen as reluctant to assist in arranging for service to take place elsewhere, or to provide alternative addresses where members and staff may be contacted.

The sheriff's office, however, con-

Figure C.1: *The image used in the post-processing demonstration images and the gabor filtering example.*



SPORT

Sparen und siegen

Sieben Jahre lang brachte Schuldenklub TSV München 1860 die Fußball-Bundesliga in Mißkredit. Der Klub stieg ab, Konkurs drohte. Jetzt ermöglicht ein Notprogramm die Rettung.

Auf dem Nürnberger Valznerweiher lief beim Training der Fußballspieler Horst Blankenburg einem ins Aus geflogenen Ball hinterher. Plötzlich stand ein Mädchen vor ihm. „Sie gefallen mir“, sagte sie. „Ich wohne in München, wollen Sie nicht lieber bei München 60 spielen?“

Blankenburg hastete auf das Spielfeld zurück. Die Mitspieler witzelten: „Na, der bist du voll gegen die Superglocken gelaufen?“ Nürnbergs Trainer Max Merkel verscheuchte „das Katzerl“ vom Turn- und Sportverein 1860 München. „Deren Methoden kenne ich“, verriet Merkel, der zuvor mit München Deutscher Meister und Pokalsieger gewesen war.

Der Abwerbungsversuch im Jahr 1968 war nur einer von vielen merkwürdigen Coups der Amateurmanager im damals 6400 Mitglieder zählenden Bundesligaklub. Geld spielte keine Rolle, Machtkämpfe unter den Funktionären spalteten den Verein. Spieler, oft schon zu alt für die Bundesliga, kauften sie zu Höchstpreisen. Ständig orientierten sie sich am finanzkräftigeren Nachbarn FC Bayern, denn München war die erste Stadt, in der zwei Bundesligaklubs spielten.

Stürmer Ferdinand Keller wurde vom TSV 1860 für 100 000 Mark an Hannover 96 abgegeben, dann, als er sich dort als Topspieler entpuppt hatte, für 550 000 zurückgekauft. Torwart Petar Radenkovic verdiente schon zu Bundesliga-Gründerzeiten 1964/65 mehr als 250 000 Mark im Jahr, obwohl nur 57 000 zulässig waren. Bisweilen spielten und kassierten zehn Nationalspieler — untereinander zerstritten — bei München 60.

Als die Schuldenlast 975 000 Mark betrug, schwadronierte Schatzmeister Anton Schmidbauer: „Kameraden, macht noch 25 000 Mark Schulden, dann ist die Million voll — das merkt sich leichter.“ Bald mußten sich die Kameraden zwei, drei und dann vier Millionen Mark Schulden merken. Obendrein stieg der Klub 1970 aus der Bundesliga ab. Die Trainer wechselten nun vierteljährlich.

„Eine fast idiotische Vereinspolitik“, schimpfte Trainer Rudi Gutendorf nach seiner Entlassung. „Es sind die größten Hornochsen der letzten Jahrzehnte gewesen.“ Gutendorfs Kollege Fritz Langner meinte nach seinem Rausschmiß: „Hier kann nur noch der liebe Gott helfen.“

Jetzt scheint irdische Hilfe doch noch möglich zu sein. Der seit 1974 amtierende Präsident und Diplomkaufmann Dr. Erich Riedl, 43, minderte die Verbindlichkeiten auf knapp zwei Millionen Mark. „Wir haben bis zum Exzeß gespart“, sagte der CSU-Bundestagsabgeordnete. In drei Jahren soll München 1860 sogar schuldenfrei sein. Auch die Rückkehr in die Bundesliga winkt.

Denn anders als die „verwöhnte Nationalmannschaft von 1860“, so Riedl, die monatlich mehr als eine halbe Millien Mark verschlungen hatte, kostet die Equipe heute nur 120 000 Mark monatlich. Nachbar FC Bayern wendet fast eine Million Mark für die Bundesliga-Equipe auf. Kein Nationalspieler, nur ein einziger Kicker, der überhaupt Bundesligaspiele bestritten hat, gehört zum „TSV 1976“ (Riedl). Um Verlust zu vermeiden, braucht der Klub pro Heimspiel nur noch 12 500 Zuschauer, letzte Saison waren 22 500 nötig, früher sogar 30 000.

In den Meisterjahren hatten die verschwenderischen Funktionäre Bilanzen nur mit Schätzzahlen angereichert. Nach dem Abstieg ließ sich ein prominenter Mann zum Präsidenten wählen: Franz Sackmann, Staatssekretär im bayerischen Wirtschaftsministerium. Sein erster Eindruck: „Das Rechnungswesen ist nicht up to date.“

Sackmann („Niemand wußte, wie hoch wir tatsächlich verschuldet sind“) sah die Zukunft im Wald: Außer Toren sollten für den Klub auch Bäume fallen: in einem fünf Hektar großen Waldstück, das der Baron Theo von Hirsch aus Planegg im Würmtal den „Sechzigern“ schenken wollte. Gegenleistung: Ein fünfmal so großes unter Landschaftsschutz stehendes Waldgebiet sollte mit Sackmanns Einfluß in Bauland umgewandelt werden. Auch hiervon hätte der Klub zwei Hektar zum Vorzugspreis bekommen.

Sackmanns Rechnung: Für vier Millionen Mark könnte der Vereinsboden sofort an eine Baugesellschaft weiter veräußert werden. Doch der Münchner Kreistag stoppte die Holzauktion.

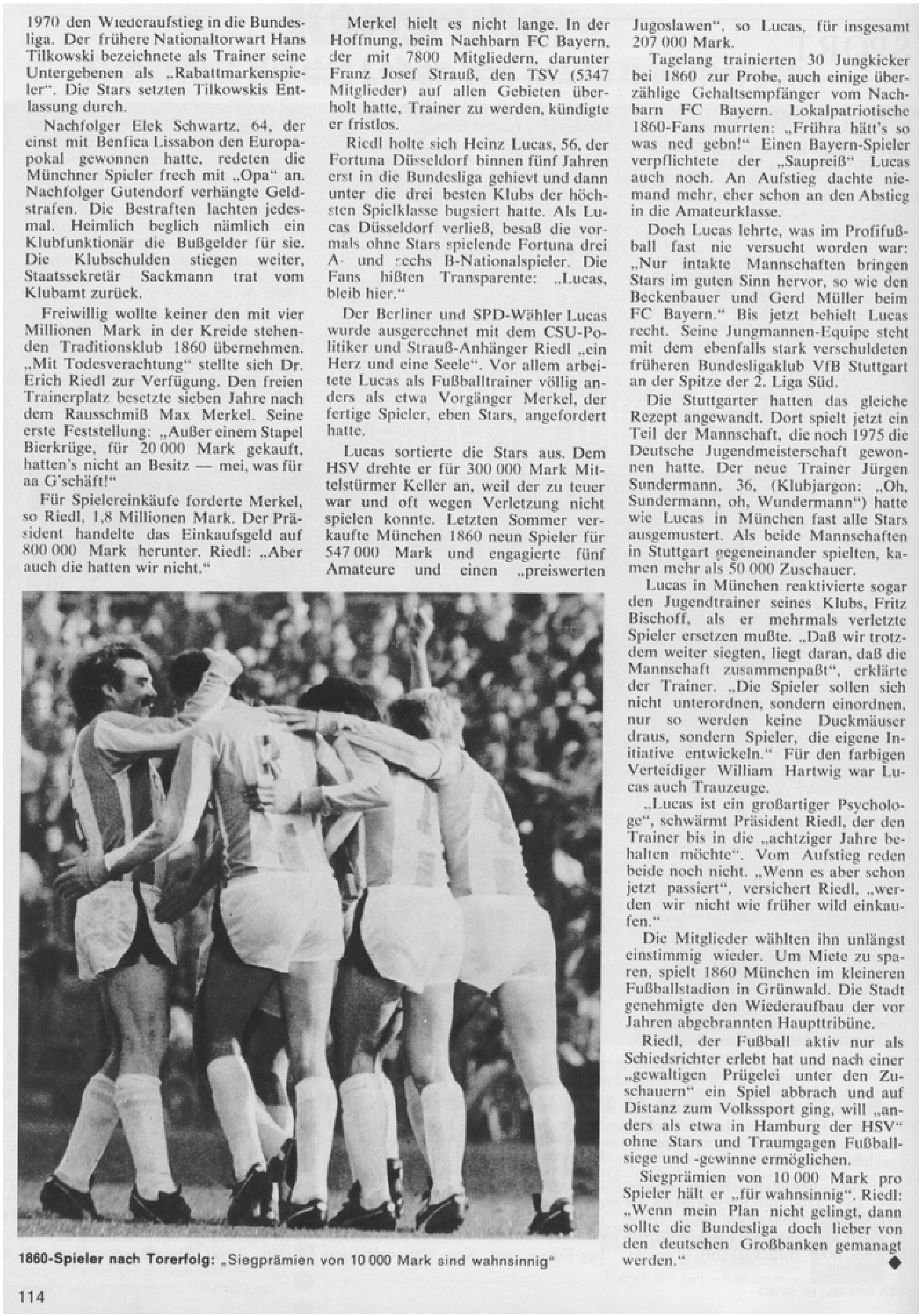
Auch andere Initiativen, etwa Spieler als Vermittler beim Verkauf von Eigentumswohnungen einzusetzen und die Provisionen dem TSV 1860 zuzuschreiben, fruchteten wenig. Ebenso mißlang eine Plakettenverkaufsaktion „I like 1860“.

Denn immer weniger liebten die Fans ihre Spieler, die zwar teuer waren, aber auf dem Spielfeld kaum etwas leisteten. Zwölf Trainer versuchten seit

Münchner 1860-Präsident Riedl, Trainer Lucas: „Wenn mein Plan nicht gelingt, müssen Großbanken die Bundesliga managen“

DER SPIEGEL, Nr. 52/1976 113

Figure C.2: *The original image 527 used in the example segmentations and for training.*



1970 den Wiederaufstieg in die Bundesliga. Der frühere Nationaltorwart Hans Tilkowski bezeichnete als Trainer seine Untergebenen als „Rabattmarkenspieler“. Die Stars setzten Tilkowskis Entlassung durch.

Nachfolger Elek Schwartz, 64, der einst mit Benfica Lissabon den Europapokal gewonnen hatte, redeten die Münchner Spieler frech mit „Opa“ an. Nachfolger Gutendorf verhängte Geldstrafen. Die Bestraften lachten jedesmal. Heimlich beglich nämlich ein Klubfunktionär die Bußgelder für sie. Die Klubschulden stiegen weiter, Staatssekretär Sackmann trat vom Klubamt zurück.

Freiwillig wollte keiner den mit vier Millionen Mark in der Kreide stehenden Traditionsklub 1860 übernehmen. „Mit Todesverachtung“ stellte sich Dr. Erich Riedl zur Verfügung. Den freien Trainerplatz besetzte sieben Jahre nach dem Rausschmiß Max Merkel. Seine erste Feststellung: „Außer einem Stapel Bierkrüge, für 20 000 Mark gekauft, hatten's nicht an Besitz — mei, was für aa G'schäft!“

Für Spielereinkäufe forderte Merkel, so Riedl, 1,8 Millionen Mark. Der Präsident handelte das Einkaufsgeld auf 800 000 Mark herunter. Riedl: „Aber auch die hatten wir nicht.“

Merkel hielt es nicht lange. In der Hoffnung, beim Nachbarn FC Bayern, der mit 7800 Mitgliedern, darunter Franz Josef Strauß, den TSV (5347 Mitglieder) auf allen Gebieten überholt hatte, Trainer zu werden, kündigte er fristlos.

Riedl holte sich Heinz Lucas, 56, der Fortuna Düsseldorf binnen fünf Jahren erst in die Bundesliga gehievt und dann unter die drei besten Klubs der höchsten Spielklasse bugsiert hatte. Als Lucas Düsseldorf verließ, besaß die vormals ohne Stars spielende Fortuna drei A- und sechs B-Nationalspieler. Die Fans hißten Transparente: „Lucas, bleib hier.“

Der Berliner und SPD-Wähler Lucas wurde ausgerechnet mit dem CSU-Politiker und Strauß-Anhänger Riedl „ein Herz und eine Seele“. Vor allem arbeitete Lucas als Fußballtrainer völlig anders als etwa Vorgänger Merkel, der fertige Spieler, eben Stars, angefordert hatte.

Lucas sortierte die Stars aus. Dem HSV drehte er für 300 000 Mark Mittelstürmer Keller an, weil der zu teuer war und oft wegen Verletzung nicht spielen konnte. Letzten Sommer verkaufte München 1860 neun Spieler für 547 000 Mark und engagierte fünf Amateure und einen „preiswerten Jugoslawen“, so Lucas, für insgesamt 207 000 Mark.

Tagelang trainierten 30 Jungkicker bei 1860 zur Probe, auch einige überzählige Gehaltsempfänger vom Nachbarn FC Bayern. Lokalpatriotische 1860-Fans murrten: „Frühra hätt's so was ned gebn!“ Einen Bayern-Spieler verpflichtete der „Saupreiß“ Lucas auch noch. An Aufstieg dachte niemand mehr, eher schon an den Abstieg in die Amateurklasse.

Doch Lucas lehrte, was im Profifußball fast nie versucht worden war: „Nur intakte Mannschaften bringen Stars im guten Sinn hervor, so wie den Beckenbauer und Gerd Müller beim FC Bayern.“ Bis jetzt behielt Lucas recht. Seine Jungmannen-Equipe steht mit dem ebenfalls stark verschuldeten früheren Bundesligaklub VfB Stuttgart an der Spitze der 2. Liga Süd.

Die Stuttgarter hatten das gleiche Rezept angewandt. Dort spielt jetzt ein Teil der Mannschaft, die noch 1975 die Deutsche Jugendmeisterschaft gewonnen hatte. Der neue Trainer Jürgen Sundermann, 36, (Klubjargon: „Oh, Sundermann, oh, Wundermann“) hatte wie Lucas in München fast alle Stars ausgemustert. Als beide Mannschaften in Stuttgart gegeneinander spielten, kamen mehr als 50 000 Zuschauer.

Lucas in München reaktivierte sogar den Jugendtrainer seines Klubs, Fritz Bischoff, als er mehrmals verletzte Spieler ersetzen mußte. „Daß wir trotzdem weiter siegten, liegt daran, daß die Mannschaft zusammenpaßt“, erklärte der Trainer. „Die Spieler sollen sich nicht unterordnen, sondern einordnen, nur so werden keine Duckmäuser draus, sondern Spieler, die eigene Initiative entwickeln.“ Für den farbigen Verteidiger William Hartwig war Lucas auch Trauzeuge.

„Lucas ist ein großartiger Psychologe“, schwärmt Präsident Riedl, der den Trainer bis in die „achtziger Jahre behalten möchte“. Vom Aufstieg reden beide noch nicht. „Wenn es aber schon jetzt passiert“, versichert Riedl, „werden wir nicht wie früher wild einkaufen.“

Die Mitglieder wählten ihn unlängst einstimmig wieder. Um Miete zu sparen, spielt 1860 München im kleineren Fußballstadion in Grünwald. Die Stadt genehmigte den Wiederaufbau der vor Jahren abgebrannten Haupttribüne.

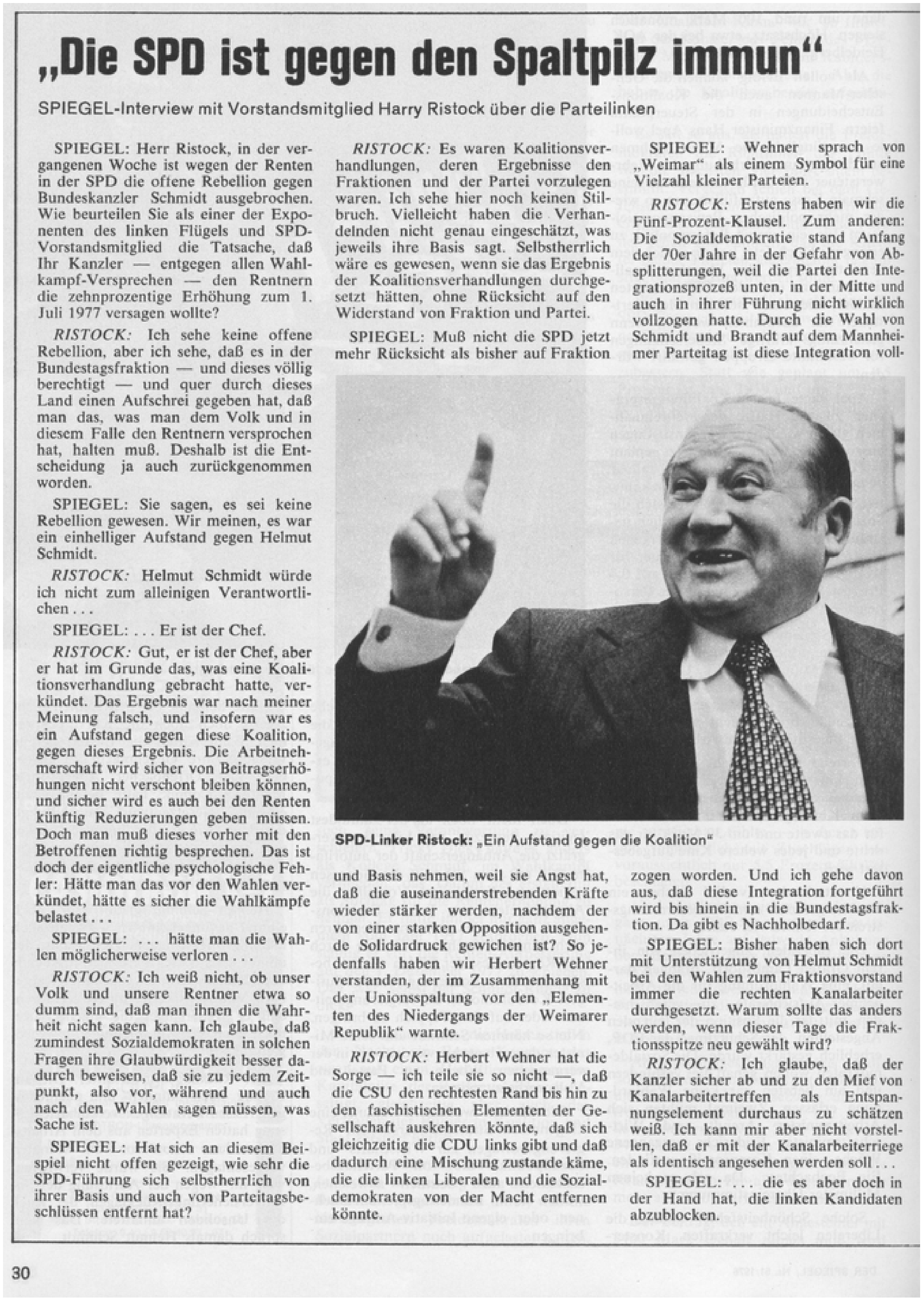
Riedl, der Fußball aktiv nur als Schiedsrichter erlebt hat und nach einer „gewaltigen Prügelei unter den Zuschauern“ ein Spiel abbrach und auf Distanz zum Volkssport ging, will „anders als etwa in Hamburg der HSV“ ohne Stars und Traumgagen Fußballsiege und -gewinne ermöglichen.

Siegprämien von 10 000 Mark pro Spieler hält er „für wahnsinnig“. Riedl: „Wenn mein Plan nicht gelingt, dann sollte die Bundesliga doch lieber von den deutschen Großbanken gemanagt werden.“ ◆

1860-Spieler nach Torerfolg: „Siegprämien von 10 000 Mark sind wahnsinnig“

114

Figure C.3: *The original image 528 used in the example segmentations.*



„Die SPD ist gegen den Spaltpilz immun“

SPIEGEL-Interview mit Vorstandsmitglied Harry Ristock über die Parteilinken

SPIEGEL: Herr Ristock, in der vergangenen Woche ist wegen der Renten in der SPD die offene Rebellion gegen Bundeskanzler Schmidt ausgebrochen. Wie beurteilen Sie als einer der Exponenten des linken Flügels und SPD-Vorstandsmitglied die Tatsache, daß Ihr Kanzler — entgegen allen Wahlkampf-Versprechen — den Rentnern die zehnprozentige Erhöhung zum 1. Juli 1977 versagen wollte?

RISTOCK: Ich sehe keine offene Rebellion, aber ich sehe, daß es in der Bundestagsfraktion — und dieses völlig berechtigt — und quer durch dieses Land einen Aufschrei gegeben hat, daß man das, was man dem Volk und in diesem Falle den Rentnern versprochen hat, halten muß. Deshalb ist die Entscheidung ja auch zurückgenommen worden.

SPIEGEL: Sie sagen, es sei keine Rebellion gewesen. Wir meinen, es war ein einhelliger Aufstand gegen Helmut Schmidt.

RISTOCK: Helmut Schmidt würde ich nicht zum alleinigen Verantwortlichen ...

SPIEGEL: ... Er ist der Chef.

RISTOCK: Gut, er ist der Chef, aber er hat im Grunde das, was eine Koalitionsverhandlung gebracht hatte, verkündet. Das Ergebnis war nach meiner Meinung falsch, und insofern war es ein Aufstand gegen diese Koalition, gegen dieses Ergebnis. Die Arbeitnehmerschaft wird sicher von Beitragserhöhungen nicht verschont bleiben können, und sicher wird es auch bei den Renten künftig Reduzierungen geben müssen. Doch man muß dieses vorher mit den Betroffenen richtig besprechen. Das ist doch der eigentliche psychologische Fehler: Hätte man das vor den Wahlen verkündet, hätte es sicher die Wahlkämpfe belastet ...

SPIEGEL: ... hätte man die Wahlen möglicherweise verloren ...

RISTOCK: Ich weiß nicht, ob unser Volk und unsere Rentner etwa so dumm sind, daß man ihnen die Wahrheit nicht sagen kann. Ich glaube, daß zumindest Sozialdemokraten in solchen Fragen ihre Glaubwürdigkeit besser dadurch beweisen, daß sie zu jedem Zeitpunkt, also vor, während und auch nach den Wahlen sagen müssen, was Sache ist.

SPIEGEL: Hat sich an diesem Beispiel nicht offen gezeigt, wie sehr die SPD-Führung sich selbstherrlich von ihrer Basis und auch von Parteitagsbeschlüssen entfernt hat?

RISTOCK: Es waren Koalitionsverhandlungen, deren Ergebnisse den Fraktionen und der Partei vorzulegen waren. Ich sehe hier noch keinen Stilbruch. Vielleicht haben die Verhandelnden nicht genau eingeschätzt, was jeweils ihre Basis sagt. Selbstherrlich wäre es gewesen, wenn sie das Ergebnis der Koalitionsverhandlungen durchgesetzt hätten, ohne Rücksicht auf den Widerstand von Fraktion und Partei.

SPIEGEL: Muß nicht die SPD jetzt mehr Rücksicht als bisher auf Fraktion und Basis nehmen, weil sie Angst hat, daß die auseinanderstrebenden Kräfte wieder stärker werden, nachdem der von einer starken Opposition ausgehende Solidardruck gewichen ist? So jedenfalls haben wir Herbert Wehner verstanden, der im Zusammenhang mit der Unionsspaltung vor den „Elementen des Niedergangs der Weimarer Republik“ warnte.

RISTOCK: Herbert Wehner hat die Sorge — ich teile sie so nicht —, daß die CSU den rechtesten Rand bis hin zu den faschistischen Elementen der Gesellschaft auskehren könnte, daß sich gleichzeitig die CDU links gibt und daß dadurch eine Mischung zustande käme, die die linken Liberalen und die Sozialdemokraten von der Macht entfernen könnte.

SPIEGEL: Wehner sprach von „Weimar“ als einem Symbol für eine Vielzahl kleiner Parteien.

RISTOCK: Erstens haben wir die Fünf-Prozent-Klausel. Zum anderen: Die Sozialdemokratie stand Anfang der 70er Jahre in der Gefahr von Absplitterungen, weil die Partei den Integrationsprozeß unten, in der Mitte und auch in ihrer Führung nicht wirklich vollzogen hatte. Durch die Wahl von Schmidt und Brandt auf dem Mannheimer Parteitag ist diese Integration vollzogen worden. Und ich gehe davon aus, daß diese Integration fortgeführt wird bis hinein in die Bundestagsfraktion. Da gibt es Nachholbedarf.

SPIEGEL: Bisher haben sich dort mit Unterstützung von Helmut Schmidt bei den Wahlen zum Fraktionsvorstand immer die rechten Kanalarbeiter durchgesetzt. Warum sollte das anders werden, wenn dieser Tage die Fraktionsspitze neu gewählt wird?

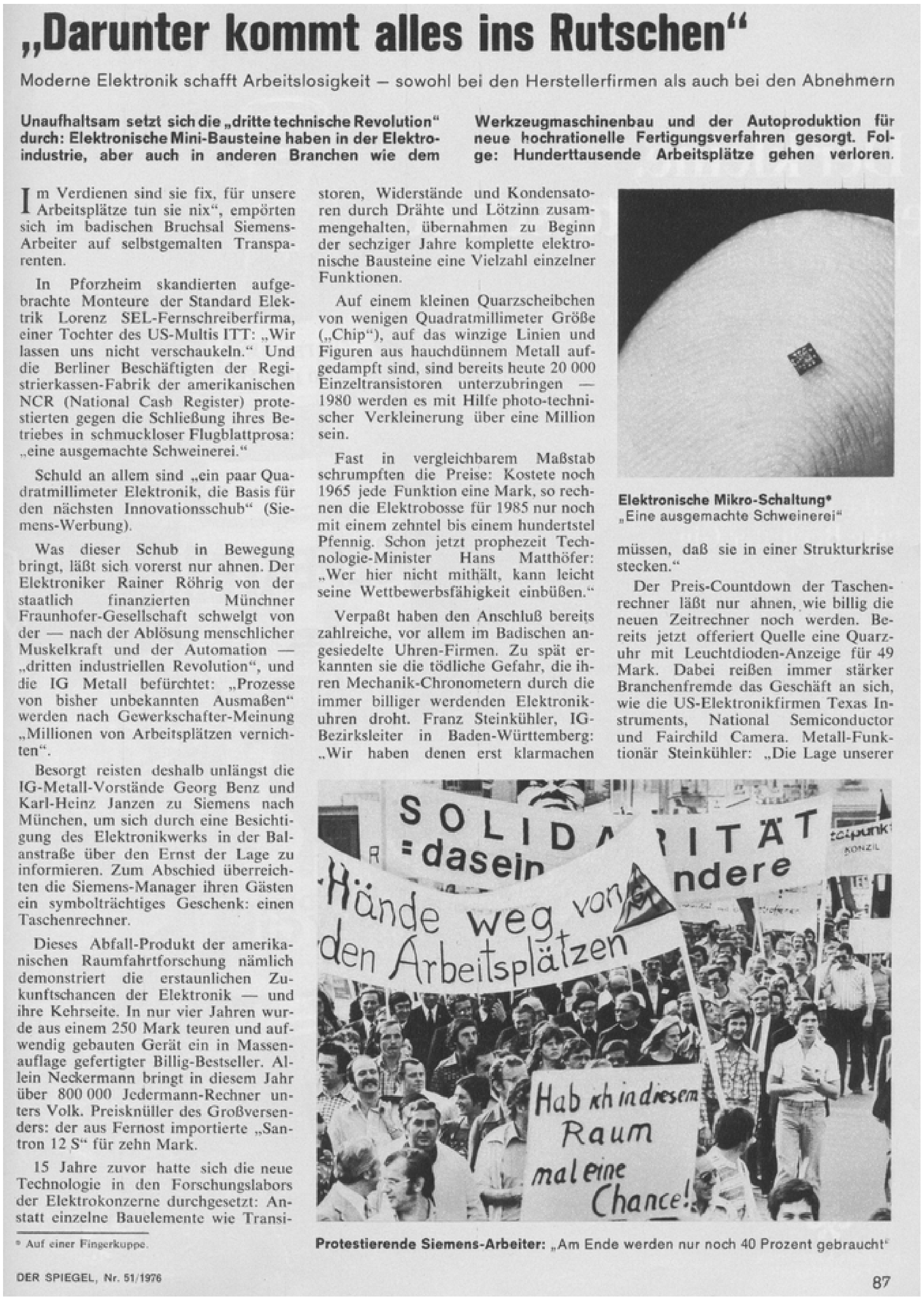
RISTOCK: Ich glaube, daß der Kanzler sicher ab und zu den Mief von Kanalarbeitertreffen als Entspannungselement durchaus zu schätzen weiß. Ich kann mir aber nicht vorstellen, daß er mit der Kanalarbeiterriege als identisch angesehen werden soll ...

SPIEGEL: ... die es aber doch in der Hand hat, die linken Kandidaten abzublocken.

SPD-Linker Ristock: „Ein Aufstand gegen die Koalition“

30

Figure C.4: *The original image 531 used in the example segmentations.*



„Darunter kommt alles ins Rutschen"

Moderne Elektronik schafft Arbeitslosigkeit — sowohl bei den Herstellerfirmen als auch bei den Abnehmern

Unaufhaltsam setzt sich die „dritte technische Revolution" durch: Elektronische Mini-Bausteine haben in der Elektroindustrie, aber auch in anderen Branchen wie dem Werkzeugmaschinenbau und der Autoproduktion für neue hochrationelle Fertigungsverfahren gesorgt. Folge: Hunderttausende Arbeitsplätze gehen verloren.

Im Verdienen sind sie fix, für unsere Arbeitsplätze tun sie nix", empörten sich im badischen Bruchsal Siemens-Arbeiter auf selbstgemalten Transparenten.

In Pforzheim skandierten aufgebrachte Monteure der Standard Elektrik Lorenz SEL-Fernschreiberfirma, einer Tochter des US-Multis ITT: „Wir lassen uns nicht verschaukeln." Und die Berliner Beschäftigten der Registrierkassen-Fabrik der amerikanischen NCR (National Cash Register) protestierten gegen die Schließung ihres Betriebes in schmuckloser Flugblattprosa: „eine ausgemachte Schweinerei."

Schuld an allem sind „ein paar Quadratmillimeter Elektronik, die Basis für den nächsten Innovationsschub" (Siemens-Werbung).

Was dieser Schub in Bewegung bringt, läßt sich vorerst nur ahnen. Der Elektroniker Rainer Röhrig von der staatlich finanzierten Münchner Fraunhofer-Gesellschaft schwelgt von der — nach der Ablösung menschlicher Muskelkraft und der Automation — „dritten industriellen Revolution", und die IG Metall befürchtet: „Prozesse von bisher unbekannten Ausmaßen" werden nach Gewerkschafter-Meinung „Millionen von Arbeitsplätzen vernichten".

Besorgt reisten deshalb unlängst die IG-Metall-Vorstände Georg Benz und Karl-Heinz Janzen zu Siemens nach München, um sich durch eine Besichtigung des Elektronikwerks in der Balanstraße über den Ernst der Lage zu informieren. Zum Abschied überreichten die Siemens-Manager ihren Gästen ein symbolträchtiges Geschenk: einen Taschenrechner.

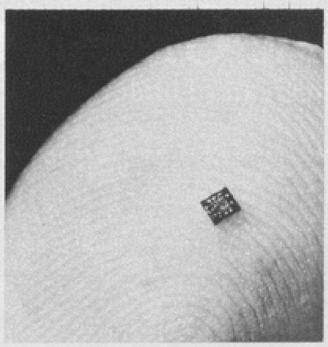
Dieses Abfall-Produkt der amerikanischen Raumfahrtforschung nämlich demonstriert die erstaunlichen Zukunftschancen der Elektronik — und ihre Kehrseite. In nur vier Jahren wurde aus einem 250 Mark teuren und aufwendig gebauten Gerät ein in Massenauflage gefertigter Billig-Bestseller. Allein Neckermann bringt in diesem Jahr über 800 000 Jedermann-Rechner unters Volk. Preisknüller des Großversenders: der aus Fernost importierte „Santron 12 S" für zehn Mark.

15 Jahre zuvor hatte sich die neue Technologie in den Forschungslabors der Elektrokonzerne durchgesetzt: Anstatt einzelne Bauelemente wie Transistoren, Widerstände und Kondensatoren durch Drähte und Lötzinn zusammengehalten, übernahmen zu Beginn der sechziger Jahre komplette elektronische Bausteine eine Vielzahl einzelner Funktionen.

Auf einem kleinen Quarzscheibchen von wenigen Quadratmillimeter Größe („Chip"), auf das winzige Linien und Figuren aus hauchdünnem Metall aufgedampft sind, sind bereits heute 20 000 Einzeltransistoren unterzubringen — 1980 werden es mit Hilfe photo-technischer Verkleinerung über eine Million sein.

Fast in vergleichbarem Maßstab schrumpften die Preise: Kostete noch 1965 jede Funktion eine Mark, so rechnen die Elektrobosse für 1985 nur noch mit einem zehntel bis einem hundertstel Pfennig. Schon jetzt prophezeit Technologie-Minister Hans Matthöfer: „Wer hier nicht mithält, kann leicht seine Wettbewerbsfähigkeit einbüßen."

Verpaßt haben den Anschluß bereits zahlreiche, vor allem im Badischen angesiedelte Uhren-Firmen. Zu spät erkannten sie die tödliche Gefahr, die ihren Mechanik-Chronometern durch die immer billiger werdenden Elektronikuhren droht. Franz Steinkühler, IG-Bezirksleiter in Baden-Württemberg: „Wir haben denen erst klarmachen müssen, daß sie in einer Strukturkrise stecken."



Elektronische Mikro-Schaltung*
„Eine ausgemachte Schweinerei"

Der Preis-Countdown der Taschenrechner läßt nur ahnen, wie billig die neuen Zeitrechner noch werden. Bereits jetzt offeriert Quelle eine Quarzuhr mit Leuchtdioden-Anzeige für 49 Mark. Dabei reißen immer stärker Branchenfremde das Geschäft an sich, wie die US-Elektronikfirmen Texas Instruments, National Semiconductor und Fairchild Camera. Metall-Funktionär Steinkühler: „Die Lage unserer

SOLIDARITÄT
= dasein ... andere
Hände weg von den Arbeitsplätzen
Hab ich in diesem Raum mal eine Chance!

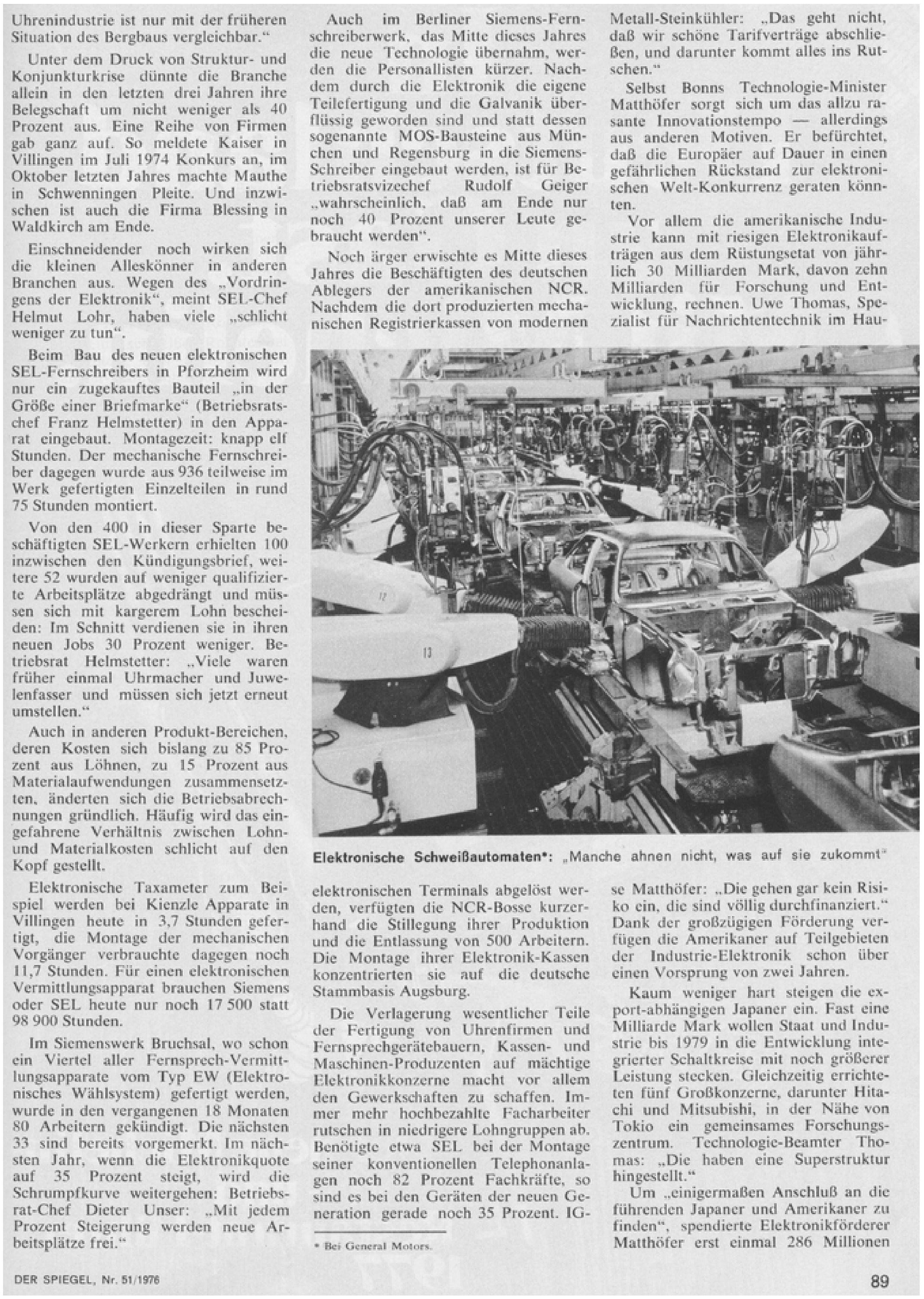
Protestierende Siemens-Arbeiter: „Am Ende werden nur noch 40 Prozent gebraucht"

* Auf einer Fingerkuppe.

DER SPIEGEL, Nr. 51/1976

87

Figure C.5: *The original image 533 used in the example segmentations.*



Uhrenindustrie ist nur mit der früheren Situation des Bergbaus vergleichbar.“

Unter dem Druck von Struktur- und Konjunkturkrise dünnte die Branche allein in den letzten drei Jahren ihre Belegschaft um nicht weniger als 40 Prozent aus. Eine Reihe von Firmen gab ganz auf. So meldete Kaiser in Villingen im Juli 1974 Konkurs an, im Oktober letzten Jahres machte Mauthe in Schwenningen Pleite. Und inzwischen ist auch die Firma Blessing in Waldkirch am Ende.

Einschneidender noch wirken sich die kleinen Alleskönner in anderen Branchen aus. Wegen des „Vordringens der Elektronik“, meint SEL-Chef Helmut Lohr, haben viele „schlicht weniger zu tun“.

Beim Bau des neuen elektronischen SEL-Fernschreibers in Pforzheim wird nur ein zugekauftes Bauteil „in der Größe einer Briefmarke“ (Betriebsratschef Franz Helmstetter) in den Apparat eingebaut. Montagezeit: knapp elf Stunden. Der mechanische Fernschreiber dagegen wurde aus 936 teilweise im Werk gefertigten Einzelteilen in rund 75 Stunden montiert.

Von den 400 in dieser Sparte beschäftigten SEL-Werkern erhielten 100 inzwischen den Kündigungsbrief, weitere 52 wurden auf weniger qualifizierte Arbeitsplätze abgedrängt und müssen sich mit kargerem Lohn bescheiden: Im Schnitt verdienen sie in ihren neuen Jobs 30 Prozent weniger. Betriebsrat Helmstetter: „Viele waren früher einmal Uhrmacher und Juwelenfasser und müssen sich jetzt erneut umstellen.“

Auch in anderen Produkt-Bereichen, deren Kosten sich bislang zu 85 Prozent aus Löhnen, zu 15 Prozent aus Materialaufwendungen zusammensetzten, änderten sich die Betriebsabrechnungen gründlich. Häufig wird das eingefahrene Verhältnis zwischen Lohn- und Materialkosten schlicht auf den Kopf gestellt.

Elektronische Taxameter zum Beispiel werden bei Kienzle Apparate in Villingen heute in 3,7 Stunden gefertigt, die Montage der mechanischen Vorgänger verbrauchte dagegen noch 11,7 Stunden. Für einen elektronischen Vermittlungsapparat brauchen Siemens oder SEL heute nur noch 17 500 statt 98 900 Stunden.

Im Siemenswerk Bruchsal, wo schon ein Viertel aller Fernsprech-Vermittlungsapparate vom Typ EW (Elektronisches Wählsystem) gefertigt werden, wurde in den vergangenen 18 Monaten 80 Arbeitern gekündigt. Die nächsten 33 sind bereits vorgemerkt. Im nächsten Jahr, wenn die Elektronikquote auf 35 Prozent steigt, wird die Schrumpfkurve weitergehen: Betriebsrat-Chef Dieter Unser: „Mit jedem Prozent Steigerung werden neue Arbeitsplätze frei.“

Auch im Berliner Siemens-Fernschreiberwerk, das Mitte dieses Jahres die neue Technologie übernahm, werden die Personallisten kürzer. Nachdem durch die Elektronik die eigene Teilefertigung und die Galvanik überflüssig geworden sind und statt dessen sogenannte MOS-Bausteine aus München und Regensburg in die Siemens-Schreiber eingebaut werden, ist für Betriebsratsvizechef Rudolf Geiger „wahrscheinlich, daß am Ende nur noch 40 Prozent unserer Leute gebraucht werden“.

Noch ärger erwischte es Mitte dieses Jahres die Beschäftigten des deutschen Ablegers der amerikanischen NCR. Nachdem die dort produzierten mechanischen Registrierkassen von modernen elektronischen Terminals abgelöst werden, verfügten die NCR-Bosse kurzerhand die Stillegung ihrer Produktion und die Entlassung von 500 Arbeitern. Die Montage ihrer Elektronik-Kassen konzentrierten sie auf die deutsche Stammbasis Augsburg.

Die Verlagerung wesentlicher Teile der Fertigung von Uhrenfirmen und Fernsprechgerätebauern, Kassen- und Maschinen-Produzenten auf mächtige Elektronikkonzerne macht vor allem den Gewerkschaften zu schaffen. Immer mehr hochbezahlte Facharbeiter rutschen in niedrigere Lohngruppen ab. Benötigte etwa SEL bei der Montage seiner konventionellen Telephonanlagen noch 82 Prozent Fachkräfte, so sind es bei den Geräten der neuen Generation gerade noch 35 Prozent. IG-Metall-Steinkühler: „Das geht nicht, daß wir schöne Tarifverträge abschließen, und darunter kommt alles ins Rutschen.“

Selbst Bonns Technologie-Minister Matthöfer sorgt sich um das allzu rasante Innovationstempo — allerdings aus anderen Motiven. Er befürchtet, daß die Europäer auf Dauer in einen gefährlichen Rückstand zur elektronischen Welt-Konkurrenz geraten könnten.

Vor allem die amerikanische Industrie kann mit riesigen Elektronikaufträgen aus dem Rüstungsetat von jährlich 30 Milliarden Mark, davon zehn Milliarden für Forschung und Entwicklung, rechnen. Uwe Thomas, Spezialist für Nachrichtentechnik im Hause Matthöfer: „Die gehen gar kein Risiko ein, die sind völlig durchfinanziert.“ Dank der großzügigen Förderung verfügen die Amerikaner auf Teilgebieten der Industrie-Elektronik schon über einen Vorsprung von zwei Jahren.

Kaum weniger hart steigen die export-abhängigen Japaner ein. Fast eine Milliarde Mark wollen Staat und Industrie bis 1979 in die Entwicklung integrierter Schaltkreise mit noch größerer Leistung stecken. Gleichzeitig errichteten fünf Großkonzerne, darunter Hitachi und Mitsubishi, in der Nähe von Tokio ein gemeinsames Forschungszentrum. Technologie-Beamter Thomas: „Die haben eine Superstruktur hingestellt.“

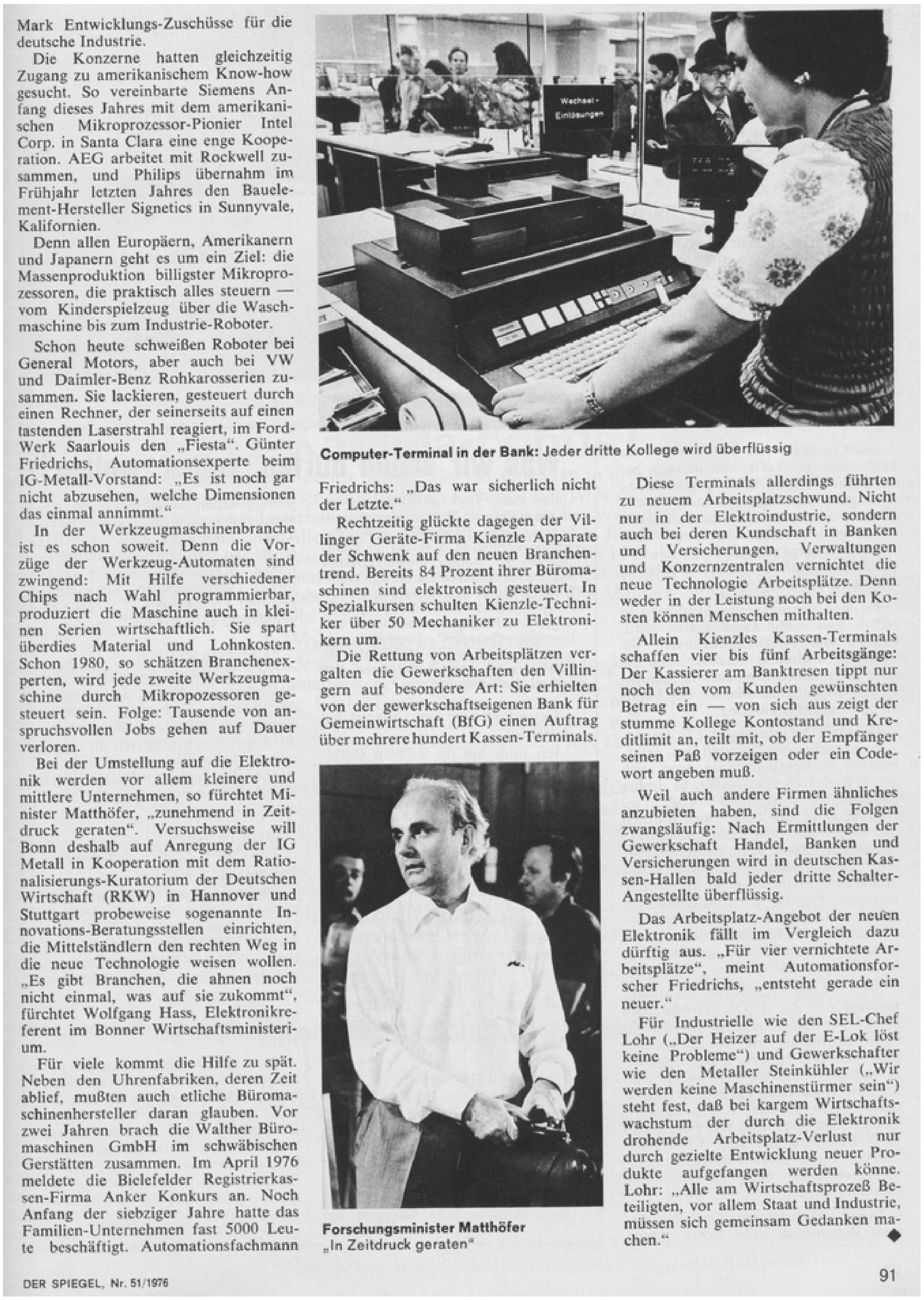
Um „einigermaßen Anschluß an die führenden Japaner und Amerikaner zu finden“, spendierte Elektronikförderer Matthöfer erst einmal 286 Millionen

Elektronische Schweißautomaten*: „Manche ahnen nicht, was auf sie zukommt“

* Bei General Motors.

DER SPIEGEL, Nr. 51/1976 89

Figure C.6: *The original image 534 used in the example segmentations.*



Mark Entwicklungs-Zuschüsse für die deutsche Industrie.

Die Konzerne hatten gleichzeitig Zugang zu amerikanischem Know-how gesucht. So vereinbarte Siemens Anfang dieses Jahres mit dem amerikanischen Mikroprozessor-Pionier Intel Corp. in Santa Clara eine enge Kooperation. AEG arbeitet mit Rockwell zusammen, und Philips übernahm im Frühjahr letzten Jahres den Bauelement-Hersteller Signetics in Sunnyvale, Kalifornien.

Denn allen Europäern, Amerikanern und Japanern geht es um ein Ziel: die Massenproduktion billigster Mikroprozessoren, die praktisch alles steuern — vom Kinderspielzeug über die Waschmaschine bis zum Industrie-Roboter.

Schon heute schweißen Roboter bei General Motors, aber auch bei VW und Daimler-Benz Rohkarosserien zusammen. Sie lackieren, gesteuert durch einen Rechner, der seinerseits auf einen tastenden Laserstrahl reagiert, im Ford-Werk Saarlouis den „Fiesta“. Günter Friedrichs, Automationsexperte beim IG-Metall-Vorstand: „Es ist noch gar nicht abzusehen, welche Dimensionen das einmal annimmt.“

In der Werkzeugmaschinenbranche ist es schon soweit. Denn die Vorzüge der Werkzeug-Automaten sind zwingend: Mit Hilfe verschiedener Chips nach Wahl programmierbar, produziert die Maschine auch in kleinen Serien wirtschaftlich. Sie spart überdies Material und Lohnkosten. Schon 1980, so schätzen Branchenexperten, wird jede zweite Werkzeugmaschine durch Mikropozessoren gesteuert sein. Folge: Tausende von anspruchsvollen Jobs gehen auf Dauer verloren.

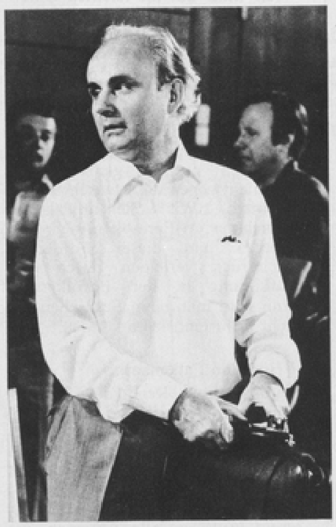
Bei der Umstellung auf die Elektronik werden vor allem kleinere und mittlere Unternehmen, so fürchtet Minister Matthöfer, „zunehmend in Zeitdruck geraten“. Versuchsweise will Bonn deshalb auf Anregung der IG Metall in Kooperation mit dem Rationalisierungs-Kuratorium der Deutschen Wirtschaft (RKW) in Hannover und Stuttgart probeweise sogenannte Innovations-Beratungsstellen einrichten, die Mittelständlern den rechten Weg in die neue Technologie weisen wollen. „Es gibt Branchen, die ahnen noch nicht einmal, was auf sie zukommt“, fürchtet Wolfgang Hass, Elektronikreferent im Bonner Wirtschaftsministerium.

Für viele kommt die Hilfe zu spät. Neben den Uhrenfabriken, deren Zeit ablief, mußten auch etliche Büromaschinenhersteller daran glauben. Vor zwei Jahren brach die Walther Büromaschinen GmbH im schwäbischen Gerstätten zusammen. Im April 1976 meldete die Bielefelder Registrierkassen-Firma Anker Konkurs an. Noch Anfang der siebziger Jahre hatte das Familien-Unternehmen fast 5000 Leute beschäftigt. Automationsfachmann Friedrichs: „Das war sicherlich nicht der Letzte.“

Computer-Terminal in der Bank: Jeder dritte Kollege wird überflüssig

Rechtzeitig glückte dagegen der Villinger Geräte-Firma Kienzle Apparate der Schwenk auf den neuen Branchentrend. Bereits 84 Prozent ihrer Büromaschinen sind elektronisch gesteuert. In Spezialkursen schulten Kienzle-Techniker über 50 Mechaniker zu Elektronikern um.

Die Rettung von Arbeitsplätzen vergalten die Gewerkschaften den Villingern auf besondere Art: Sie erhielten von der gewerkschaftseigenen Bank für Gemeinwirtschaft (BfG) einen Auftrag über mehrere hundert Kassen-Terminals.



Forschungsminister Matthöfer
„In Zeitdruck geraten“

Diese Terminals allerdings führten zu neuem Arbeitsplatzschwund. Nicht nur in der Elektroindustrie, sondern auch bei deren Kundschaft in Banken und Versicherungen, Verwaltungen und Konzernzentralen vernichtet die neue Technologie Arbeitsplätze. Denn weder in der Leistung noch bei den Kosten können Menschen mithalten.

Allein Kienzles Kassen-Terminals schaffen vier bis fünf Arbeitsgänge: Der Kassierer am Banktresen tippt nur noch den vom Kunden gewünschten Betrag ein — von sich aus zeigt der stumme Kollege Kontostand und Kreditlimit an, teilt mit, ob der Empfänger seinen Paß vorzeigen oder ein Codewort angeben muß.

Weil auch andere Firmen ähnliches anzubieten haben, sind die Folgen zwangsläufig: Nach Ermittlungen der Gewerkschaft Handel, Banken und Versicherungen wird in deutschen Kassen-Hallen bald jeder dritte Schalter-Angestellte überflüssig.

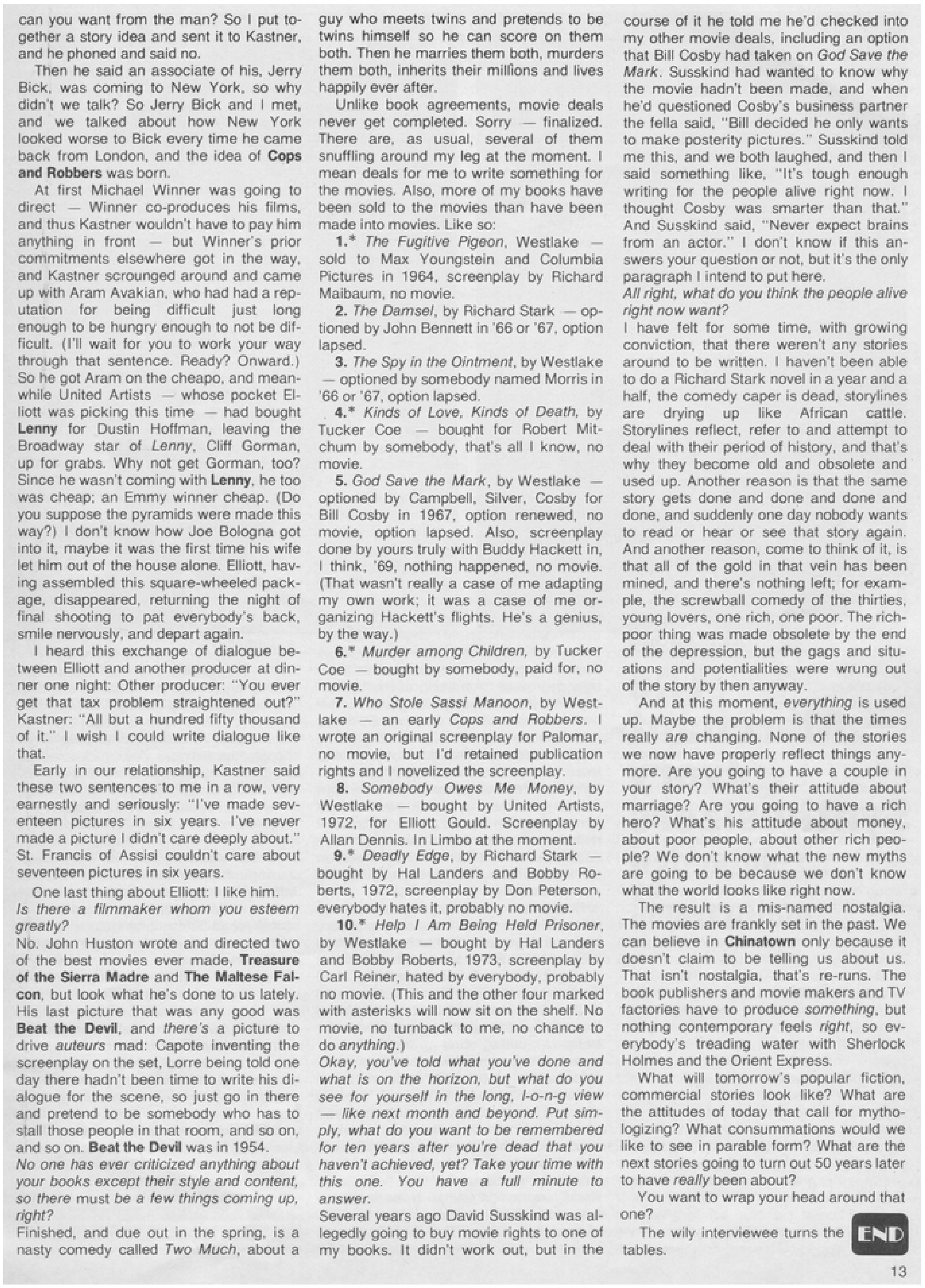
Das Arbeitsplatz-Angebot der neuen Elektronik fällt im Vergleich dazu dürftig aus. „Für vier vernichtete Arbeitsplätze“, meint Automationsforscher Friedrichs, „entsteht gerade ein neuer.“

Für Industrielle wie den SEL-Chef Lohr („Der Heizer auf der E-Lok löst keine Probleme“) und Gewerkschafter wie den Metaller Steinkühler („Wir werden keine Maschinenstürmer sein“) steht fest, daß bei kargem Wirtschaftswachstum der durch die Elektronik drohende Arbeitsplatz-Verlust nur durch gezielte Entwicklung neuer Produkte aufgefangen werden könne. Lohr: „Alle am Wirtschaftsprozeß Beteiligten, vor allem Staat und Industrie, müssen sich gemeinsam Gedanken machen.“ ◆

DER SPIEGEL, Nr. 51/1976

91

Figure C.7: *The original image 535 used in the example segmentations.*



can you want from the man? So I put together a story idea and sent it to Kastner, and he phoned and said no.

Then he said an associate of his, Jerry Bick, was coming to New York, so why didn't we talk? So Jerry Bick and I met, and we talked about how New York looked worse to Bick every time he came back from London, and the idea of **Cops and Robbers** was born.

At first Michael Winner was going to direct — Winner co-produces his films, and thus Kastner wouldn't have to pay him anything in front — but Winner's prior commitments elsewhere got in the way, and Kastner scrounged around and came up with Aram Avakian, who had had a reputation for being difficult just long enough to be hungry enough to not be difficult. (I'll wait for you to work your way through that sentence. Ready? Onward.) So he got Aram on the cheapo, and meanwhile United Artists — whose pocket Elliott was picking this time — had bought **Lenny** for Dustin Hoffman, leaving the Broadway star of *Lenny*, Cliff Gorman, up for grabs. Why not get Gorman, too? Since he wasn't coming with **Lenny**, he too was cheap; an Emmy winner cheap. (Do you suppose the pyramids were made this way?) I don't know how Joe Bologna got into it, maybe it was the first time his wife let him out of the house alone. Elliott, having assembled this square-wheeled package, disappeared, returning the night of final shooting to pat everybody's back, smile nervously, and depart again.

I heard this exchange of dialogue between Elliott and another producer at dinner one night: Other producer: "You ever get that tax problem straightened out?" Kastner: "All but a hundred fifty thousand of it." I wish I could write dialogue like that.

Early in our relationship, Kastner said these two sentences to me in a row, very earnestly and seriously: "I've made seventeen pictures in six years. I've never made a picture I didn't care deeply about." St. Francis of Assisi couldn't care about seventeen pictures in six years.

One last thing about Elliott: I like him.

Is there a filmmaker whom you esteem greatly?

No. John Huston wrote and directed two of the best movies ever made, **Treasure of the Sierra Madre** and **The Maltese Falcon**, but look what he's done to us lately. His last picture that was any good was **Beat the Devil**, and *there's* a picture to drive *auteurs* mad: Capote inventing the screenplay on the set, Lorre being told one day there hadn't been time to write his dialogue for the scene, so just go in there and pretend to be somebody who has to stall those people in that room, and so on, and so on. **Beat the Devil** was in 1954.

No one has ever criticized anything about your books except their style and content, so there must be a few things coming up, right?

Finished, and due out in the spring, is a nasty comedy called *Two Much*, about a guy who meets twins and pretends to be twins himself so he can score on them both. Then he marries them both, murders them both, inherits their millions and lives happily ever after.

Unlike book agreements, movie deals never get completed. Sorry — finalized. There are, as usual, several of them snuffling around my leg at the moment. I mean deals for me to write something for the movies. Also, more of my books have been sold to the movies than have been made into movies. Like so:

1.* *The Fugitive Pigeon*, Westlake — sold to Max Youngstein and Columbia Pictures in 1964, screenplay by Richard Maibaum, no movie.

2. *The Damsel*, by Richard Stark — optioned by John Bennett in '66 or '67, option lapsed.

3. *The Spy in the Ointment*, by Westlake — optioned by somebody named Morris in '66 or '67, option lapsed.

4.* *Kinds of Love, Kinds of Death*, by Tucker Coe — bought for Robert Mitchum by somebody, that's all I know, no movie.

5. *God Save the Mark*, by Westlake — optioned by Campbell, Silver, Cosby for Bill Cosby in 1967, option renewed, no movie, option lapsed. Also, screenplay done by yours truly with Buddy Hackett in, I think, '69, nothing happened, no movie. (That wasn't really a case of me adapting my own work; it was a case of me organizing Hackett's flights. He's a genius, by the way.)

6.* *Murder among Children*, by Tucker Coe — bought by somebody, paid for, no movie.

7. *Who Stole Sassi Manoon*, by Westlake — an early *Cops and Robbers*. I wrote an original screenplay for Palomar, no movie, but I'd retained publication rights and I novelized the screenplay.

8. *Somebody Owes Me Money*, by Westlake — bought by United Artists, 1972, for Elliott Gould. Screenplay by Allan Dennis. In Limbo at the moment.

9.* *Deadly Edge*, by Richard Stark — bought by Hal Landers and Bobby Roberts, 1972, screenplay by Don Peterson, everybody hates it, probably no movie.

10.* *Help I Am Being Held Prisoner*, by Westlake — bought by Hal Landers and Bobby Roberts, 1973, screenplay by Carl Reiner, hated by everybody, probably no movie. (This and the other four marked with asterisks will now sit on the shelf. No movie, no turnback to me, no chance to do *anything*.)

Okay, you've told what you've done and what is on the horizon, but what do you see for yourself in the long, l-o-n-g view — like next month and beyond. Put simply, what do you want to be remembered for ten years after you're dead that you haven't achieved, yet? Take your time with this one. You have a full minute to answer.

Several years ago David Susskind was allegedly going to buy movie rights to one of my books. It didn't work out, but in the course of it he told me he'd checked into my other movie deals, including an option that Bill Cosby had taken on *God Save the Mark*. Susskind had wanted to know why the movie hadn't been made, and when he'd questioned Cosby's business partner the fella said, "Bill decided he only wants to make posterity pictures." Susskind told me this, and we both laughed, and then I said something like, "It's tough enough writing for the people alive right now. I thought Cosby was smarter than that." And Susskind said, "Never expect brains from an actor." I don't know if this answers your question or not, but it's the only paragraph I intend to put here.

All right, what do you think the people alive right now want?

I have felt for some time, with growing conviction, that there weren't any stories around to be written. I haven't been able to do a Richard Stark novel in a year and a half, the comedy caper is dead, storylines are drying up like African cattle. Storylines reflect, refer to and attempt to deal with their period of history, and that's why they become old and obsolete and used up. Another reason is that the same story gets done and done and done and done, and suddenly one day nobody wants to read or hear or see that story again. And another reason, come to think of it, is that all of the gold in that vein has been mined, and there's nothing left; for example, the screwball comedy of the thirties, young lovers, one rich, one poor. The rich-poor thing was made obsolete by the end of the depression, but the gags and situations and potentialities were wrung out of the story by then anyway.

And at this moment, *everything* is used up. Maybe the problem is that the times really *are* changing. None of the stories we now have properly reflect things anymore. Are you going to have a couple in your story? What's their attitude about marriage? Are you going to have a rich hero? What's his attitude about money, about poor people, about other rich people? We don't know what the new myths are going to be because we don't know what the world looks like right now.

The result is a mis-named nostalgia. The movies are frankly set in the past. We can believe in **Chinatown** only because it doesn't claim to be telling us about us. That isn't nostalgia, that's re-runs. The book publishers and movie makers and TV factories have to produce *something*, but nothing contemporary feels *right*, so everybody's treading water with Sherlock Holmes and the Orient Express.

What will tomorrow's popular fiction, commercial stories look like? What are the attitudes of today that call for mythologizing? What consummations would we like to see in parable form? What are the next stories going to turn out 50 years later to have *really* been about?

You want to wrap your head around that one?

The wily interviewee turns the tables.

END

13

Figure C.8: *The original image 548 used in the example segmentations.*

Appendix D

K-Means based method results

The following pages contain the post-processed K-Means based segmentations of the images in appendix C.

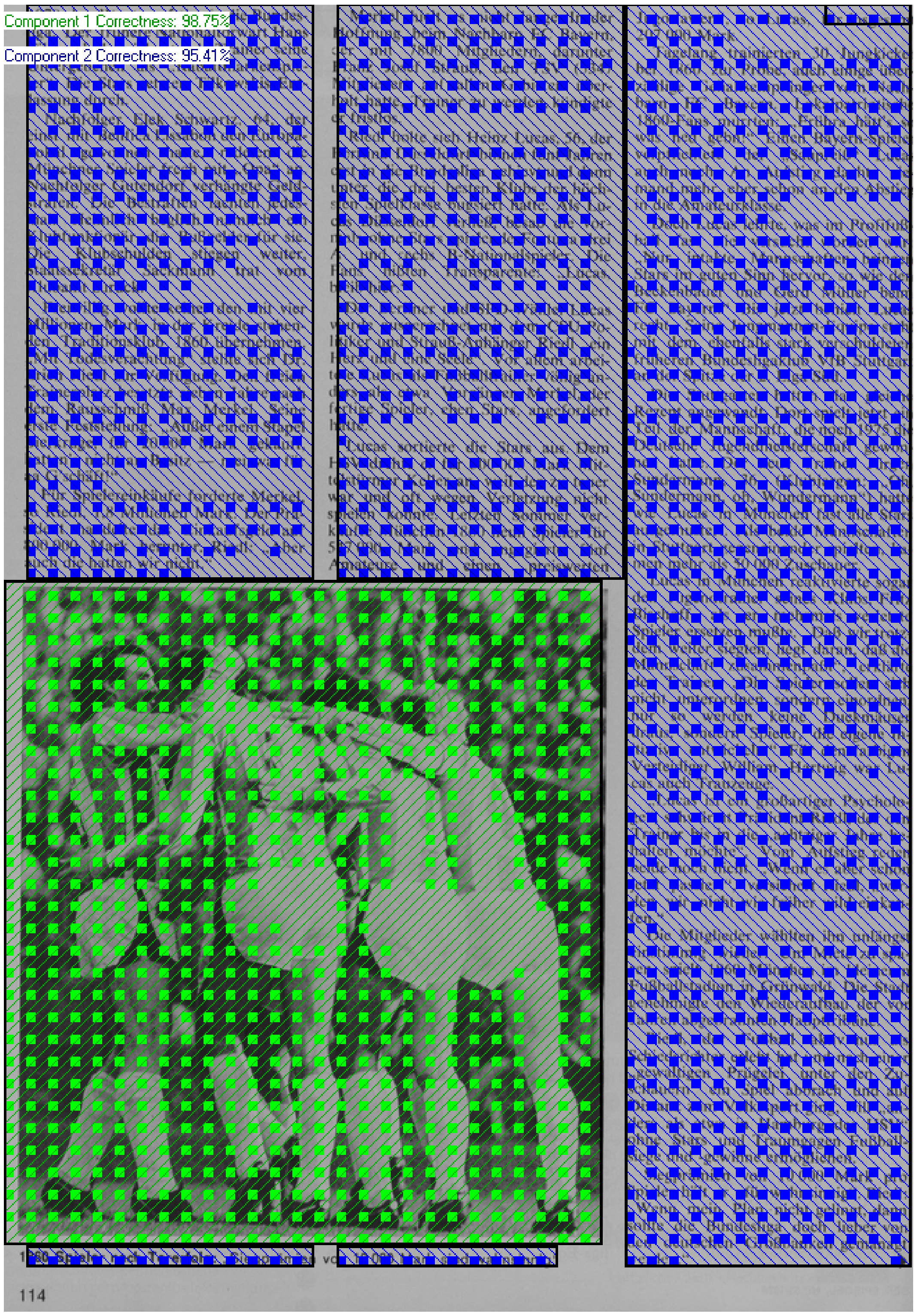

Component 1 Correctness: 98.75%
Component 2 Correctness: 95.41%
114

Figure D.1: *Image 528 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm.*

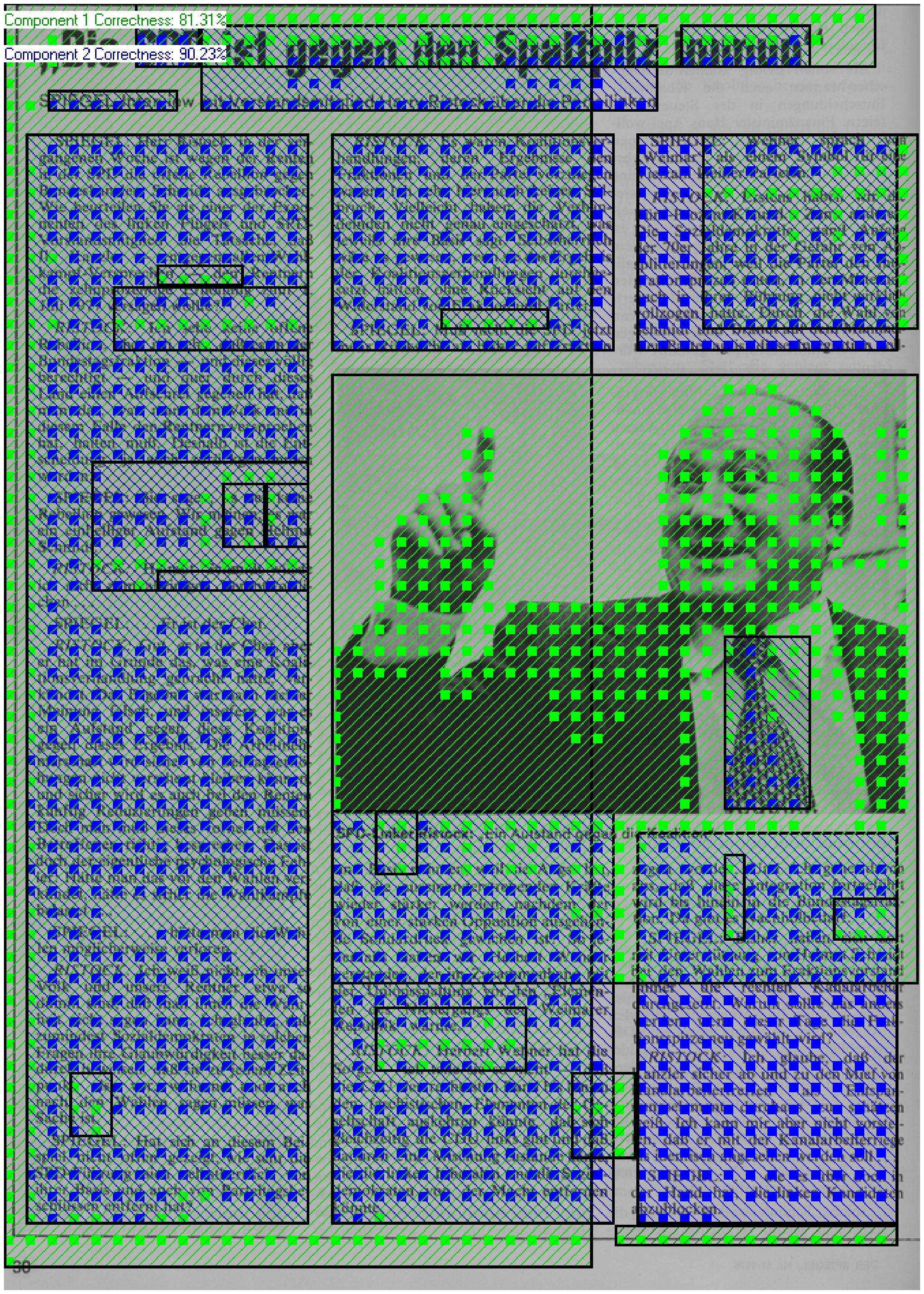

Component 1 Correctness: 81.31%
Component 2 Correctness: 90.23%

Figure D.2: *Image 531 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm.*

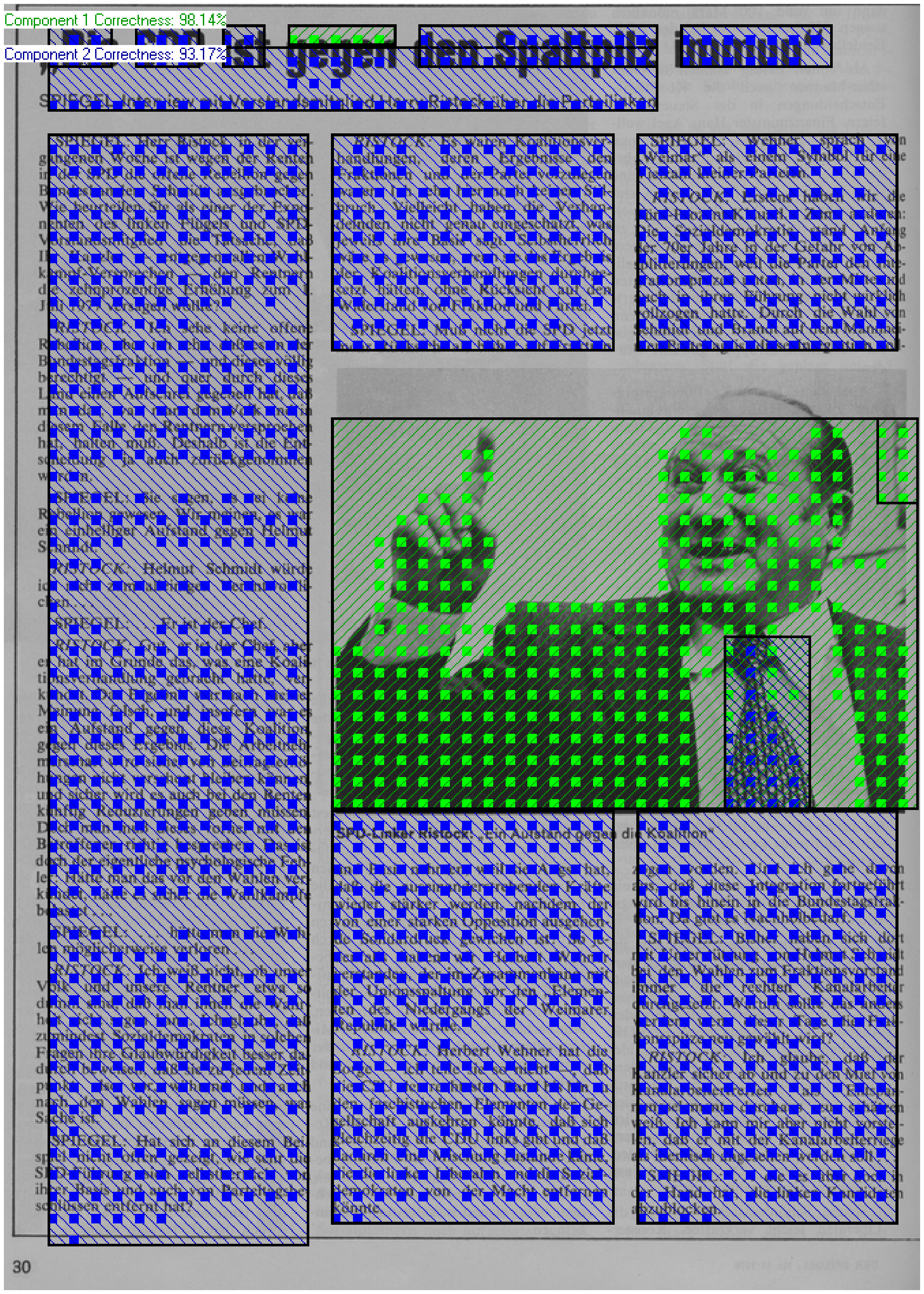

Component 1 Correctness: 98.14%
Component 2 Correctness: 93.17%

Figure D.3: *Image 531 (see appendix C for the original image) segmented (again) by the post-processed K-Means based algorithm.*

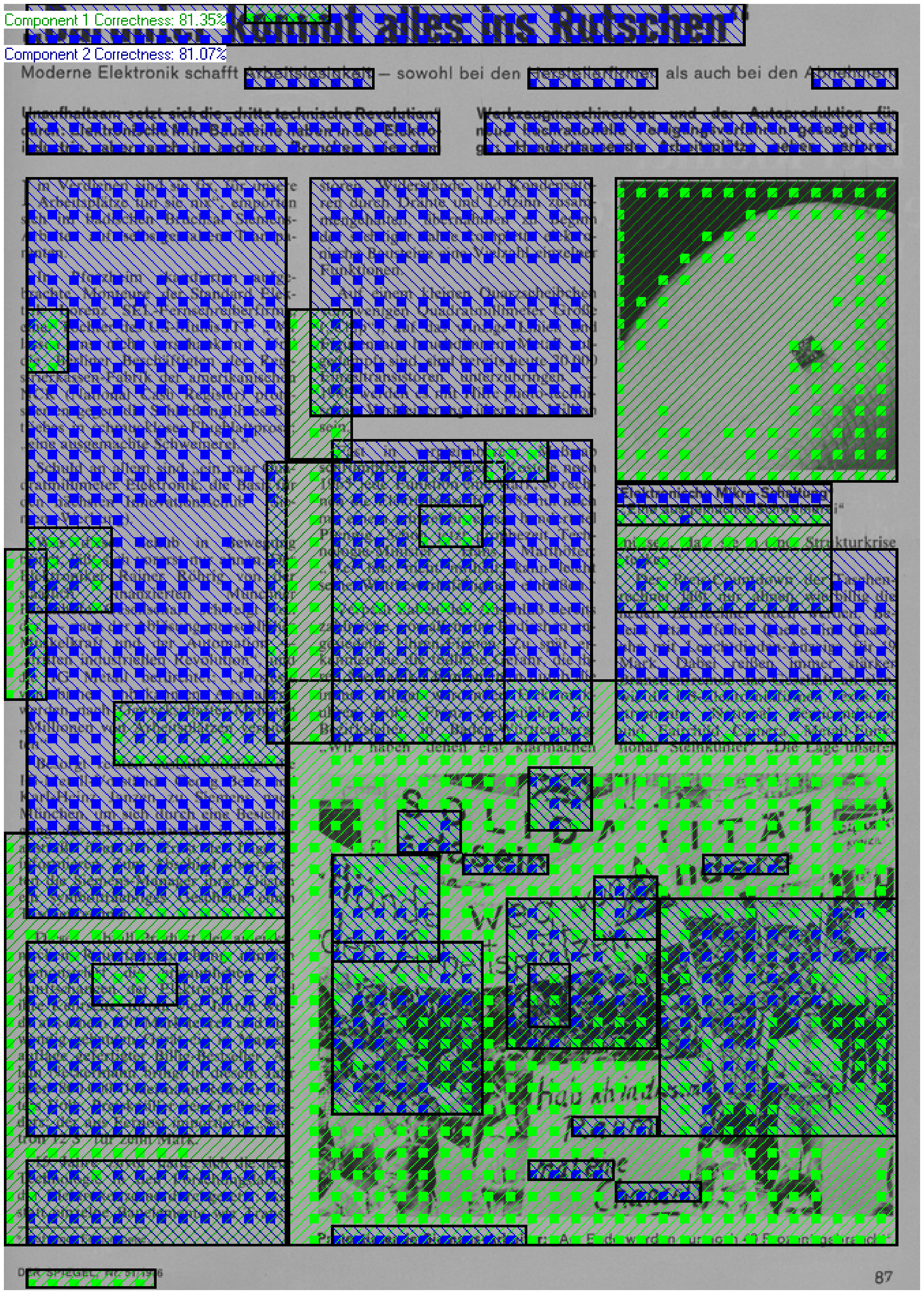

Component 1 Correctness: 81.35%
Component 2 Correctness: 81.07%
Moderne Elektronik schafft Arbeitslosigkeit — sowohl bei den Herstellerfirmen als auch bei den Abnehmern
87

Figure D.4: *Image 533 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm.*

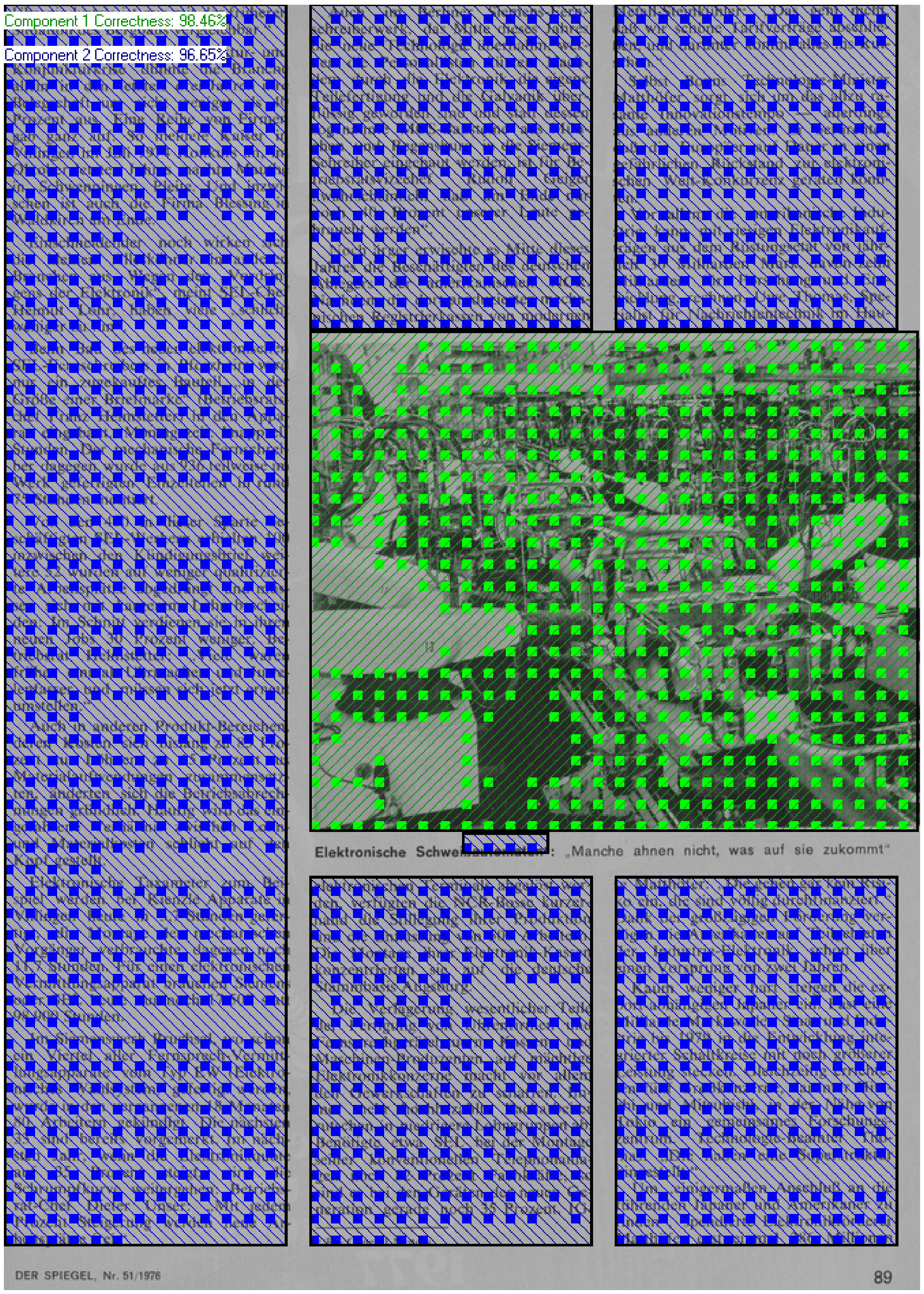

Component 1 Correctness: 98.46%
Component 2 Correctness: 96.65%
Elektronische Schweißautomaten: „Manche ahnen nicht, was auf sie zukommt"
DER SPIEGEL, Nr. 51/1976
89

Figure D.5: *Image 534 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm.*

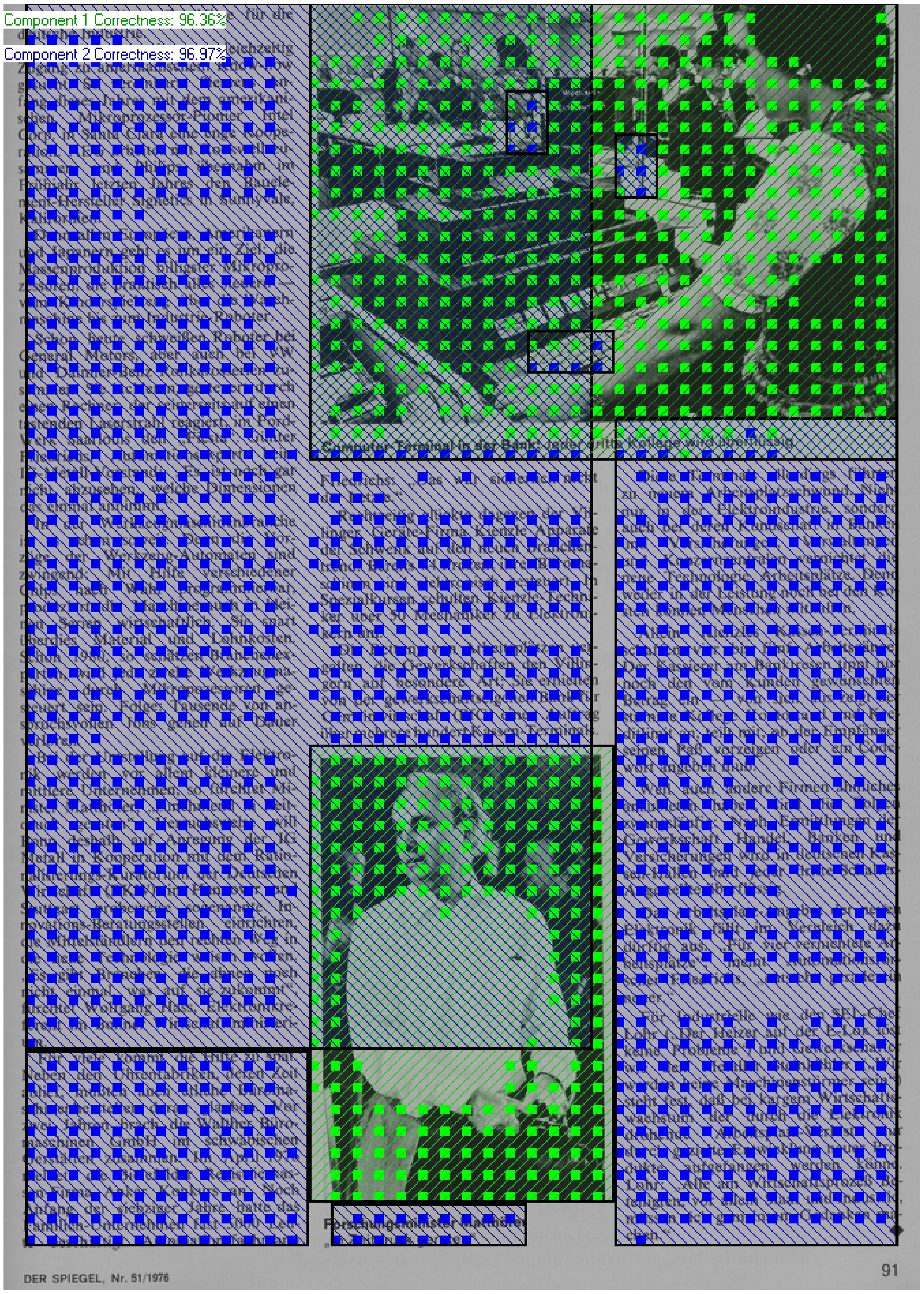

Component 1 Correctness: 96.36%
Component 2 Correctness: 96.97%
DER SPIEGEL, Nr. 51/1976
91

Figure D.6: *Image 535 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm.*

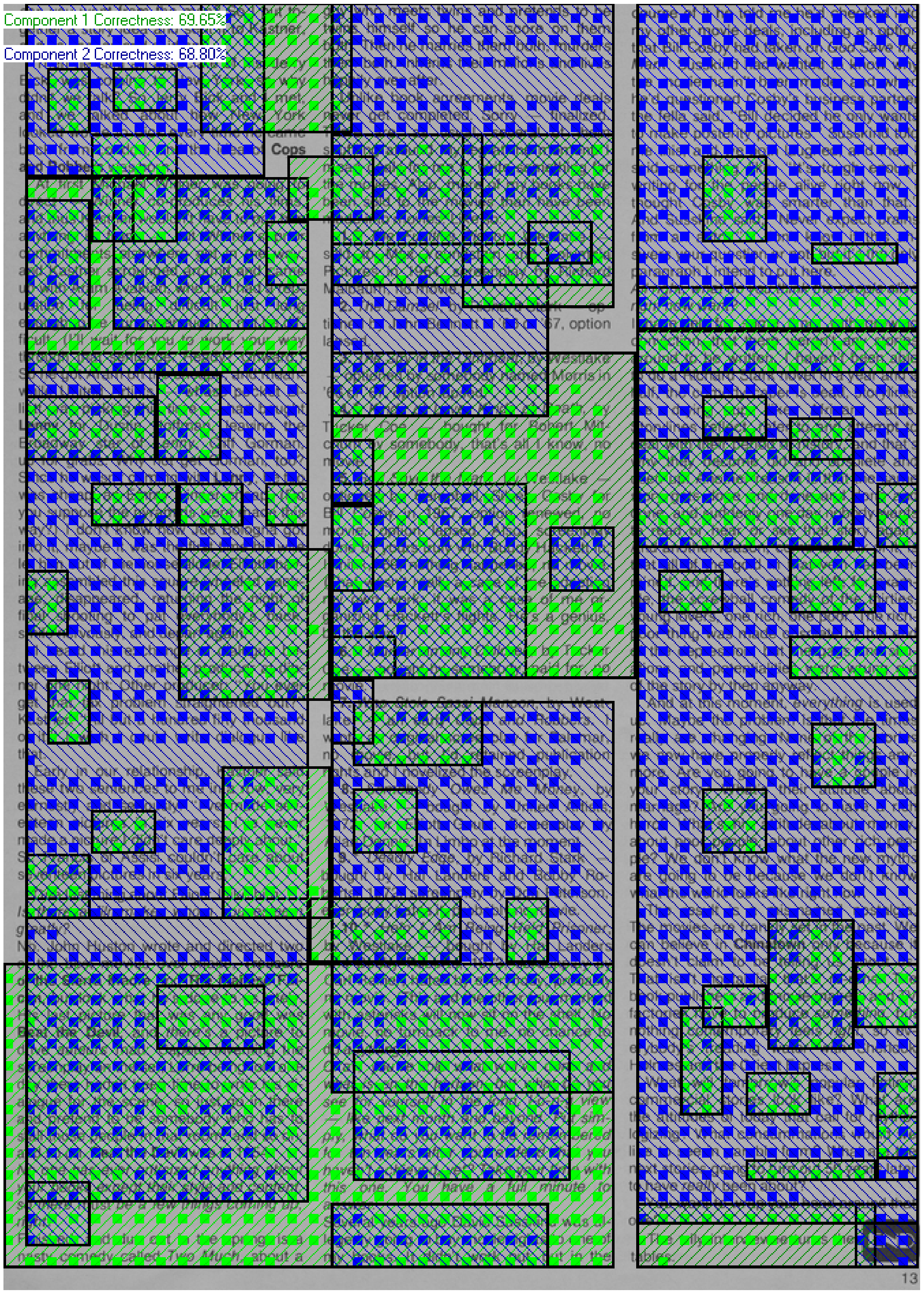

Component 1 Correctness: 69.65%
Component 2 Correctness: 68.80%

Figure D.7: *Image 548 (see appendix C for the original image) segmented by the post-processed K-Means based algorithm. This image demonstrates the problem that occurs when attempting to segment two natural clusters (text and background) into three clusters (text, image and background) via the K-Means or C-Means algorithms.*

Appendix E

Genetic programming based method results

The following pages contain the genetic programming based segmentations of the images in appendix C.

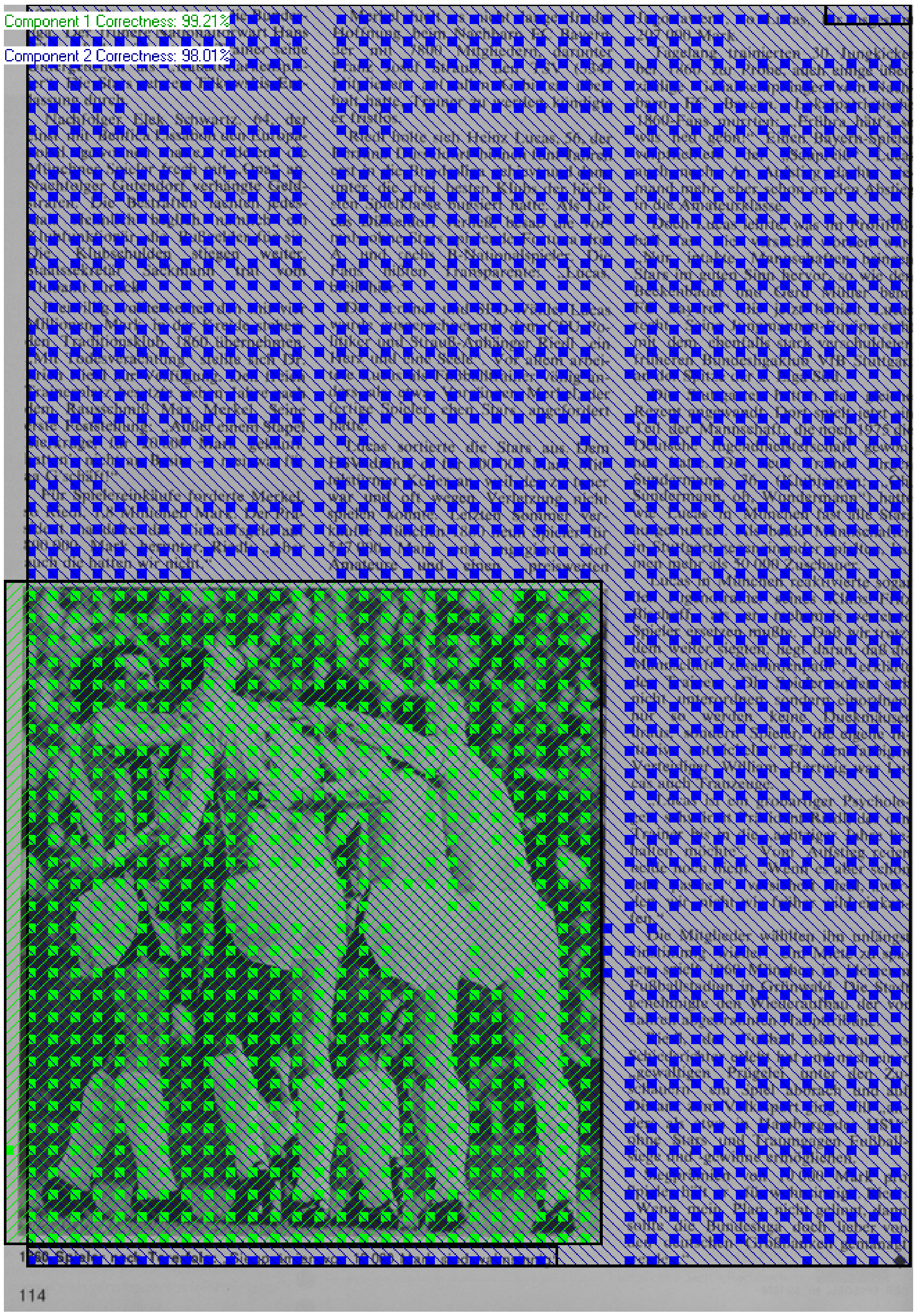

Component 1 Correctness: 99.21%
Component 2 Correctness: 98.01%

Figure E.1: *Image 528 (see section C for the original image) segmented by the genetic programming based algorithm.*

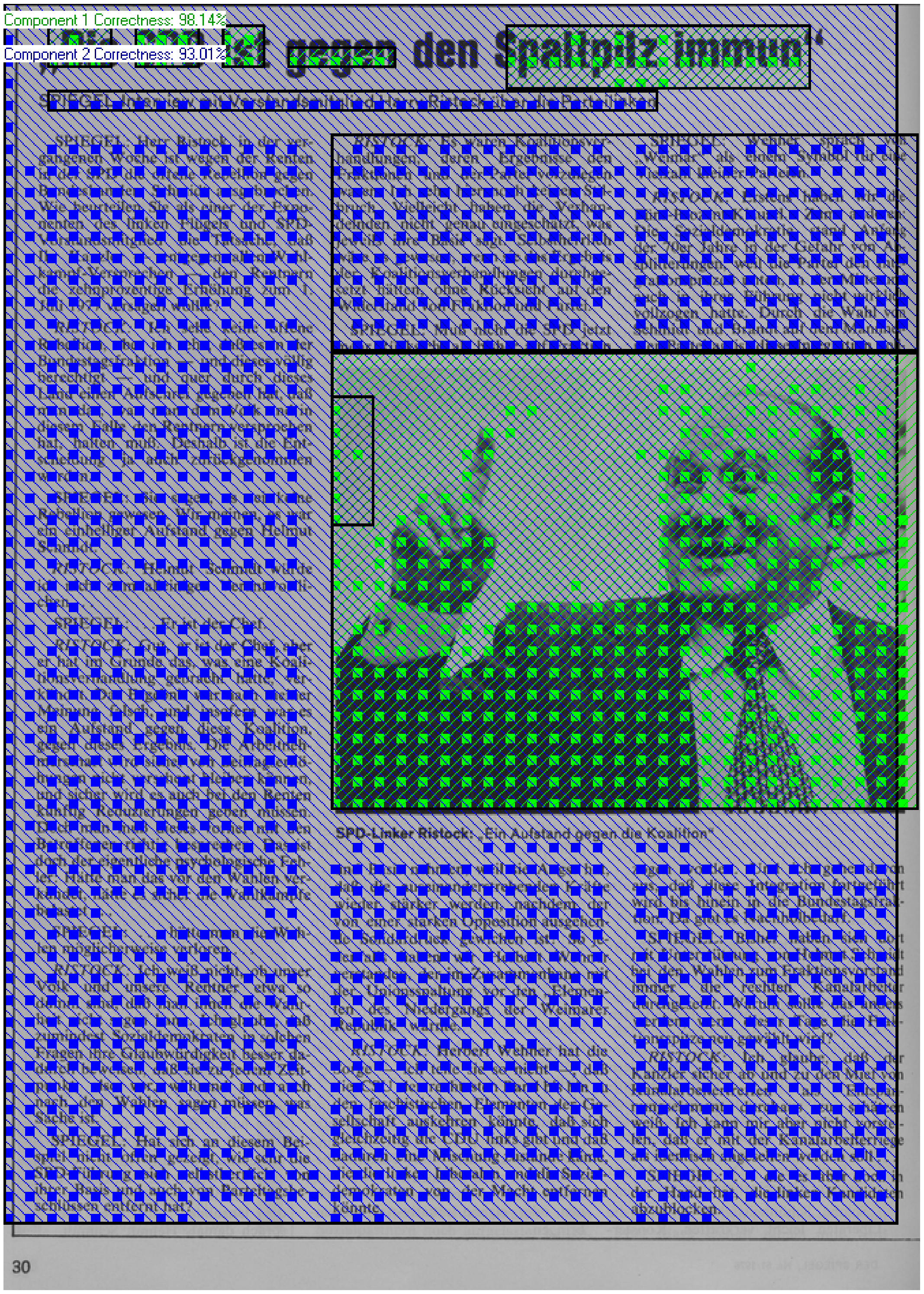

Component 1 Correctness: 98.14%
Component 2 Correctness: 93.01%
„Die SPD ist gegen den Spaltpilz immun“
SPIEGEL-Interview mit Vorstandsmitglied Harry Ristock über die Parteilinke
SPD-Linker Ristock: „Ein Aufstand gegen die Koalition“
30

Figure E.2: *Image 531 (see section C for the original image) segmented by the genetic programming based algorithm.*

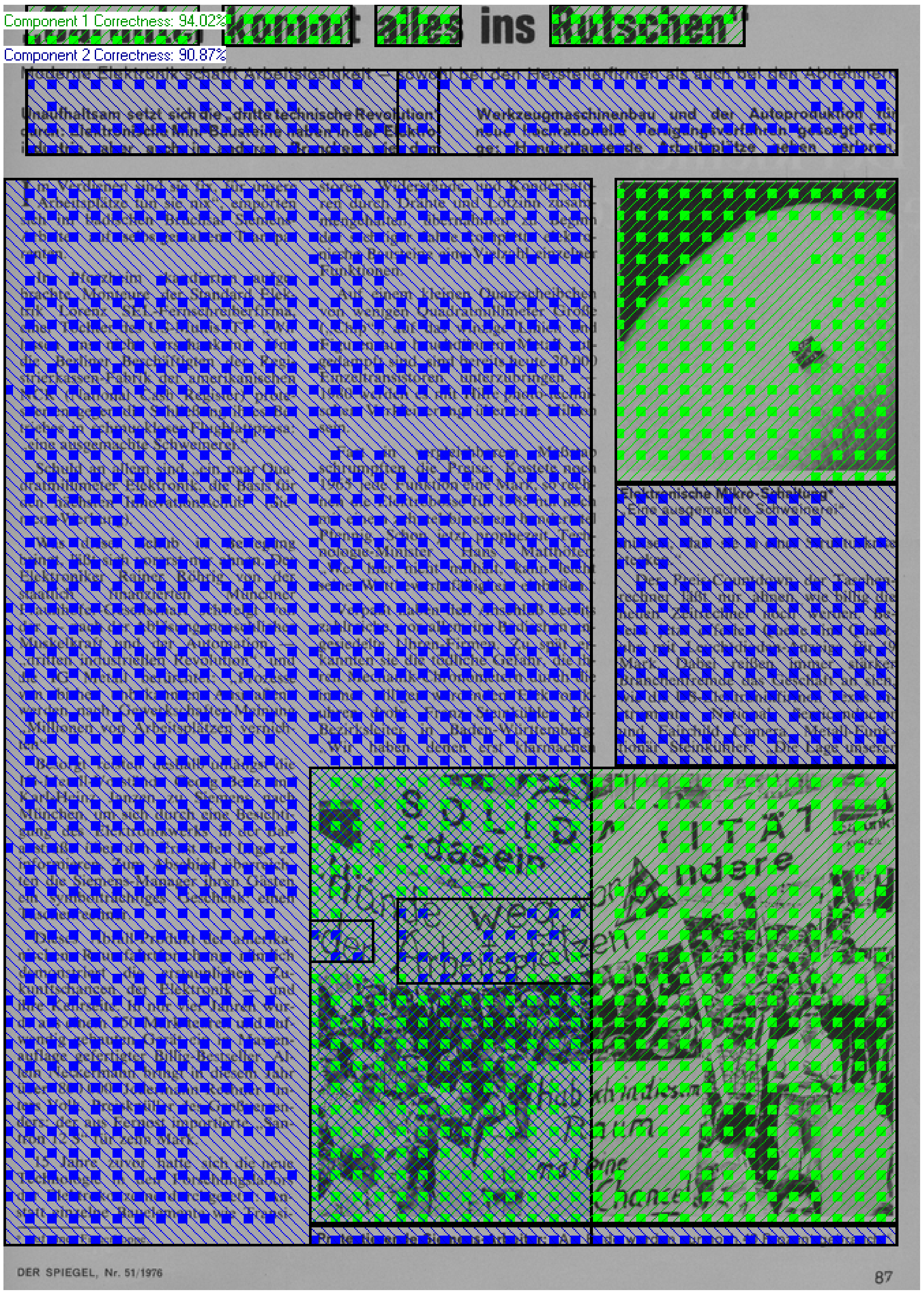

Component 1 Correctness: 94.02%
Component 2 Correctness: 90.87%
DER SPIEGEL, Nr. 51/1976
87

Figure E.3: *Image 533 (see section C for the original image) segmented by the genetic programming based algorithm.*

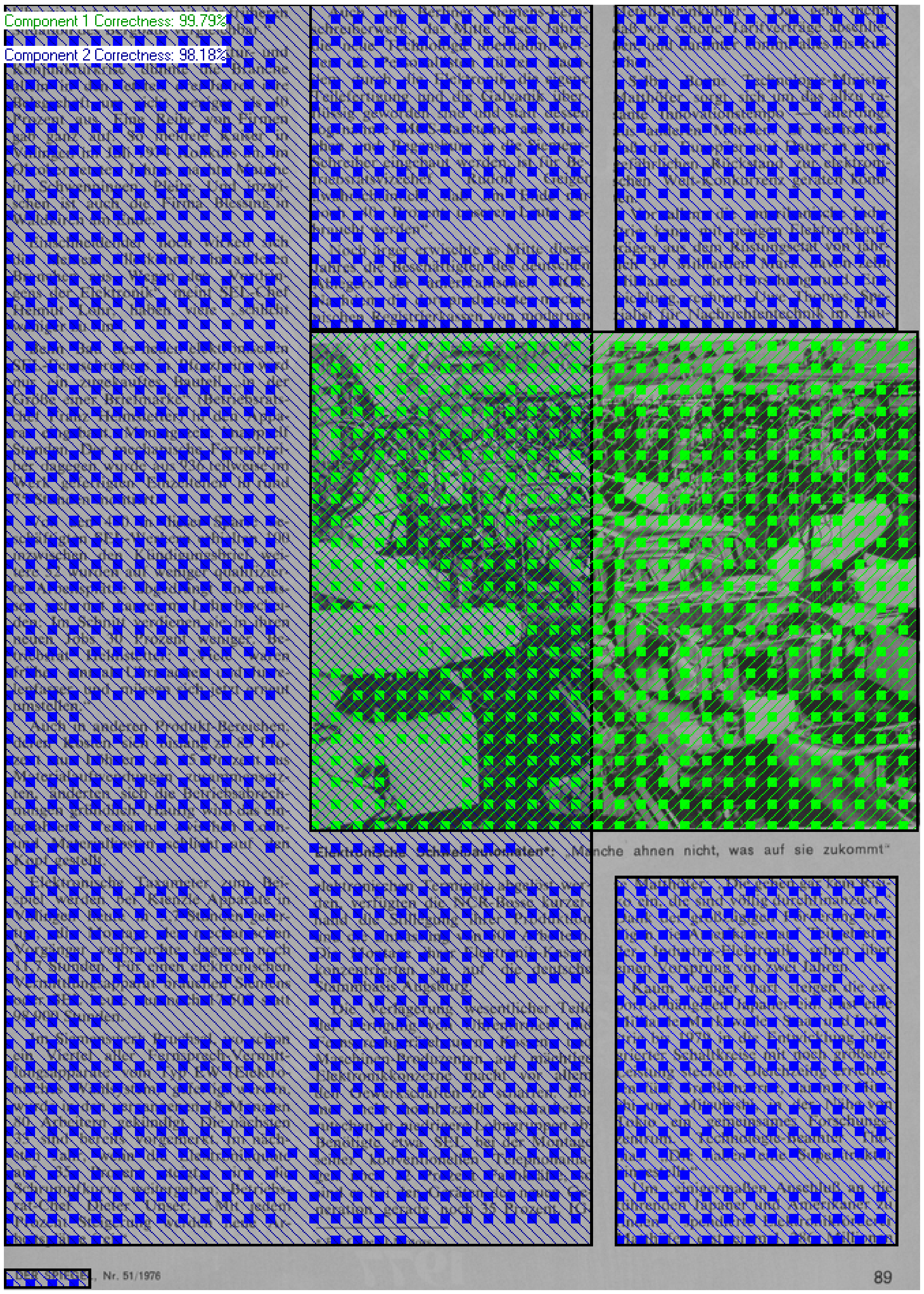

Component 1 Correctness: 99.79%
Component 2 Correctness: 98.18%
Elektronische Schweißautomaten*: „Manche ahnen nicht, was auf sie zukommt"
Nr. 51/1976
89

Figure E.4: *Image 534 (see section C for the original image) segmented by the genetic programming based algorithm.*

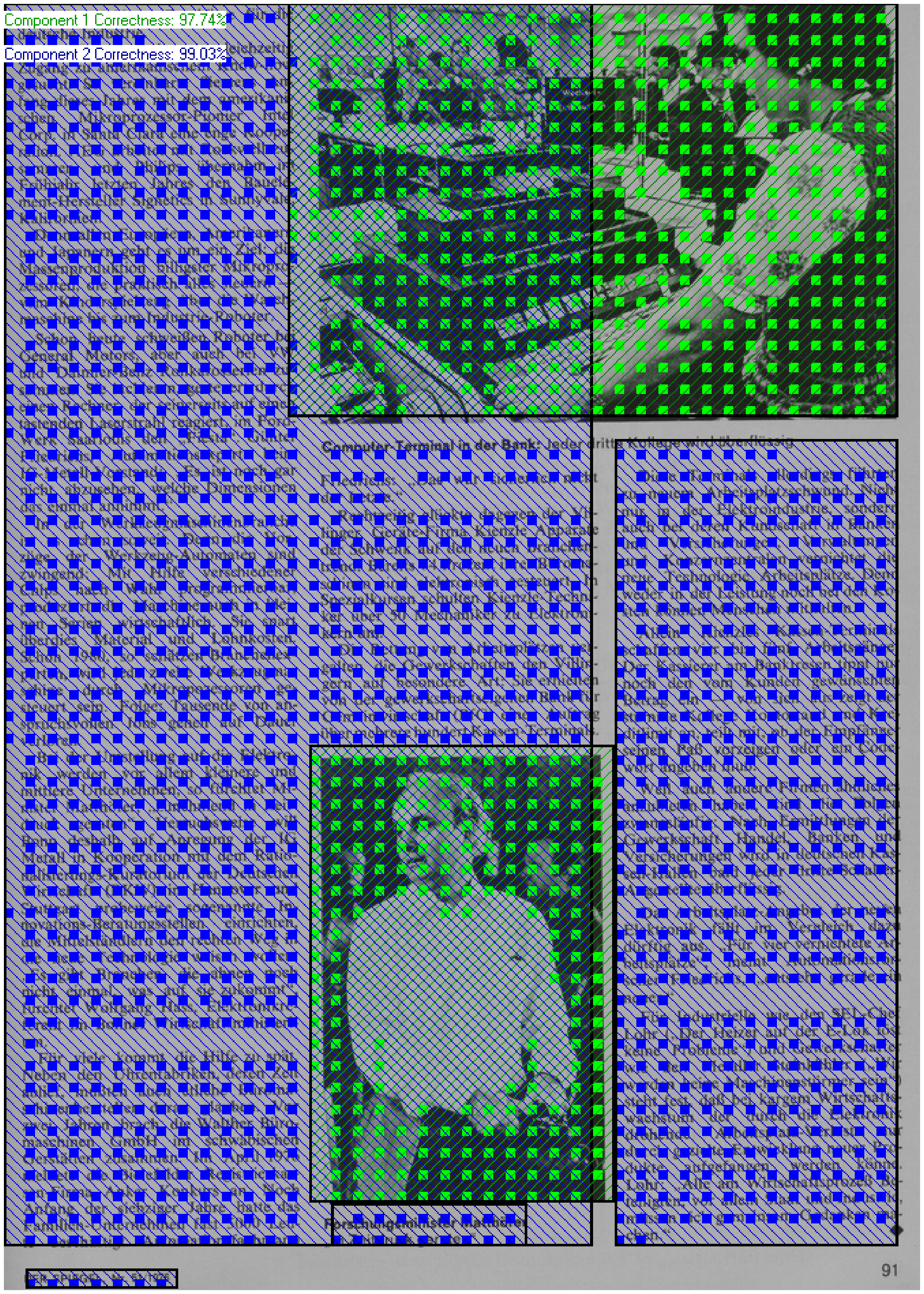

Component 1 Correctness: 97.74%
Component 2 Correctness: 99.03%

Figure E.5: *Image 535 (see section C for the original image) segmented by the genetic programming based algorithm.*

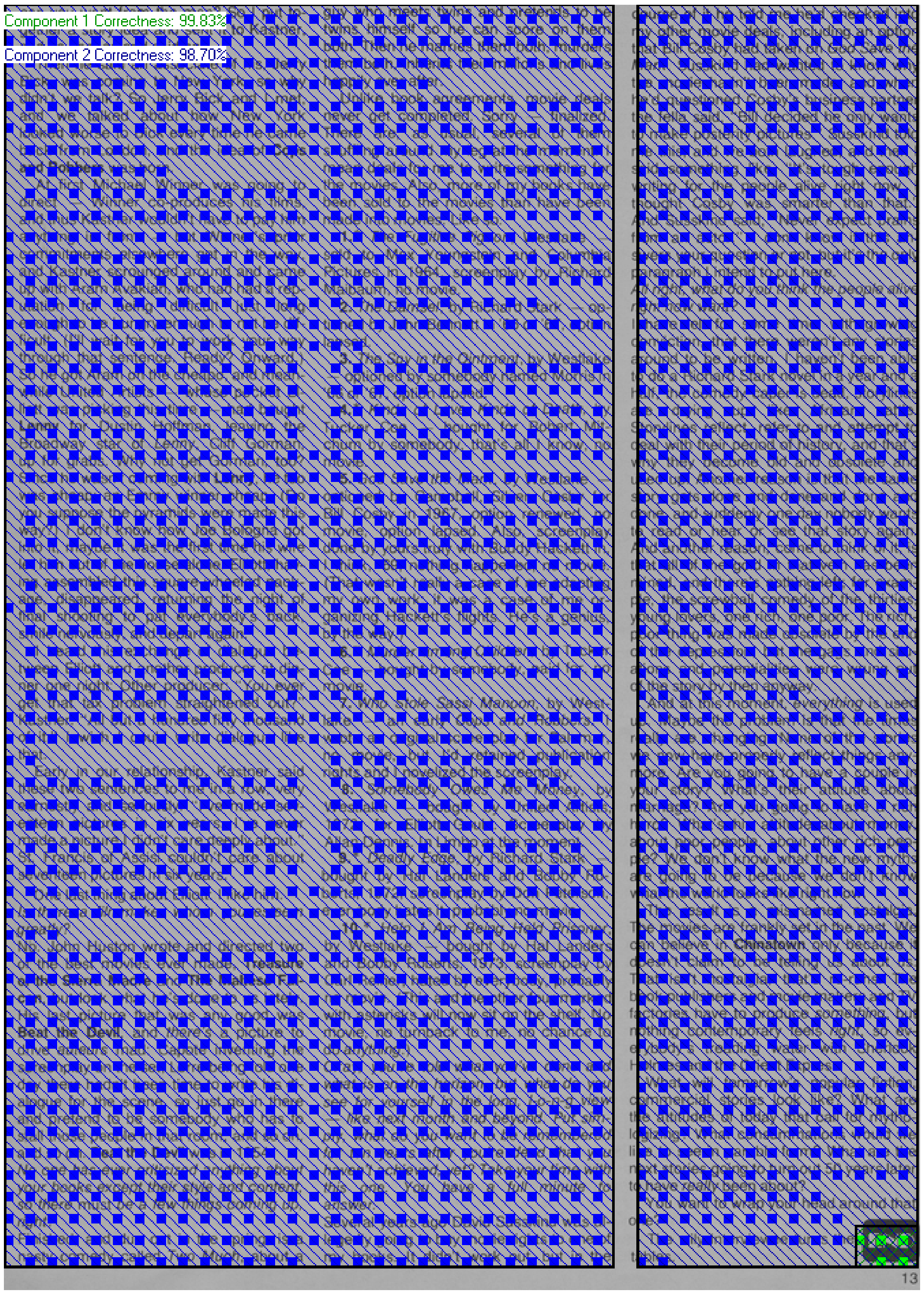

Component 1 Correctness: 99.83%
Component 2 Correctness: 98.70%

Figure E.6: *Image 548 (see section C for the original image) segmented by the genetic programming based algorithm.*